DTFH61-03-C-00138

Enhancement of the TEXAS Model for Simulating Intersection Collisions, Driver Interaction with Messaging, and ITS Sensors - Final Report

Thomas W. Rioux, Ph.D., P.E.

Research Report Number DTFH61-03-C-00138

conducted for

United States Department of Transportation
Federal Highway Administration

Solicitation Number DTFH61-03-R-00117
Contract Number DTFH61-03-C-00138

Center for Transportation Research
The University of Texas at Austin, Austin, Texas

August 2005

This page intentionally left blank to facilitate printing on 2 sides.

PRELIMINARY REVIEW COPY

Technical Report Documentation Page

1. Report No.
DTFH61-03-C-00138

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
ENHANCEMENT OF THE TEXAS MODEL FOR SIMULATING INTERSECTION COLLISIONS, DRIVER INTERACTION WITH MESSAGING, AND ITS SENSORS - FINAL REPORT

5. Report Date
August 2005

6. Performing Organization Code

7. Author(s)
Thomas W. Rioux, Ph.D., P.E.

8. Performing Organization Report No.
DTFH61-03-C-00138

9. Performing Organization Name and Address
Center for Transportation Research
The University of Texas at Austin
3208 Red River Street
Austin, Texas 78705-2698

10. Work Unit No. (TRAIS)

11. Contract or Grant No.
DTFH61-03-C-00138

12. Sponsoring Agency Name and Address
Office of Operations R&D
Enabling Technologies Team (HRDO-4)
Federal Highway Administration
6300 Georgetown Pike
McLean, Virginia 22101-2296

13. Type of Report and Period Covered
Final Report
April 2004–September 2005

14. Sponsoring Agency Code

15. Supplementary Notes
Contracting Officer's Technical Representative (COTR) - David Gibson, FHWA R&D (HRDO-4) Room T204.

16. Abstract
The TEXAS Model for Intersection Traffic is a powerful traffic engineering analysis tool with a uniquely high resolution simulation of driver/vehicle actions. The primary purpose of this project is to have the TEXAS Model accurately simulate crashes. The second purpose is to provide tools for modeling interactions between the driver and the traffic control system during the period just prior to the crash. The third purpose is to measure the vehicles' actions with high-fidelity Intelligent Transportation Systems (ITS) sensors.

In order to model crashes, cars must be made to stop when the driver ordinarily would go, and to go when the driver ordinarily would stop. To model intersection collision avoidance, the driver must be made to respond to Vehicle Message System (VMS) and to dynamic message signs (DMS). These capabilities are critical to allow modeling of dangerous responses to a traffic signal, to approaching traffic, and to proactive infrastructure system interventions. Behaviors such as red light running or turning in front of a close, rapidly approaching vehicle are typical dangerous behaviors.

17. Key Words
TEXAS Model for Intersection Traffic, microscopic traffic simulation, ITS, Java, FORTRAN, Linux, Windows

18. Distribution Statement
No restrictions. This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.

19. Security Classif. (of report)	20. Security Classif. (this page)	21. No. of pages	22. Price
Unclassified	Unclassified	91	

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

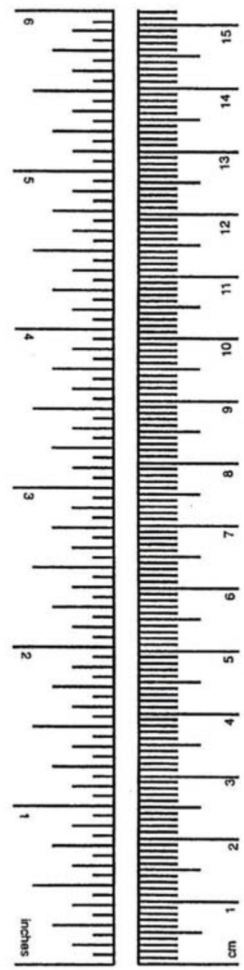
METRIC (SI*) CONVERSION FACTORS

APPROXIMATE CONVERSIONS TO SI UNITS

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
in	inches	2.54	centimeters	cm
ft	feet	0.3048	meters	m
yd	yards	0.914	meters	m
mi	miles	1.61	kilometers	km
		AREA		
in^2	square inches	645.2	millimeters squared	mm^2
ft^2	square feet	0.0929	meters squared	m^2
yd^2	square yards	0.836	meters squared	m^2
mi^2	square miles	2.59	kilometers squared	km^2
ac	acres	0.395	hectares	ha
		MASS (weight)		
oz	ounces	28.35	grams	g
lb	pounds	0.454	kilograms	kg
T	short tons (2,000 lb)	0.907	megagrams	Mg
		VOLUME		
fl oz	fluid ounces	29.57	milliliters	mL
gal	gallons	3.785	liters	L
ft^3	cubic feet	0.0328	meters cubed	m^3
yd^3	cubic yards	0.0765	meters cubed	m^3

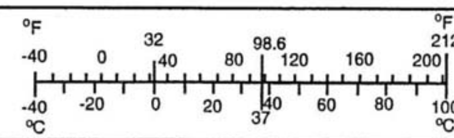
NOTE: Volumes greater than 1,000 L shall be shown in m^3.

Symbol	When You Know	Multiply by	To Find	Symbol
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C



APPROXIMATE CONVERSIONS FROM SI UNITS

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
mm	millimeters	0.039	inches	in
m	meters	3.28	feet	ft
m	meters	1.09	yards	yd
km	kilometers	0.621	miles	mi
		AREA		
mm^2	millimeters squared	0.0016	square inches	in^2
m^2	meters squared	10.764	square feet	ft^2
m^2	meters squared	1.20	square yards	yd^2
km^2	kilometers squared	0.39	square miles	mi^2
ha	hectares (10,000 m^2)	2.53	acres	ac
		MASS (weight)		
g	grams	0.0353	ounces	oz
kg	kilograms	2.205	pounds	lb
Mg	megagrams (1,000 kg)	1.103	short tons	T
		VOLUME		
mL	milliliters	0.034	fluid ounces	fl oz
L	liters	0.264	gallons	gal
m^3	meters cubed	35.315	cubic feet	ft^3
m^3	meters cubed	1.308	cubic yards	yd^3
		TEMPERATURE (exact)		
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F



These factors conform to the requirement of FHWA Order 5190.1A.

* SI is the symbol for the International System of Measurements

Enhancement of the TEXAS Model for Simulating Intersection Collisions, Driver Interaction with Messaging, and ITS Sensors - Final Report

By
Thomas W. Rioux, Ph.D., P.E.

Research Report Number DTFH61-03-C-00138

conducted for

United States Department of Transportation
Federal Highway Administration

Solicitation Number DTFH61-03-R-00117
Contract Number DTFH61-03-C-00138

by
Center for Transportation Research
The University of Texas at Austin, Austin, Texas

August 2005

Prepared in cooperation with the United States Department of Transportation, Federal Highway Administration.

TABLE OF CONTENTS

This page intentionally left blank to facilitate printing on 2 sides.

LIST OF FIGURES

This page intentionally left blank to facilitate printing on 2 sides.

CHAPTER 1. INTRODUCTION

Initial development of the Traffic EXperimental Analytical Simulation (TEXAS) Model for Intersection Traffic began in 1971 under Research Study 3-18-72-184, and four reports documenting this mainframe, batch-processing version of the microscopic computer model were published in 1977 [1,2,3,4]. Because of the cumbersome communication with the TEXAS Model through hand-written coding forms, it was not extensively utilized other than by researchers. Research Study 3-18-84-361 produced a "User-Friendly" TEXAS Model in 1985 that allows both input to the model and running of the model on an IBM-compatible microcomputer as well as on mainframe machines [5]. This version also features an animated-graphics display of intersection traffic activities on the screen of a microcomputer. Research Study 3-18-84-443, entitled "TEXAS Diamond - A Microscopic Simulation Model for Diamond Interchanges", Study Report 443-1F, "TEXAS Model, Version 3.0 (Diamond Interchanges)" was published in January of 1989 [6]. Version 3.0 incorporates all the user-friendly features of the previous versions and also handles the simulation of geometry, traffic, and traffic control (including actuated controllers) for most conventional diamond interchanges. In addition to running on IBM (and compatible) mainframe and microcomputers, Version 3.0 has been adapted to run in the UNIX environment in INTERGRAPH Corporation's Clipper workstations. A unique, high-quality animated-graphics screen display is available on this platform. It features simultaneous, synchronized viewing and manipulation of the animated-graphics from four runs of the TEXAS Model. Research Project 0-1258 added the following features: handling separate U-turn lanes at diamond interchanges (ramp-to-ramp connecting roadways), generating exact percentage of desired driver-vehicle units, implementing sight-distance checking in the user-friendly version (already available in mainframe versions), simulating NEMA dual-ring traffic signal controllers, simulating volume-density traffic signal controllers, providing user choice between "City of Dallas" and "TxDOT" (Texas Department of Transportation) numbering scheme for traffic phases at diamond interchanges, presenting output summary statistics in graphical form (spreadsheet capabilities to selected types of plotters), developing generic plotter-driver output routines and interface capabilities to selected types of plotters, and automating the required number of replicate runs to achieve stability in selected measures of effectiveness [7]. Research Project 0-1258 created Version 3.12 in January of 1993. Several modifications to the TEXAS Model were made in November of 1993 and released as Version 3.20, followed by Version 3.21 in May of 1994, then Version 3.22 in June of 1994, then in February of 1996 Version 3.23, which modified the car-following logic, and finally in September of 1998 Version 3.24, which fixed the Year 2000 Bug until 2090. During FHWA Project DTFH61-01-Q-00166, Dr. Rioux (1) analyzed how to simulate intersection collision avoidance with the TEXAS Model, (2) converted the TEXAS Model to compile and run on Absoft FORTRAN 1995 compilers on the Windows and Linux operating systems, (3) converted the TEXAS Model Animation Processor to the latest version of Java for the Windows and Linux operating environments, and (4) estimated the resources required to complete each of the design requirements.

The TEXAS Model was originally written to serve as a single intersection analysis tool. It was designed to be a companion tool to NETSIM. NETSIM was to serve as a multiple intersection analysis tool. As such, the TEXAS Model was designed to be a path-based, ultradetailed model. These features are what make it suitable for a base for intersection collision avoidance simulation.

Dr. Clyde E. Lee was the supervising professor for all research performed for the TEXAS Model until his retirement. Dr. Thomas W. Rioux was the leader of the team of graduate students that developed the computer programs, developed the Geometry Processor as his thesis, developed the Simulation Processor with the assistance of Mr. Charlie Copeland and other graduate students as his dissertation, supervised the development of the Driver-Vehicle Processor by Mr. Charlie Copeland, and participated in many enhancement projects. Mr. Robert F. Inman was the leader of the team that developed the "user friendly" input processors and the PC animation. Dr. Randy B. Machemehl supervised numerous graduate students who have used the TEXAS Model in their research and assisted in several enhancements since the initial development of the TEXAS Model.

This project consisted of six tasks. The first four were performed sequentially, with a demonstration of successful execution prior to initiation of the subsequent task. The fifth task, modification of data input programs and user guides took place concurrently with the first four tasks. The tasks were:

1. Modify the driver/vehicle stop/go behavior.
2. Modify user-specifiable time step from 0.1 second to 0.01 second.
3. Model dynamic message signs.
4. Model enhanced Intelligent Transportation Systems (ITS) sensors,
5. Modify input and output data programs to handle new data features.
6. Prepare final report and deliverables.

Throughout this document, the following definitions apply:

- The term Vehicle Message System (VMS) refers to the software logic used in the TEXAS Model for controlling driver and vehicular behaviors. One use of VMS is to simulate Dynamic Message Signs (DMS) while another use is to simulate In-Vehicle Driver Messaging Systems (IVDMS). The third use is to induce driver behaviors which can simulate Illegal Behavior (IB) such as red light running. The term does NOT mean Variable Message Signs, an obsolete term for Dynamic Message Signs.
- The term DT (Delta Time) refers to the value of the time step increment for the simulation in seconds and is a real number.
- The term A**B means A to the power of B.

CHAPTER 2. MODIFY THE DRIVER/VEHICLE STOP/GO BEHAVIOR

Task 1 modified the driver's behavior so that the driver can be made to stop, go, or change lanes when it would be otherwise illogical to do so. This allows modeling of the driver's illogical behavior, which leads to accidents in the "real world." This task provided the hooks for the Vehicle Messaging System (VMS) to influence the driver's behavior. During Task 1 new versions of the Absoft FORTRAN compilers were purchased and installed on the Intel/Windows and Intel/Linux computers. Task 1 consisted of three sub-tasks.

Task 1.1 modified the driver's behavior to force a stop by a control system or user-set flags. This behavior is required in order to initiate responses to simulated DMS, IVDMS, or IB messages telling the vehicle to stop, whether the DMS is on roadway or the IVDMS is in the vehicle. Setting desired speeds, lane changes, and stopping locations allows simulation of vehicles stopping in the middle of intersections to wait for gaps and speed control systems to minimize red light running and traffic turbulence.

To implement this feature, additional independent variables were added to the diagram and computer code was developed to implement the changes. See Figure 5.14 (Logical binary network for acceleration and deceleration responses) on page 189 of "The TEXAS Model for Intersection Traffic - Development," by Lee, Rioux, and Copeland, December 1977, and Figure 1 (Modified logical binary network for acceleration and deceleration responses).

Task 1.2 modified the driver's red light response behavior by modifying the routines that process the driver's response to a signal indication to determine whether the movement is green, yellow, red, or protected green. See Figure 5.15 (Logical binary network for intersection entry control) on page 198 of "The TEXAS model for Intersection Traffic - Development," by Lee, Rioux, and Copeland, December 1977, and Figure 2 (Logical binary network for intersection entry control). Previously, for the first DT when the signal changes from green to yellow, the driver determines whether there is adequate stopping distance to stop at the stop line. If it is farther from the stop line than the stopping distance, then it will stop on yellow; otherwise it will go on yellow. Additional parameters were added to force vehicle to go on yellow when otherwise unwarranted, thus running the red signal.

Task 1.3 demonstrated collisions for each of the five major scenarios. The major purpose of Task 1 was to enable simulation of crashes in the TEXAS Model. Therefore, the demonstrations that were prepared for FHWA showed each of the five major intersection crash scenarios - Left Turn Across Path/Opposite Direction (LTAP/OD), Left Turn Across Path/Lateral Direction (LTAP/LD), Left Turn Into Path/Merge (LTIP), Right Turn Into Path/Merge (RTIP), and Straight Crossing Paths (SCP) - which are to be handled by intersection collision avoidance systems. (Note: Right Turn Across Path/Lateral Direction (RTAP/LD) was not simulated.)

Figure 5.14 from CTR Research Report 184-1 [(1)] was modified to add the questions "Am I forced to stop?" and "Am I forced to go?" and their responses.

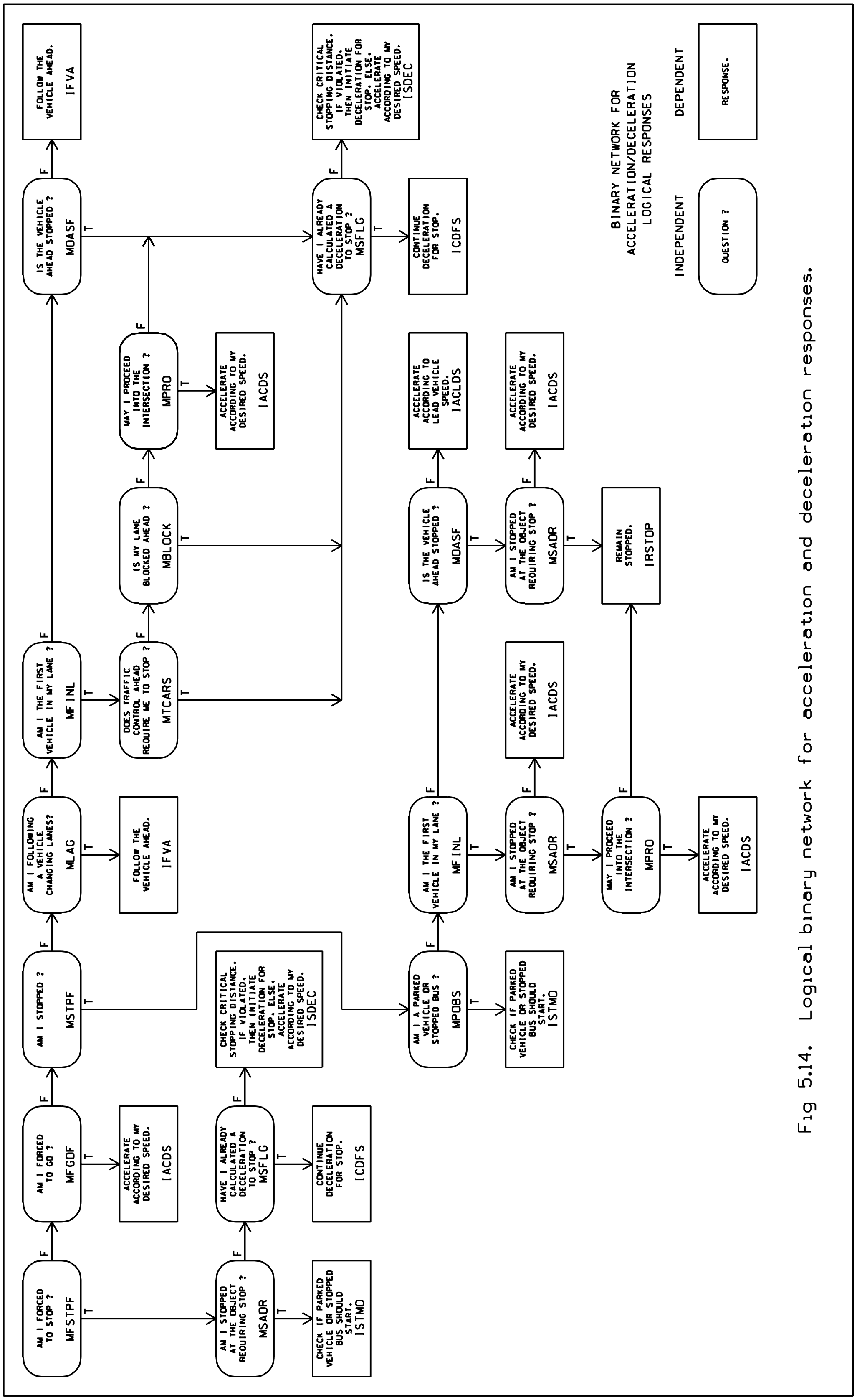


Figure 1. Modified logical binary network for acceleration and deceleration responses.

In Figure 1 all true branches go down the page from top to bottom and all false branches go across the page from left to right. The variable name after the question or response is the variable interrogated for the question or the variable set true for the response. There are some duplicate questions and responses. Only one path through the logical binary network is true and all other paths are false. Since a logical "or" is used when there are multiple paths to the same response, one path can be true and others false to produce a true response. Starting at the beginning question in the upper left corner of the figure (Q1) "Am I forced to stop? MFSTPF," the true branch goes to (Q7) "Am I stopped at the object requiring stop? MSAOR" and the false branch goes to (Q2) "Am I forced to go? MFGOF". For (Q2) "Am I forced to go? MFGOF," the true branch goes to (R2) "Accelerate according to my desired speed. IACDS" and the false branch goes to (Q3) "Am I stopped? MSTPF". For (Q3) "Am I stopped? MSTPF," the true branch goes to (Q13) "Am I a parked vehicle or stopped bus? MPOBS" and the false branch goes to (Q4) "Am I following a vehicle changing lanes? MLAG". For (Q4) "Am I following a vehicle changing lanes? MLAG," the true branch goes to (R3) "Follow the vehicle ahead. IFVA" and the false branch goes to (Q5) "Am I the first vehicle in my lane? MFINL". For (Q5) "Am I the first vehicle in my lane? MFINL," the true branch goes to (Q9) "Does traffic control ahead require me to stop? MTCARS" and the false branch goes to (Q6) "Is the vehicle ahead stopped? MOASF". For (Q6) "Is the vehicle ahead stopped? MOASF," the true branch goes to (Q12) "Have I already calculated a deceleration to stop? MSFLG" and the false branch goes to (R1) "Follow the vehicle ahead. IFVA". For (Q7) "Am I stopped at the object requiring stop? MSAOR," the true branch goes to (R6) "Check if parked vehicle or stopped bus should start. ISTMO" and the false branch goes to (Q8) "Have I already calculated a deceleration to stop? MSFLG". For (Q8) "Have I already calculated a deceleration to stop? MSFLG," the true branch goes to (R7) "Continue deceleration for stop. ICDFS" and the false branch goes to (R4) "Check critical stopping distance. If violated, then initiate deceleration for stop, else, accelerate according to my desired speed. ISDEC". For (Q9) "Does traffic control ahead require me to stop? MTCARS," the true branch goes to (Q12) "Have I already calculated a deceleration to stop? MSFLG" and the false branch goes to (Q10) "Is my lane blocked ahead? MBLOCK". For (Q10) "Is my lane blocked ahead? MBLOCK," the true branch goes to (Q12) "Have I already calculated a deceleration to stop? MSFLG" and the false branch goes to (Q11) "May I proceed into the intersection? MPRO". For (Q11) "May I proceed into the intersection? MPRO," the true branch goes to (R5) "Accelerate according to my desired speed. IACDS" and the false branch goes to (Q12) "Have I already calculated a deceleration to stop? MSFLG". For (Q12) "Have I already calculated a deceleration to stop? MSFLG," the true branch goes to (R10) "Continue deceleration for stop. ICDFS" and the false branch goes to (R8) "Check critical stopping distance. If violated, then initiate deceleration for stop, else, accelerate according to my desired speed. ISDEC". For (Q13) "Am I a parked vehicle or stopped bus? MPOBS," the true branch goes to (R11) "Check if parked vehicle or stopped bus should start. ISTMO" and the false branch goes to (Q14) "Am I the first vehicle in my lane? MFINL". For (Q14) "Am I the first vehicle in my lane? MFINL," the true branch goes to (Q16) "Am I stopped at the object requiring stop? MSAOR" and the false branch goes to (Q15) "Is the vehicle ahead stopped? MOASF". For (Q15) "Is the vehicle ahead stopped? MOASF," the true branch goes to (Q17) "Am I stopped at the object requiring stop? MSAOR" and the false branch goes to (R9) "Accelerate according to lead vehicle speed. IACLDS". For (Q16) "Am I stopped at the object requiring stop? MSAOR," the true branch goes to (Q18) "May I proceed into the intersection? MPRO" and the false branch goes to (R12) "Accelerate according to my desired speed. IACDS".

For (Q17) “Am I stopped at the object requiring stop? MSAOR,” the true branch goes to (R14) “Remain stopped. IRSTOP” and the false branch goes to (R13) “Accelerate according to my desired speed. IACDS”. For (Q18) “May I proceed into the intersection? MPRO,” the true branch goes to (R15) “Accelerate according to my desired speed. IACDS” and the false branch goes to (R14) “Remain stopped. IRSTOP”.

Intersection collision avoidance logic could set MFSTPF true to force a vehicle to stop and could set MFGOF to true to force a vehicle to go. The code that implements ISDEC was modified to set the required stopping location if MFSTPF is true. The code that implements IACDS was modified to set the required desired speed if MFGOF is true; the value could be greater than or less than the current speed.

Figure 5.15 from CTR Research Report 184-1 [1] was not modified but is presented here for clarity.

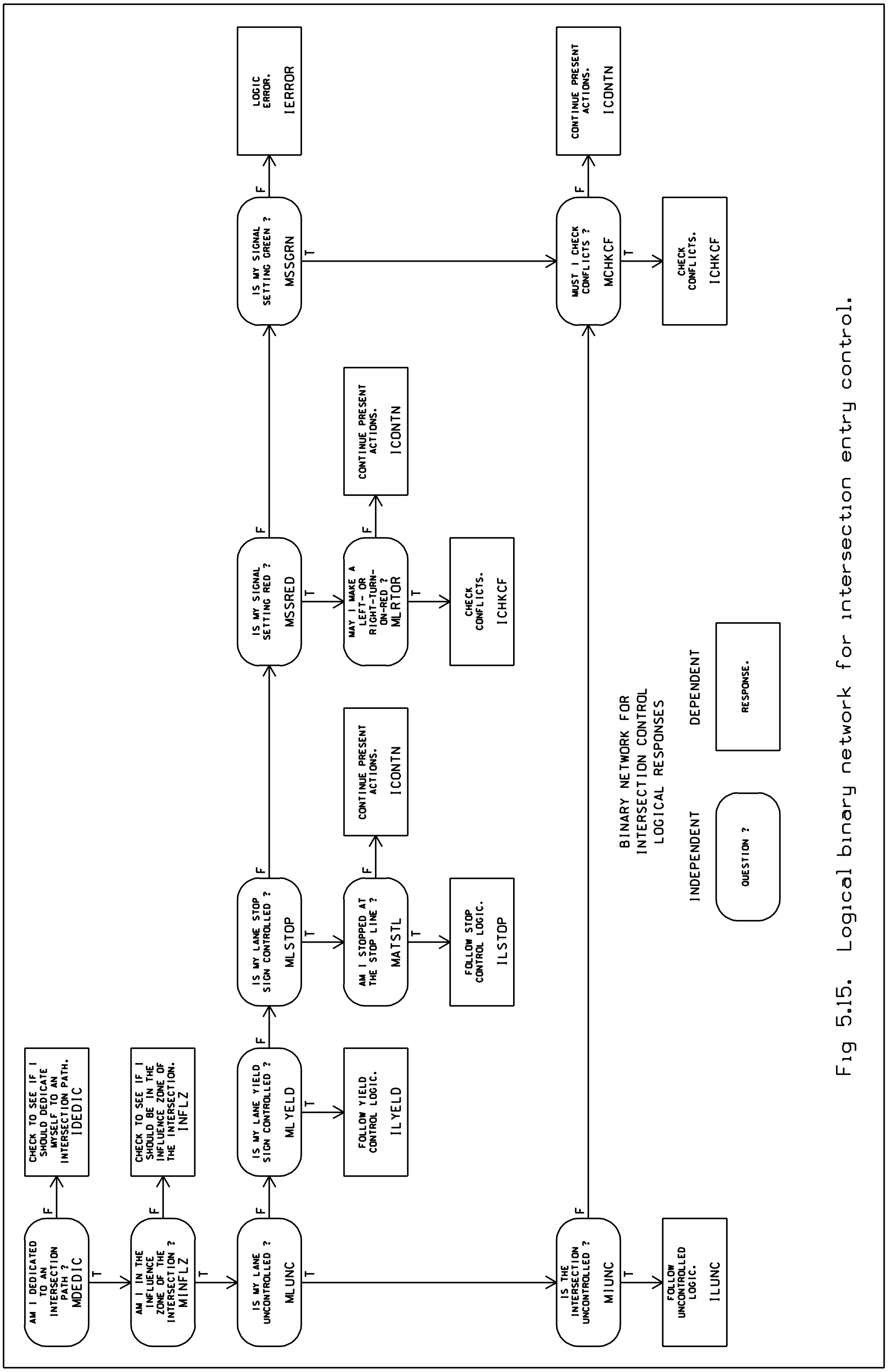


Figure 2. Logical binary network for intersection entry control.

In Figure 2 all true branches go down the page from top to bottom, and all false branches go across the page from left to right. The variable name after the question or response is the variable interrogated for the question or the variable set true for the response. There are some duplicate questions and responses. Only one path through the logical binary network is true and all other paths are false. Since a logical "or" is used when there are multiple paths to the same response, one path can be true and others false to produce a true response. Starting at the beginning question in the upper left corner of the figure (Q1) "Am I dedicated to an intersection path? MDEDIC," the true branch goes to (Q2) "Am I in the influence zone of the intersection? MINFLZ" and the false branch goes to (R1) "Check to see if I should dedicate myself to an intersection path. IDEDIC". For (Q2) "Am I in the influence zone of the intersection? MINFLZ," the true branch goes to (Q3) "Is my lane uncontrolled? MLUNC" and the false branch goes to (R2) "Check to see if I should be in the influence zone of the intersection. INFLZ". For (Q3) "Is my lane uncontrolled? MLUNC," the true branch goes to (Q10) "Is the intersection uncontrolled? MIUNC" and the false branch goes to (Q4) "Is my lane yield sign controlled? MLYELD". For (Q4) "Is my lane yield sign controlled? MLYELD," the true branch goes to (R4) "Follow yield control logic. ILYELD" and the false branch goes to (Q5) "Is my lane stop sign controlled? MLSTOP". For (Q5) "Is my lane stop sign controlled? MLSTOP," the true branch goes to (Q8) "Am I stopped at the stop line? MATSTL" and the false branch goes to (Q6) "Is my signal setting red? MSSRED". For (Q6) "Is my signal setting red? MSSRED," the true branch goes to (Q9) "May I make a left-turn or right-turn-on-red? MLRTOR" and the false branch goes to (Q7) "Is my signal setting green? MSSGRN". For (Q7) "Is my signal setting green? MSSGRN," the true branch goes to (Q11) "Must I check conflicts? MCHKCF" and the false branch goes to (R3) "Logic error. IERROR". For (Q8) "Am I stopped at the stop line? MATSTL," the true branch goes to (R7) "Follow stop control logic. ILSTOP" and the false branch goes to (R5) "Continue present actions. ICONTN". For (Q9) "May I make a left-turn or right-turn-on-red? MLRTOR," the true branch goes to (R8) "Check conflicts. ICHKCF" and the false branch goes to (R6) "Continue present actions. ICONTN". For (Q10) "Is the intersection uncontrolled? MIUNC," the true branch goes to (R10) "Follow uncontrolled logic. ILUNC" and the false branch goes to (Q11) "Must I check for conflicts? MCHKCF". For (Q11) "Must I check for conflicts? MCHKCF," the true branch goes to (R11) "Check conflicts. ICHKCF" and the false branch goes to (R9) "Continue present actions. ICONTN".

This feature is controlled by adding additional input parameters for a special vehicle. The current version of GDVDATA does not have the capability of entering special vehicles, but GDVPRO does have this capability. GDVDATA was NOT modified to allow entry of special vehicles. GDVCONV, GDVPRO, and SIMPRO were modified to process the additional input parameters for a special vehicle.

For all subroutines and COMMON blocks, all logical dependent and logical independent variables were changed from type INTEGER to type LOGICAL. Additionally, all variables using the variable LTRUE and LFALSE were changed from type INTEGER to type LOGICAL.

The following variables were added to SIMPRO for all vehicles (IV is the vehicle index):

FGOATM(IV)	Real	forced go active time (zero for no forced go); in file VEHD, COMMON block VEHDR

FGOTIM(IV)	Real	forced go time (zero for no forced go); in file VEHD, COMMON block VEHDR
FRRATM(IV)	Real	forced run red signal active time (zero for no forced run red signal); in file VEHD, COMMON block VEHDR
FRRTIM(IV)	Real	forced run red signal time (zero for no forced run red signal); in file VEHD, COMMON block VEHDR
FSTDTM(IV)	Real	forced stop dwell time (zero for no forced stop dwell); in file VEHD, COMMON block VEHDR
FSTPIA(IV)	Integer	approach number (positive) or intersection path number (negative) for forced stop (zero for no forced stop); in file VEHD; COMMON blockVEHD2
FSTPOS(IV)	Real	forced stop position on approach or intersection path (zero for no forced stop); in file VEHD, COMMON block VEHDR
FSTTIM(IV)	Real	forced stop time (zero for no forced stop); in file VEHD, COMMON block VEHDR
GACTIM(IV)	Integer	forced go active time counter (number of DTs); in file VEHD; COMMON blockVEHD2
MFGOF(IV)	Logical	forced go flag (true/false); in file VEHD, COMMON block VEHDL
MFSTPF(IV)	Logical	forced stop flag (true/false); in file VEHD, COMMON block VEHDL
SDWELL(IV)	Integer	forced stop dwell time counter (number of DTs); in file VEHD; COMMON blockVEHD2

The following variables were added to SIMPRO for vehicles read from the driver-vehicle processor input file and awaiting logging into the system (listed in the order read) (IB is the queue-in buffer index):

QSTTIM(IB)	Real	forced stop time (hundredths of a second) (zero for no forced stop); stored in FSTTIM(IV) for the vehicle; in file QUE, COMMON block QUER
IBUF(IB,9)	Integer	approach number (positive) or intersection path number (negative) for forced stop (zero for no forced stop); stored in FSTPIA(IV) for the vehicle; in file QUE, COMMON block QUE2
QSTPOS(IB)	Real	forced stop position on approach or intersection path (zero for no forced stop); stored in FSTPOS(IV) for the vehicle; in file QUE, COMMON block QUER
QSTDTM(IB)	Real	forced stop dwell time (hundredths of a second) (zero for no forced stop dwell); stored in FSTDTM(IV) for the vehicle; in file QUE, COMMON block QUER
QGOTIM(IB)	Real	forced go time (hundredths of a second) (zero for no forced go); stored in FGOTIM(IV) for the vehicle; in file QUE, COMMON block QUER

QGOATM(IB)	Real	forced go active time (hundredths of a second) (zero for no forced go); stored in FGOATM(IV) for the vehicle; in file QUE, COMMON block QUER
QRRTIM(IB)	Real	forced run red signal time (hundredths of a second) (zero for no forced run red signal); stored in FRRTIM(IV) for the vehicle; in file QUE, COMMON block QUER
QRRATM(IB)	Real	forced run red signal active time (hundredths of a second) (zero for no forced run red signal); stored in FRRATM(IV) for the vehicle; in file QUE, COMMON block QUER

For reading data for a vehicle from the driver-vehicle processor input file, in subroutine RDVPRD and in subroutine LOGIN, the format was extended as follows:

Field	Description	Start Col	Num Cols	Data Type	Variable Name	Units
1	Queue-in Time	1	6	Integer	QTIME(IB)	Hundredths of a second
2	Vehicle Class	7	2	Integer	IBUF(IB,1)	None
3	Driver Class	9	1	Integer	IBUF(IB,2)	None
4	Desired Velocity	10	3	Integer	IBUF(IB,3)	Feet per second
5	Desired Outbound Leg	13	2	Integer	IBUF(IB,4)	None
6	Inbound Leg	15	2	Integer	IBUF(IB,5)	None
7	Inbound Lane	17	1	Integer	IBUF(IB,6)	1=median & N=curb
8	Print on Logout	18	1	Integer	IBUF(IB,7)	O=no & 1=yes
9	Free U-Turn Option	19	1	Character	IFUT	F=use free U-turn
10	Forced Stop Time	20	6	Integer	QSTTIM(IB)	Hundredths of a second
11	Forced Stop Leg/Path	26	4	Integer	IBUF(IB,9)	None; +=leg & -=path
12	Forced Stop Position	30	7	Real	QSTPOS(IB)	Feet on leg/path
13	Forced Stop Dwell Time	37	6	Integer	QSTDTM(IB)	Hundredths of a second
14	Forced Go Time	43	6	Integer	QGOTIM(IB)	Hundredths of a second
15	Forced Go Active Time	49	6	Integer	QGOATM(IB)	Hundredths of a second
16	Forced Run Red Signal Time	55	6	Integer	QRRTIM(IB)	Hundredths of a second
17	Forced Run Red Signal Active Time	61	6	Integer	QRRATM(IB)	Hundredths of a second

When the vehicle logs into the system, (1) the forced stop flag is set false, (2) the forced go flag is set false, (3) the forced stop dwell time counter is set to zero, (4) the forced go active time counter is set to zero, and (5) the forced stop time, forced stop leg/path, forced stop position, forced stop dwell time, forced go time, forced go active time, the forced run red signal time, and the forced run red signal active time are set to the values read from input for the vehicle.

The following describes the general logic to implement a forced stop. When the time to initiate a forced stop is reached and the vehicle is on the specified leg or intersection path, (1) the forced stop flag is set true, (2) the forced go flag is set false, (3) the forced go active time counter is set to zero, (4) the previous vehicle's position is set to the specified stopping position, and (5) the previous vehicle's velocity, acceleration, and jerk are set to zero. As the vehicle travels down the lane, (1) the vehicle is made to stop at the specified position using the normal logic that stops a vehicle behind another stopped vehicle (the previous vehicle) and (2) any normal previous vehicle position that is closer to the forced stop vehicle than the forced stop position is used, so a collision with another real vehicle can be avoided. When the vehicle comes to a stop at the specified position, the forced stop dwell time counter is set. The forced stop dwell time counter is decremented by one each DT. When the forced stop dwell time counter is zero, the forced stop flag is set false and the previous vehicle's position, velocity, acceleration, and jerk are set to usual values so that the vehicle can continue normally.

The following describes the general logic to implement a forced go. When the time to initiate a forced go is reached, (1) the forced go flag is set true, (2) the forced go active time counter is set, (3) the forced stop flag is set false, (4) the forced stop dwell time counter is set to zero, and (5) the previous vehicle's position, velocity, acceleration, and jerk are set to usual values. The forced go active time counter is decremented by one each DT. When the forced go active time counter is zero, the forced go flag is set false, the forced stop flag is set false, and the forced stop dwell time counter is set to zero. While the vehicle's forced go flag is true (these same actions are performed for the forced run red signal), (1) the vehicle does not check any sight distance restrictions, (2) the vehicle does not check any intersection conflicts, (3) all other vehicles ignore the vehicle when checking intersection conflicts, (4) the vehicle does not try to avoid any intersection conflicts, (5) all other vehicles ignore the vehicle when trying to avoid intersection conflicts, and (6) the vehicle will respond to all signal indications as if they were a protected green.

The following describes the general logic to implement a forced run red signal. While the simulation time is greater than or equal to the vehicle's forced run red signal time and less than or equal to the vehicle's forced run red signal time plus forced run red signal active time (these same actions are performed for the forced go), (1) the vehicle does not check any sight distance restrictions, (2) the vehicle does not check any intersection conflicts, (3) all other vehicles ignore the vehicle when checking intersection conflicts, (4) the vehicle does not try to avoid any intersection conflicts, (5) all other vehicles ignore the vehicle when trying to avoid intersection conflicts, and (6) the vehicle will respond to all signal indications as if they were a protected green.

For each vehicle being logged into the system, in subroutine LOGIN, set GACTIM(IV) to zero, SDWELL(IV) to zero, FGOATM(IV) to QGOATM(IB), FGOTIM(IV) to QGOTIM(IB), FRRATM(IV) to QRRATM(IB), FRRTIM(IV) to QRRTIM(IB), FSTDTM(IV) to QSTDTM(IB), FSTPIA(IV) to IBUF(IB,9), FSTPOS(IV) to QSTPOS(IB), FSTTIM(IV) to QSTTIM(IB), MFGOF(IV) to false, and MFSTPF(IV) to false.

For all vehicles, in subroutine LOGIC, for the acceleration/deceleration logic, the equations for the dependant variables were modified to use logical variables.

For a forced stop, in subroutine PREST1, if the forced stop time FSTTIM(IV) is greater than zero, the forced stop time FSTTIM(IV) is greater than the simulation time at the start of the DT (TIME-DT), and the forced stop time FSTTIM(IV) is less than or equal to the simulation time TIME, then (a) if the vehicle is in the intersection (MININT(IV) is true) and the current intersection path LPRES(IV) is equal to the intersection path for a forced stop FSTPIA(IV), then (1) set the forced stop flag MFSTPF(IV) to true, (2) set the forced go flag MFGOF(IV) to false, (3) set the forced go active time timer GACTIM(IV) to zero, (4) set the previous vehicle's position PVPOS to the forced stop position on the intersection path FSTPOS(IV), (5) set the previous vehicle's velocity PVVEL to zero, (6) set the previous vehicle's acceleration PVACC to zero, and (7) set the previous vehicle's jerk rate PVSLP to zero else (b) if the vehicle is not in the intersection (i.e., MININT(IV) is false) and the current approach number ISNA(LPRES(IV)) is equal to the approach for a forced stop FSTPIA(IV), then (1) set the forced stop flag MFSTPF(IV) to true, (2) set the forced go flag MFGOF(IV) to false, (3) set the forced go active time timer GACTIM(IV) to zero, (4) set the previous vehicle's position PVPOS to the forced stop position on the intersection path FSTPOS(IV), (5) set the previous vehicle's velocity PVVEL to zero, (6) set the previous vehicle's acceleration PVACC to zero, and (7) set the previous vehicle's jerk rate PVSLP to zero. Subroutine PREST1 is called by IBAP, INTERP, and OBAP for each vehicle as one of the first things when processing a vehicle.

For a forced stop, in virtually all subroutines that set PVPOS, if the forced stop flag MFSTPF(IV) is true and the forced stop position FSTPOS(IV) is less than the previous vehicle's position PVPOS, then set the previous vehicle's position PVPOS to the forced stop position FSTPOS(IV).

For a forced stop, in subroutine ACDCP when processing the acceleration/deceleration logic values, if the forced stop flag MFSTPF(IV) is true, then (a) if the forced stop dwell time counter SDWELL(IV) is greater than zero, then decrement the forced stop dwell time counter SDWELL(IV) by one and (b) if the forced stop dwell time counter SDWELL(IV) is equal to zero, then (1) set the vehicle intersection control logic timer LOGTMP and LOGFLG(IV) to one to cause a re-calculation of intersection control logic; (2) set the forced stop flag MFSTPF(IV) to false; (3) set the vehicle stopped at object requiring stop flag MSAOR(IV) to false; (4) if the vehicle is in the intersection (MININT(IV) is true), then (i) set the previous vehicle's number IVPV to zero, (ii) set the previous vehicle's position PVPOS to the end of the lane/path ENDLN), (iii) set the previous vehicle's velocity PVVEL to the speed limit for the intersection path LIMP(IP), (iv) set the previous vehicle's acceleration PVACC to zero, and (v) set the previous vehicle's jerk rate PVSLP to zero; (5) if the vehicle is not in the intersection (MININT(IV) is false), then (i) set the previous vehicle's number IVPV to zero, (ii) set the previous vehicle's position PVPOS to the end of the lane/path ENDLN), (iii) set the previous vehicle's velocity PVVEL to the speed limit for the approach ISLIM(IA), (iv) set the previous vehicle's acceleration PVACC to zero, and (v) set the previous vehicle's jerk rate PVSLP to zero; (6) if there is a vehicle ahead in the same lane, then set the previous vehicle's parameters IVPV, PVPOS, PVVEL, PVACC, and PVSLP to that vehicle; (7) if the lane is blocked (MBLOCK(IV) is true) and the previous vehicle's position PVPOS is greater than the end of the lane ENDLN, then set the vehicle first in lane flag MFINL(IV) to true; and (8) if the vehicle is first in lane (MFINL(IV) is true) and the lane is blocked (MBLOCK(IV) is true), then set the previous vehicle's parameters IVPV, PVPOS, PVVEL, PVACC, and PVSLP for the end of the blocked lane.

For a forced stop, in subroutine ACDCP when the vehicle stops at the object requiring a stop, if the forced stop flag MFSTPF(IV) is true and the forced stop dwell time counter SDWELL(IV) is equal to zero, then (a) set the forced stop dwell time counter SDWELL(IV) to the forced stop dwell time FSTDTM(IV) divided by DT and round the result to the nearest integer, (b) if the forced stop dwell time counter SDWELL(IV) is equal to zero, then (1) set the vehicle intersection control logic timer LOGTMP and LOGFLG(IV) to one to cause a re-calculation of intersection control logic; (2) set the forced stop flag MFSTPF(IV) to false; (3) set the vehicle stopped at object requiring stop flag MSAOR(IV) to false; (4) if the vehicle is in the intersection (MININT(IV) is true), then (i) set the previous vehicle's number IVPV to zero, (ii) set the previous vehicle's position PVPOS to the end of the lane/path ENDLN), (iii) set the previous vehicle's velocity PVVEL to the speed limit for the intersection path LIMP(IP), (iv) set the previous vehicle's acceleration PVACC to zero, and (v) set the previous vehicle's jerk rate PVSLP to zero; (5) if the vehicle is not in the intersection (MININT(IV) is false), then (i) set the previous vehicle's number IVPV to zero, (ii) set the previous vehicle's position PVPOS to the end of the lane/path ENDLN), (iii) set the previous vehicle's velocity PVVEL to the speed limit for the approach ISLIM(IA), (iv) set the previous vehicle's acceleration PVACC to zero, and (v) set the previous vehicle's jerk rate PVSLP to zero; (6) if there is a vehicle ahead in the same lane, then set the previous vehicle's parameters IVPV, PVPOS, PVVEL, PVACC, and PVSLP to that vehicle; (7) if the lane is blocked (MBLOCK(IV) is true) and the previous vehicle's position PVPOS is greater than the end of the lane ENDLN, then set the vehicle first in lane flag MFINL(IV) to true; and (8) if the vehicle is first in lane (MFINL(IV) is true) and the lane is blocked (MBLOCK(IV) is true); then set the previous vehicle's parameters IVPV, PVPOS, PVVEL, PVACC, and PVSLP for the end of the blocked lane.

For a forced go or forced run red signal, in subroutine AVDCON, if the forced go flag MFGOF(IV) is true or the forced run the red signal time FRRTIM(IV) is greater than zero and the simulation time TIME is greater than or equal to the forced run the red signal time FRRTIM(IV) and the simulation time TIME is less than or equal to the forced run red signal time FRRTIM(IV) plus the forced run red signal active time FRRATM(IV), then do not try to avoid any intersection conflicts (proceed normally). While checking other vehicles, if the forced go flag MFGOF(IVCONF) is true or the forced run the red signal time FRRTIM(IVCONF) is greater than zero and the simulation time at the start of the DT (TIME-DT) is greater than the forced run the red signal time FRRTIM(IVCONV) and the simulation time TIME is less than or equal to the forced run red signal time FRRTIM(IVCONF) plus the forced run red signal active time FRRATM(IVCONF), then do not try to avoid an intersection conflict with this vehicle (ignore the IVCONF vehicle).

For a forced go or forced run red signal, in subroutine IBAP, if the forced go flag MFGOF(IV) is true or the forced run the red signal time FRRTIM(IV) is greater than zero and the simulation time TIME is greater than or equal to the forced run the red signal time FRRTIM(IV) and the simulation time TIME is less than or equal to the forced run red signal time FRRTIM(IV) plus the forced run red signal active time FRRATM(IV), then set the signal indication KSISET to the current cam stack position and inbound lane number ISISET(ICAMPC,IBLN(IL)) (the signal indication needs to be processed each DT, not just when the signal indication changes).

For a forced go or forced run red signal, in subroutine SIGRES, if the forced go flag MFGOF(IV) is true or the forced run the red signal time FRRTIM(IV) is greater than zero and the simulation time TIME is greater than or equal to the forced run the red signal time FRRTIM(IV) and the simulation time TIME is less than or equal to the forced run red signal time FRRTIM(IV) plus the forced run red signal active time FRRATM(IV), then set the signal indication IGARP to four (protected green) and respond accordingly (disregard the actual indication).

For a forced go or forced run red signal, in subroutine CHKSDR, if the forced go flag MFGOF(IV) is true or the forced run the red signal time FRRTIM(IV) is greater than zero and the simulation time TIME is greater than or equal to the forced run the red signal time FRRTIM(IV) and the simulation time TIME is less than or equal to the forced run red signal time FRRTIM(IV) plus the forced run red signal active time FRRATM(IV), then ignore any sight distance restrictions.

For a forced go or forced run red signal, in subroutine CHKCON, if the forced go flag MFGOF(IV) is true or the forced run the red signal time FRRTIM(IV) is greater than zero and the simulation time TIME is greater than or equal to the forced run the red signal time FRRTIM(IV) and the simulation time TIME is less than or equal to the forced run red signal time FRRTIM(IV) plus the forced run red signal active time FRRATM(IV), then ignore any intersection conflicts and set that the vehicle may proceed into the intersection. While checking other vehicles, if the forced go flag MFGOF(IVCONF) is true or the forced run the red signal time FRRTIM(IVCONF) is greater than zero and the simulation time at the start of the DT (TIME-DT) is greater than the forced run the red signal time FRRTIM(IVCONV) and the simulation time TIME is less than or equal to the forced run red signal time FRRTIM(IV) plus the forced run red signal active time FRRATM(IV), then ignore any intersection conflict with this vehicle (ignore the IVCONF vehicle).

For a forced go, in subroutine PREST1, if the forced go flag MFGOF(IV) is true, then (1) if the forced go active time counter GACTIM(IV) is greater than zero, then decrement the forced go active time counter GACTIM(IV) by one and (2) if the forced go active time counter GACTIM(IV) is equal to zero, then (1) set the vehicle intersection control logic timer LOGTMP and LOGFLG(IV) to one to cause a re-calculation of intersection control logic, (2) set the forced go flag MFGOF(IV) to false, (3) set the forced stop flag MFSTPF(IV) to false, and (4) set the vehicle stopped dwell timer SDWELL(IV) to zero.

For a forced go, in subroutine PREST1, if the forced go time FGOTIM(IV) is greater than zero and the forced go time FGOTIM(IV) is greater than the simulation time at the start of the DT (TIME-DT) and the forced go time FGOTIM(IV) is less than or equal to the simulation time TIME, then (1) set the vehicle intersection control logic timer LOGTMP and LOGFLG(IV) to one to cause a re-calculation of intersection control logic; (2) set the forced go flag MFGOF(IV) to true; (3) set the forced go active time counter GACTIM(IV) to the forced go active time FGOATM(IV) divided by DT and round the result to the nearest integer; (4) set the forced stop flag MFSTPF(IV) to false; (5) set the forced stop dwell time counter SDWELL(IV) to zero; (6) set the vehicle stopped at object requiring stop flag MSAOR(IV) to false; (7) if the vehicle is in the intersection (MININT(IV) is true), then (i) set the previous vehicle's number IVPV to zero, (ii) set the previous vehicle's position PVPOS to the end of the lane/path ENDLN), (iii) set the previous vehicle's velocity PVVEL to the speed limit for the intersection path LIMP(IP), (iv) set the previous vehicle's acceleration PVACC to zero, and (v) set the previous vehicle's jerk rate PVSLP to zero; (8) if the vehicle is not in the intersection (MININT(IV) is false), then (i) set the previous vehicle's number IVPV to zero, (ii) set the previous vehicle's position PVPOS to the end of the lane/path ENDLN), (iii) set the previous vehicle's velocity PVVEL to the speed limit for the approach ISLIM(IA), (iv) set the previous vehicle's acceleration PVACC to zero, and (v) set the previous vehicle's jerk rate PVSLP to zero; (9) if there is a vehicle ahead in the same lane, then set the previous vehicle's parameters IVPV, PVPOS, PVVEL, PVACC, and PVSLP to that vehicle; (10) if the lane is blocked (MBLOCK(IV) is true) and the previous vehicle's position PVPOS is greater than the end of the lane ENDLN, then set the vehicle first in lane flag MFINL(IV) to true; and (11) if the vehicle is first in lane (MFINL(IV) is true) and the lane is blocked (MBLOCK(IV) is true), then set the previous vehicle's parameters IVPV, PVPOS, PVVEL, PVACC, and PVSLP for the end of the blocked lane.

This page intentionally left blank to facilitate printing on 2 sides.

CHAPTER 3. MODIFY USER-SPECIFIABLE TIME STEP FROM 0.1 SECOND TO 0.01 SECOND

A user-specifiable high-resolution time step is required to accurately portray the interaction of the traffic control system, the driver, and the vehicle.

Task 2 modified the variables used to store position, velocity, acceleration, jerk, and other variables related to vehicle movement to double precision and eliminate factoring to allow accurate calculations at a time step (DT) of 0.01 seconds. This required going through the code systematically to identify all logic using these variables and modifying it as appropriate to handle the higher-resolution variables.

Note that although the TEXAS Model will then run at a DT of 0.01 second, the timing fields of traffic signal controllers are specified at 0.1 seconds. Since real world signal controllers use this value for data entry, there is no need to change the data entry values for signal timing. The TEXAS Model traffic signal controller logic is executed once per DT.

SIMCONV and SIMPRO were modified to read and process this new minimum value for DT.

The following COMMON BLOCK variables were changed from INTEGER or REAL to DOUBLE PRECISION:

Variable Name	Description	Units	File	COMMON Block Old	COMMON Block New
ACCNEW	Acceleration new	Feet per second squared	ABIAS	ABIAS	ABIAS
ACCOLD	Acceleration old	Feet per second squared	ABIAS	ABIAS	ABIAS
ADESPD	Average desired speed	Feet per second	SUMST1	SUMSTR	SUMSTD
ADM	Average delay below XX miles per hour	Seconds	VEHDS1 VEHDS2	VEHD2	VEHD2
ADMAST	Average delay below XX miles per hour divided by the average travel time	Percent	VEHDS1 VEHDS2	VEHD2	VEHD2
ADSPD	Average desired speed	Miles per hour	VEHDS1 VEHDS2	VEHD2	VEHD2
AMAX	Vehicle maximum acceleration	Feet per second squared	CLASS	CLASSR	CLASSD
AMAXV	Average maximum acceleration	Feet per second squared	VEHDS1 VEHDS2	VEHD2	VEHD2

AO	Acceleration old for conflict checking	Feet per second squared	CONCHK	CONCHR	CONCHD
APIJR	Average PIJR time	Seconds	USER	USERR	USERD
APLVDV	Average percent log in velocity per desired speed	Percent	VEHDS1 VEHDS2	VEHD2	VEHD2
AQD	Average queue delay	Seconds	VEHDS1 VEHDS2	VEHD2	VEHD2
AQDAST	Average queue delay divided by the average travel time	Percent	VEHDS1 VEHDS2	VEHD2	VEHD2
AQUEUE	Average queue length	Vehicles	VEHDS1 VEHDS2	VEHD2	VEHD2
ASD	Average stopped delay	Seconds	VEHDS1 VEHDS2	VEHD2	VEHD2
ASDAST	Average stopped delay divided by the average travel time	Percent	VEHDS1 VEHDS2	VEHD2	VEHD2
ASPEED	Average speed	Feet per second	SUMST1	SUMSTR	SUMSTD
ASTIM	Average travel time	Seconds	VEHDS1 VEHDS2	VEHD2	VEHD2
ATD	Average total delay	Seconds	VEHDS1 VEHDS2	VEHD2	VEHD2
ATDAST	Average total delay divided by average travel time	Percent	VEHDS1 VEHDS2	VEHD2	VEHD2
AUTOL	Acceleration uniform to linear factor	None	USER	USERR	USERD
AVMT	Average vehicle miles of travel	Vehicle miles	VEHDS1 VEHDS2	VEHD2	VEHD2
AVSF	Acceleration of the vehicle on the side in front	Feet per second squared	LANECH	LANECR	LANECD
AVSR	Acceleration of the vehicle on the side in rear	Feet per second squared	LANECH	LANECR	LANECD
CAREQA	Car following equation A	Unknown	USER	USERR	USERD
CAREQL	Car following equation L	Unknown	USER	USERR	USERD
CAREQM	Car following equation M	Unknown	USER	USERR	USERD
DCHAR	Driver characteristic	None	CLASS	CLASSR	CLASSD
DCHARM	DCHAR maximum	None	CLASS	CLASSR	CLASSD

DEG2RD	Conversion degrees to radians	None	USER	USERR	USERD
DESVEL	Desired velocity	Feet per second	ABIAS	ABIAS	ABIAS
DISTAD	Total distance traveled on previous links	Feet	PRTPVA	PRTPVA	PRTPVA
DMAX	Vehicle deceleration maximum	Feet per second squared	CLASS	CLASSR	CLASSD
DMAXV	Average maximum deceleration	Feet per second squared	VEHDS1 VEHDS2	VEHD2	VEHD2
DMPH	Delay below XX miles per hour	Seconds	SUMST1	SUMSTR	SUMSTD
DT	Time step increment	Seconds	USER	USERR	USERD
DTCU	DT cubed	Seconds cubed	USER	USERR	USERD
DTMAX	Maximum DT	Seconds	USER	USERR	USERD
DTSQ	DT squared	Seconds squared	USER	USERR	USERD
DUTOL	Deceleration uniform to linear factor	None	USER	USERR	USERD
DVILNI	Sum of hesitation beyond current queue delay in LSTOP	Seconds	SIMPRO1 SIMPRO3 SIMPRO5	TWRR	TWRD
ENDLN	End of lane	Feet	ABIAS	ABIAS	ABIAS
EOM	End of max	Seconds	PHASES	PHASER	PHASED
FACTOR	Lane change factor	Feet per second squared	LANECH	LANECR	LANECD
FGOATM	Forced go active time	Seconds	VEHD	VEHDR	VEHDD
FGOTIM	Forced go time	Seconds	VEHD	VEHDR	VEHDD
FRRATM	Forced run red signal active time	Seconds	VEHD	VEHDR	VEHDD
FRRTIM	Forced run red signal time	Seconds	VEHD	VEHDR	VEHDD
FSTDTM	Forced stop dwell time	Seconds	VEHD	VEHDR	VEHDD
FSTPOS	Forced stop position	Feet	VEHD	VEHDR	VEHDD
FSTTIM	Forced stop time	seconds	VEHD	VEHDR	VEHDD
FVILNI	Sum of FHES in LSTOP	Seconds	SIMPRO1 SIMPRO3 SIMPRO5	TWRR	TWRD
GVILNI	Sum of GHES in LSTOP	Seconds	SIMPRO1 SIMPRO3 SIMPRO5	TWRR	TWRD

HESFAC	Hesitation added to PIJR for first vehicle	Seconds	USER	USERR	USERD
HVILNI	Sum of HHES in LSTOP	Seconds	SIMPRO1 SIMPRO3 SIMPRO5	TWRR	TWRD
IACC	Saved acceleration	Feet per second squared	VHED	VEHD4	VEHDD
IACC00	Zero acceleration value	Feet per second squared	PARAMS	parameter	parameter
IACCFR	Acceleration factor	None	PARAMS	parameter	parameter
IACCSH	Acceleration shift	None	PARAMS	parameter	parameter
IDTS	Saved distance traveled	Feet	VHED	VEHD4	VEHDD
IDTSI	Saved distance traveled on internal lanes for a diamond interchange	Feet	VHED	VEHD4	VEHDD
IEXTII	Extra time at login for internal lanes for a diamond interchange	Percent of DT	VEHF	VEHF4	VEHFD
IEXTIM	Extra time at login	Percent of DT	VEHF	VEHF4	VEHFD
IPOS	Saved position	Feet	VHED	VEHD4	VEHDD
IPOSFR	Position factor	None	PARAMS	parameter	parameter
ISLP	Saved slope	Feet per second cubed	VHED	VEHD4	VEHDD
ISLPFR	Slope factor	None	PARAMS	parameter	parameter
ISLPSH	Slope shift	None	PARAMS	parameter	parameter
ISPDS	Saved average desired speed (by time)	Feet per second	VHED	VEHD4	VEHDD
ISPDSI	Saved average desired speed (by time) on internal lanes for a diamond interchange	Feet per second	VHED	VEHD4	VEHDD
IVEL	Saved acceleration	Feet per second squared	VHED	VEHD4	VEHDD
IVELFR	Velocity factor	None	PARAMS	parameter	parameter
LATPFR	Lane change factor	None	PARAMS	parameter	parameter
LATPOS	Lateral position for lane change	Feet	VEHD	VEHD2	VEHDD
LATPSH	Lane change shift	None	PARAMS	parameter	parameter
OADM	Overall average delay below XX	Seconds	VEHDS1 VEHDS2	VEHD2	VEHD2

	miles per hour				
OAQD	Overall average queue delay	Seconds	VEHDS1 VEHDS2	VEHD2	VEHD2
OASD	Overall average stopped delay	Seconds	VEHDS1 VEHDS2	VEHD2	VEHD2
OATD	Overall average total delay	Seconds	VEHDS1 VEHDS2	VEHD2	VEHD2
OLDDTI	Old distance intersection	Feet	ABIAS	ABIAS	ABIAS
OLDDTS	Old distance statistics	Feet	ABIAS	ABIAS	ABIAS
P	Position of conflict	Feet	CONCHK	CONCHR	CONCHD
PDM	Percent of vehicles incurring delay below XX miles per hour	Percent	VEHDS1 VEHDS2	VEHD2	VEHD2
PI	PI	Unknown	USER	USERR	USERD
PIJR	Driver PIJR time	Seconds	CLASS	CLASSR	CLASSD
PLVDV	Percent login velocity to desired speed	None	SUMST1	SUMSTR	SUMSTD
PO	Position old for conflict checking	Feet	CONCHK	CONCHR	CONCHD
POSNEW	Position new	Feet	ABIAS	ABIAS	ABIAS
POSOLD	Position old	Feet	ABIAS	ABIAS	ABIAS
PQD	Percent of vehicles incurring queue delay	Percent	VEHDS1 VEHDS2	VEHD2	VEHD2
PSD	Percent of vehicles incurring stopped delay	Percent	VEHDS1 VEHDS2	VEHD2	VEHD2
PTD	Percent of vehicles incurring total delay	Percent	VEHDS1 VEHDS2	VEHD2	VEHD2
PTURN	Percent of vehicles making turn movement	Percent	VEHDS1 VEHDS2	VEHD2	VEHD2
PVACC	Previous vehicle acceleration	Feet per second squared	ABIAS	ABIAS	ABIAS
PVPOS	Previous vehicle position	Feet	ABIAS	ABIAS	ABIAS
PVSF	Position of the vehicle on the side in front	Feet	LANECH	LANECR	LANECD
PVSLP	Previous vehicle slope	Feet per second cubed	ABIAS	ABIAS	ABIAS

PVSR	Position of the vehicle on the side in rear	Feet	LANECH	LANECR	LANECD
PVVEL	Previous vehicle velocity	Feet per second	ABIAS	ABIAS	ABIAS
QD	Queue delay	Seconds	SUMST1	SUMSTR	SUMSTD
QGOATM	Queue-in forced go active time	Seconds	QUE	QUER	QUED
QGOTIM	Queue-in forced go time	Seconds	QUE	QUER	QUED
QRRATM	Queue-in forced run red signal active time	Seconds	QUE	QUER	QUED
QRRTIM	Queue-in forced run red signal time	Seconds	QUE	QUER	QUED
QSTDTM	Queue-in forced stop dwell time	Seconds	QUE	QUER	QUED
QSTPOS	Queue-in forced stop position	Feet	QUE	QUER	QUED
QSTTIM	Queue-in forced stop time	Seconds	QUE	QUER	QUED
QTIME	Queue-in time	Seconds	QUE	QUER	QUED
RELEND	Distance to end of lane	Feet	ABIAS	ABIAS	ABIAS
RELPOS	Distance to previous vehicle	Feet	ABIAS	ABIAS	ABIAS
RELVEL	Relative velocity to previous vehicle	Feet per second	ABIAS	ABIAS	ABIAS
SD	Stopped delay	Seconds	SUMST1	SUMSTR	SUMSTD
SLPBLK	Slope for intersection blockage	Feet per second cubed	ABIAS	ABIAS	ABIAS
SLPCON	Slope for intersection conflicts	Feet per second cubed	ABIAS	ABIAS	ABIAS
SLPLCH	Slope for a lane change	Feet per second cubed	LANECH	LANECR	LANECD
SLPMAX	Slope maximum	Feet per second cubed	ABIAS	ABIAS	ABIAS
SLPNEW	Slope new	Feet per second cubed	ABIAS	ABIAS	ABIAS
SLPNOF	Slope for NOF vehicle	Feet per second cubed	ABIAS	ABIAS	ABIAS
SLPOLD	Slope old	Feet per second cubed	ABIAS	ABIAS	ABIAS
SMSPD	Space mean speed	Feet per second	VEHDS1	VEHD2	VEHD2

			VEHDS2		
SO	Slope old for conflict checking	Feet per second cubed	CONCHK	CONCHR	CONCHD
STIME	Travel time	Seconds	SUMST1	SUMSTR	SUMSTD
STOPLD	Loop detector stop location	Feet	LOOPS	LOOPSR	LOOPSD
STRTLD	Loop detector start location	Feet	LOOPS	LOOPSR	LOOPSD
SVSF	Slope of the vehicle on the side in front	Feet per second cubed	LANECH	LANECR	LANECD
SVSR	Slope of the vehicle on the side in rear	Feet per second cubed	LANECH	LANECR	LANECD
TAIMID	Time into phase middle for max out	Seconds	PHASES	PHASER	PHASED
TAR	Time for all red phase	Seconds	PHASES	PHASER	PHASED
TCAMSP	Cam stack phase time	Seconds	SIGCAM	SIGCAR	SIGCAD
TCI	Time for clearance interval	Seconds	PHASES	PHASER	PHASED
TD	Total delay	Seconds	SUMST1	SUMSTR	SUMSTD
TGAPPC	Time to gap out	Seconds	PHASES	PHASER	PHASED
TGAPPH	Time to gap out for phase	Seconds	PHASES	PHASER	PHASED
TGRMID	Time into phase middle for gap out	Seconds	PHASES	PHASER	PHASED
TII	Time for initial interval	Seconds	PHASES	PHASER	PHASED
TIIADD	NEMA vol/den added initial per actuation	Seconds	PHASES	PHASER	PHASED
TIIMAX	NEMA vol/den maximum initial	Seconds	PHASES	PHASER	PHASED
TIIVEH	Initial interval time for vehicle	Seconds	PHASES	PHASER	PHASED
TIMRCR	Texas Diamond time for recall	Seconds	TXDSIG	TXDSIR	TXDSID
TLAG	Time for lag safety zone	Seconds	USER	USERR	USERD
TLEAD	Time for lead safety zone	Seconds	USER	USERR	USERD
TMAXPC	Time into phase for max out	Seconds	PHASES	PHASER	PHASED
TMAXPH	Time into phase for max out	Seconds	PHASES	PHASER	PHASED

TMI	Time for minimum interval (TII+TVI)	Seconds	PHASES	PHASER	PHASED
TMRSET	Texas Diamond timer setting	Seconds	TXDSIG	TXDSIR	TXDSID
TMRVAL	Texas Diamond timer value	Seconds	TXDSIG	TXDSIR	TXDSID
TMRVAT	Timer value temporary storage	Seconds	TESTER	TESTEF	TESTED
TMSPD	Time mean speed	Feet per second	VEHDS1 VEHDS2	VEHD2	VEHD2
TMTIME	Computer processing time	Seconds	SUMST1	SUMSTR	SUMSTD
TMX	Maximum extension after demand on red	Seconds	PHASES	PHASER	PHASED
TP	Time into the current phase	Seconds	SIGCAM	SIGCAR	SIGCAD
TPT	Time into phase for ring	Seconds	SIGCAM	SIGCAR	SIGCAD
TQUEUE	Time the vehicle started accumulating queue delay	Seconds	TAPE10	TAP10R	TAP10D
TR	Time remaining in phase	Seconds	SIGCAM	SIGCAR	SIGCAD
TRLAST	Time remaining last for ring	Seconds	SIGCAM	SIGCAR	SIGCAD
TRT	Time remaining in ring	Seconds	SIGCAM	SIGCAR	SIGCAD
TSTATS	Time interval for intermediate statistics	Seconds	USER	USERR	USERD
TTIMPC	Time into phase for time out	Seconds	PHASES	PHASER	PHASED
TTIMPH	Time into phase for time out	Seconds	PHASES	PHASER	PHASED
TVATIN	Time vehicle arrived at the intersection	Seconds	INTER	INTERR	INTERD
TVI	Vehicle interval	Seconds	PHASES	PHASER	PHASED
TVIBEG	Time for vehicle interval beginning	Seconds	PHASES	PHASER	PHASED
TVILNI	Sum of THES in LSTOP	Seconds	SIMPRO1 SIMPRO3 SIMPRO5	TWRR	TWRD
TVIMIN	NEMA vol/den minimum gap	Seconds	PHASES	PHASER	PHASED
TVISLP	NEMA vol/den	None	PHASES	PHASER	PHASED

	slope				
TVITBR	NEMA vol/den time before reduction	Seconds	PHASES	PHASER	PHASED
TVITTR	NEMA vol/den time to reduce	Seconds	PHASES	PHASER	PHASED
VCHAR	Vehicle characteristic	None	CLASS	CLASSR	CLASSD
VELNEW	Velocity new	Feet per second	ABIAS	ABIAS	ABIAS
VELOLD	Velocity old	Feet per second	ABIAS	ABIAS	ABIAS
VMAX	Vehicle maximum velocity	Feet per second	CLASS	CLASSR	CLASSD
VMAXA	Vehicle maximum acceleration	Feet per second	SUMST1	SUMSTR	SUMSTD
VMAXD	Vehicle maximum deceleration	Feet per second	SUMST1	SUMSTR	SUMSTD
VMT	Vehicle miles of travel	Feet	SUMST1	SUMSTR	SUMSTD
VO	Velocity old for conflict checking	Feet per second	CONCHK	CONCHR	CONCHD
VOLUME	Vehicles processed	Vehicles per hour	VEHDS1 VEHDS2	VEHD2	VEHD2
VVSF	Velocity of the vehicle on the side in front	Feet per second	LANECH	LANECR	LANECD
VVSR	Velocity of the vehicle on the side in rear	Feet per second	LANECH	LANECR	LANECD
XFPS	Value for XX for delay below XX miles per hour	Feet per second	SUMST1	SUMSTR	SUMSTD
XQDIST	Maximum distance for being in a queue	Feet	SUMST1	SUMSTR	SUMSTD
XREL	Distance for moving up	Feet	ABIAS	ABIAS	ABIAS
XRELMI	XREL minimum	Feet	ABIAS	ABIAS	ABIAS
XRELMX	XREL maximum	Feet	ABIAS	ABIAS	ABIAS

The following local variables were changed from REAL to DOUBLE PRECISION:

Subroutine	Variable Name	Description	Units	File
ACCEL	A	A/B/C for quadratic equation	Unknown	SIMPRO3
ACCEL	ACCMAX	Acceleration maximum	Feet per second squared	SIMPRO3
ACCEL	ACCN	Acceleration new	Feet per second	SIMPRO3

			squared	
ACCEL	ACCVEH	Acceleration of the vehicle	Feet per second squared	SIMPRO3
ACCEL	B	A/B/C for quadratic equation	Unknown	SIMPRO3
ACCEL	C	A/B/C for quadratic equation	Unknown	SIMPRO3
ACCEL	CRISLP	Critical slope	Feet per second cubed	SIMPRO3
ACCEL	RADICL	Radical for square root	Unknown	SIMPRO3
ACCEL	RELPN	Relative position new	Feet	SIMPRO3
ACCEL	SLOPE	Slope required to get AC/DC to zero	Feet per second cubed	SIMPRO3
ACCEL	T	Time	Seconds	SIMPRO3
ACCEL	VT	Velocity at T	Feet per second	SIMPRO3
ACDCP	CRISLP	Critical slope	Feet per second cubed	SIMPRO3
ACDCP	DISEND	Distance to the end of the lane	Feet	SIMPRO3
ACDCP	DISLCH	Distance for a lane change	Feet	SIMPRO3
ACDCP	RADICL	Radical for square root	Unknown	SIMPRO3
ACDCP	T	Time to stop	Seconds	SIMPRO3
ACDCP	TS	Time to stop	Seconds	SIMPRO3
ACDCP	VELSTP	Velocity to be considered stopped	Feet per second	SIMPRO3
ACDCP	XCRIT	Critical distance	Feet	SIMPRO3
ACTSIG	DTIME	Time at the end of the current DT	Seconds	SIMPRO2
ACTSIG	TBIG	A very large time value	Seconds	SIMPRO2
ACTSIG	TMAG1	Minimum assured green time for first single left phase following the dual left signal phase	Seconds	SIMPRO2
ACTSIG	TMAG2	Minimum assured green time for second single left phase following the dual left signal phase	Seconds	SIMPRO2
ACTSTA	AAIMID	Average NEMA vol/den initial between min & max	Seconds	SIMPRO5
ACTSTA	AGRMID	Average NEMA vol/den gap between min & max	Seconds	SIMPRO5
ACTSTA	ATGAPO	Average time into phase for gap-out	Seconds	SIMPRO5
ACTSTA	ATMAXO	Average length of main street green or average time into phase for max-out	Seconds	SIMPRO5
ACTSTA	ATTIMO	Average time into phase for time-out	Seconds	SIMPRO5
ACTSTA	PAIMAX	Percent of cycles for NEMA vol/den initial at maximum value	None	SIMPRO5
ACTSTA	PAIMID	Percent of cycles for NEMA vol/den initial between min & max	None	SIMPRO5
ACTSTA	PAIMIN	Percent of cycles for NEMA vol/den initial at minimum value	None	SIMPRO5

ACTSTA	PERGRN	Percent of green time for phase	None	SIMPRO5
ACTSTA	PGRMAX	Percent of cycles for NEMA vol/den gap at maximum value	None	SIMPRO5
ACTSTA	PGRMID	Percent of cycles for NEMA vol/den gap between min & max	None	SIMPRO5
ACTSTA	PGRMIN	Percent of cycles for NEMA vol/den gap at minimum value	None	SIMPRO5
ACTSTA	SUMGRN	Sum of green time for all phases	Seconds	SIMPRO5
ACTSTA	TOTGRN	Total green time	Seconds	SIMPRO5
ADDSTA	ASTIME	Average travel time	Seconds	SIMPRO5
AVDCON	ACCVEH	Acceleration of the vehicle	Feet per second squared	SIMPRO3
AVDCON	ACH	Acceleration at conflict for him	Feet per second squared	SIMPRO3
AVDCON	ACM	Acceleration at conflict for me	Feet per second squared	SIMPRO3
AVDCON	AFACT	Conflict angle factor	None	SIMPRO3
AVDCON	CARDIS	Car following distance	Feet	SIMPRO3
AVDCON	CRISLP	Critical slope	Feet per second cubed	SIMPRO3
AVDCON	DCH	Distance to the conflict for him	Feet	SIMPRO3
AVDCON	DCM	Distance to the conflict for me	Feet	SIMPRO3
AVDCON	DVH	Desired velocity for him	Feet per second	SIMPRO3
AVDCON	DVM	Desired velocity for me	Feet per second	SIMPRO3
AVDCON	ERRJUD	Error in judgment	Seconds	SIMPRO3
AVDCON	FHES	Hesitation factor based on number of vehicles stopped at the stop line at the intersection	none	SIMPRO3
AVDCON	GHES	Hesitation factor based on number of 50-foot (15.24-meter)sections within 200 feet (60.96 meters) of the intersection	None	SIMPRO3
AVDCON	HHES	Hesitation factor combining FHES and GHES	None	SIMPRO3
AVDCON	POSVEH	Position of the vehicle	Feet	SIMPRO3
AVDCON	SLOPE	Slope	Feet per second cubed	SIMPRO3
AVDCON	SLPTCM	Slope for time to conflict for me	Feet per second cubed	SIMPRO3
AVDCON	SLPTFZ	Slope for time to front zone	Feet per second cubed	SIMPRO3
AVDCON	SLPTMP	Slope temporary	Feet per second cubed	SIMPRO3
AVDCON	SLPTRZ	Slope for time to rear zone	Feet per second cubed	SIMPRO3

AVDCON	SLPVEH	Slope of the vehicle	Feet per second cubed	SIMPRO3
AVDCON	TCH	Time to conflict for him	Seconds	SIMPRO3
AVDCON	TCM	Time to conflict for me	Seconds	SIMPRO3
AVDCON	TCRASH	Time to crash	Seconds	SIMPRO3
AVDCON	TFZ	Time for front zone	Seconds	SIMPRO3
AVDCON	THES	Time for hesitation	Seconds	SIMPRO3
AVDCON	TIM	Time to conflict for checking	Seconds	SIMPRO3
AVDCON	TMP	Temporary time	Seconds	SIMPRO3
AVDCON	TPASSC	Time to pass through the conflict	Seconds	SIMPRO3
AVDCON	TPASSH	Time to pass through the conflict for him	Seconds	SIMPRO3
AVDCON	TPASSM	Time to pass through the conflict for me	Seconds	SIMPRO3
AVDCON	TRZ	Time for rear zone	Seconds	SIMPRO3
AVDCON	VCH	Velocity at conflict for him	Feet per second	SIMPRO3
AVDCON	VCM	Velocity at conflict for me	Feet per second	SIMPRO3
AVDCON	VELREL	Velocity relative	Feet per second	SIMPRO3
AVDCON	VELVEH	Velocity of the vehicle	Feet per second	SIMPRO3
BANGS	POSL	Lateral position for previous vehicle	Feet	SIMPRO3
BANGS	POSLAT	Lateral position for current vehicle	Feet	SIMPRO3
CALNEW	ACC	Acceleration	Feet per second squared	SIMPRO3
CALNEW	ACCMAX	Acceleration maximum for the driver	Feet per second squared	SIMPRO3
CALNEW	ACCTGT	Acceleration target	Feet per second squared	SIMPRO3
CALNEW	ACCTIM	Time to bring acc/dec to zero using TQCSLP	Seconds	SIMPRO3
CALNEW	ACCVEH	Acceleration for the vehicle	Feet per second squared	SIMPRO3
CALNEW	DECMAX	Deceleration maximum	Feet per second squared	SIMPRO3
CALNEW	POS	Position	Feet	SIMPRO3
CALNEW	SLP	Slope	Feet per second cubed	SIMPRO3
CALNEW	SLPACC	Slope required to bring the current acc/dec to the target acc/dec in one second	Feet per second cubed	SIMPRO3
CALNEW	SLPCRI	Slope critical	Feet per second cubed	SIMPRO3
CALNEW	SLPVEL	Slope required to bring the current velocity to the target velocity in time ACCTIM	Feet per second cubed	SIMPRO3
CALNEW	TQCSLP	75% critical slope	Feet per second	SIMPRO3

			cubed	
CALNEW	VEL	Velocity	Feet per second	SIMPRO3
CALNEW	VELDIF	Velocity difference between current and target	Feet per second	SIMPRO3
CALNEW	VELTGT	Velocity target	Feet per second	SIMPRO3
CARFOL	A	A/B/C for quadratic equation	Unknown	SIMPRO3
CARFOL	ACCMAX	Acceleration maximum	Feet per second squared	SIMPRO3
CARFOL	ACCN	Acceleration needed	Feet per second squared	SIMPRO3
CARFOL	B	A/B/C for quadratic equation	Unknown	SIMPRO3
CARFOL	C	A/B/C for quadratic equation	Unknown	SIMPRO3
CARFOL	CARDEC	Car following equation deceleration	Feet per second squared	SIMPRO3
CARFOL	CARDIS	Car following distance	Feet	SIMPRO3
CARFOL	CRISLP	Critical slope	Feet per second cubed	SIMPRO3
CARFOL	DECVEH	Deceleration of the vehicle	Feet per second squared	SIMPRO3
CARFOL	DIST	Distance traveled	Feet	SIMPRO3
CARFOL	FACT	Factor	Various	SIMPRO3
CARFOL	LATNOW	Lateral position now	Feet	SIMPRO3
CARFOL	PERNEW	Percent SLPNEW	None	SIMPRO3
CARFOL	PVSTP	Position vehicle ahead when stopped	Feet	SIMPRO3
CARFOL	RADICL	Radical for square root	Unknown	SIMPRO3
CARFOL	RANGE	Range for slope	Feet per second cubed	SIMPRO3
CARFOL	SFACT	Slope factor	None	SIMPRO3
CARFOL	SLOPE	Slope	Feet per second cubed	SIMPRO3
CARFOL	SLOPEU	Slope	Feet per second cubed	SIMPRO3
CARFOL	SLPSTP	Slope to stop	Feet per second cubed	SIMPRO3
CARFOL	SPD	Speed	Feet per second	SIMPRO3
CARFOL	T	Time	Seconds	SIMPRO3
CARFOL	T1	Time	Seconds	SIMPRO3
CARFOL	TS	Time to stop	Seconds	SIMPRO3
CARFOL	VT1	Velocity at T1	Feet per second	SIMPRO3
CARFOL	XCRIT	Critical distance	Feet	SIMPRO3
CHGMLN	DECMAX	Deceleration maximum	Feet per second squared	SIMPRO3
CHGMLN	F3	4/3	None	SIMPRO3
CHGMLN	POSLAT	Position lateral	Feet	SIMPRO3
CHGMLN	XCRIT	Critical distance	Feet	SIMPRO3

CHKCON	ACH	Acceleration at the conflict for him	Feet per second squared	SIMPRO3
CHKCON	ACM	Acceleration at the conflict for me	Feet per second squared	SIMPRO3
CHKCON	AFACT	Angle factor	None	SIMPRO3
CHKCON	CARDIS	Car following distance	Feet	SIMPRO3
CHKCON	DCH	Distance to the conflict for him	Feet	SIMPRO3
CHKCON	DCM	Distance to the conflict for me	Feet	SIMPRO3
CHKCON	DVH	Desired velocity for him	Feet per second	SIMPRO3
CHKCON	DVM	Desired velocity for me	Feet per second	SIMPRO3
CHKCON	ERRJUD	Error in judgment	Seconds	SIMPRO3
CHKCON	FCLEAR	Front clear time	Seconds	SIMPRO3
CHKCON	FHES	Hesitation factor based on number of vehicles stopped at the stop line at the intersection	none	SIMPRO3
CHKCON	GHES	Hesitation factor based on number of 50-foot (15.24-meter) sections within 200 feet (60.96 meters) of the intersection	None	SIMPRO3
CHKCON	HHES	Hesitation factor combining FHES and GHES	None	SIMPRO3
CHKCON	RCLEAR	Rear clear time	Seconds	SIMPRO3
CHKCON	SLOPE	75% of maximum slope for driver	Feet per second cubed	SIMPRO3
CHKCON	TCH	Time to the conflict for him	Seconds	SIMPRO3
CHKCON	TCM	Time to the conflict for me	Seconds	SIMPRO3
CHKCON	TCRASH	Time to crash	Seconds	SIMPRO3
CHKCON	TFZ	Time for front zone	Seconds	SIMPRO3
CHKCON	THES	Hesitation time using HHES	Seconds	SIMPRO3
CHKCON	TIM	Maximum time from the end of the lane that this vehicle may decide to proceed if the intersection conflicts are clear	Seconds	SIMPRO3
CHKCON	TMP	Temporary time	Seconds	SIMPRO3
CHKCON	TPASSC	Time to pass car following distance	Seconds	SIMPRO3
CHKCON	TPASSH	Time to pass through the conflict for him	Seconds	SIMPRO3
CHKCON	TPASSM	Time to pass through the conflict for me	Seconds	SIMPRO3
CHKCON	TRZ	Time for the rear zone	Seconds	SIMPRO3
CHKCON	VCH	Velocity at the conflict for him	Feet per second	SIMPRO3
CHKCON	VCM	Velocity at the conflict for me	Feet per second	SIMPRO3
CHKCON	VELREL	Velocity relative between him and me	Feet per second	SIMPRO3
CHKDSP	RADICL	Radical for square root	Unknown	SIMPRO3

CHKDSP	SLOPE	Slope desired	Feet per second cubed	SIMPRO3
CHKDSP	SPD	Desired speed for path	Feet per second	SIMPRO3
CHKDSP	T	Time to reduce speed	Seconds	SIMPRO3
CHKDSP	XCRIT	Critical distance	Feet	SIMPRO3
CHKINT	ACCVEH	Acceleration of the vehicle	Feet per second squared	SIMPRO3
CHKINT	DISAVL	Distance available	Feet	SIMPRO3
CHKINT	PIVJL	Position of the IVJL vehicle	Feet	SIMPRO3
CHKINT	POSVEH	Position of the vehicle	Feet	SIMPRO3
CHKINT	SLPVEH	Slope of the vehicle	Feet per second cubed	SIMPRO3
CHKINT	SUMLEN	Sum of the length of the vehicles	Feet	SIMPRO3
CHKINT	VELVEH	Velocity of the vehicle	Feet per second	SIMPRO3
CHKINT	VIVIL	Velocity of the IVIL vehicle	Feet per second	SIMPRO3
CHKINT	VIVJL	Velocity of the IVJL vehicle	Feet per second	SIMPRO3
CHKINT	VIVJP	Velocity of the IVJP vehicle	Feet per second	SIMPRO3
CHKLDT	DSTOP	Distance to stop of detector	Feet	SIMPRO3
CHKLDT	DSTRT	Distance to start of detector	Feet	SIMPRO3
CHKLDT	POSNRB	Position new for rear bumper	Feet	SIMPRO3
CHKLDT	POSORB	Position old for rear bumper	Feet	SIMPRO3
CHKNOF	ACCVEH	Acceleration of the vehicle	Feet per second squared	SIMPRO3
CHKNOF	POSRB	Position of the rear bumper	Feet	SIMPRO3
CHKNOF	POSVEH	Position of the vehicle	Feet	SIMPRO3
CHKNOF	SLPVEH	Slope of the vehicle	Feet per second cubed	SIMPRO3
CHKNOF	VELVEH	Velocity of the vehicle	Feet per second	SIMPRO3
CHKSDR	ACM	Acceleration at the conflict for me	Feet per second squared	SIMPRO3
CHKSDR	DCH	Distance to the conflict for him for C4 statements	Feet	SIMPRO3
CHKSDR	DCM	Distance to the conflict for me	Feet	SIMPRO3
CHKSDR	ERRJUD	Error in judgment	Seconds	SIMPRO3
CHKSDR	TCH	Time to the conflict for him		SIMPRO3
CHKSDR	TCM	Time to the conflict for me	Seconds	SIMPRO3
CHKSDR	TFZ	Time for front zone	Seconds	SIMPRO3
CHKSDR	TIM	Maximum time from the end of the lane that this vehicle may decide to proceed if the sight distance restrictions are clear	Seconds	SIMPRO3
CHKSDR	TPASSM	Time to pass through the conflict for me	Seconds	SIMPRO3
CHKSDR	VCM	Velocity at the conflict for me	Feet per second	SIMPRO3
CRIDIS	CRISLP	Critical slope	Feet per second	SIMPRO3

			cubed	
CRIDIS	DECMAX	Deceleration maximum	Feet per second squared	SIMPRO3
CRIDIS	DECTST	Deceleration for testing	Feet per second squared	SIMPRO3
CRIDIS	DISPRT	Distance traveled during REACTT	Feet	SIMPRO3
CRIDIS	F3	4/3	None	SIMPRO3
CRIDIS	OLDACC	Old acceleration	Feet per second squared	SIMPRO3
CRIDIS	OLDVEL	Velocity old	Feet per second	SIMPRO3
CRIDIS	RADICL	Radical for square root	Unknown	SIMPRO3
CRIDIS	REACTT	Reaction time	Seconds	SIMPRO3
CRIDIS	RELNEW	Relative distance new	Feet	SIMPRO3
CRIDIS	RELOLD	Relative distance old	Feet	SIMPRO3
CRIDIS	SLPN	Slope new	Feet per second cubed	SIMPRO3
CRIDIS	T	Time	Seconds	SIMPRO3
CRIDIS	TS	Time to stop	Seconds	SIMPRO3
CRIDIS	V	Velocity after T seconds	Feet per second	SIMPRO3
CRIDIS	X	Distance after T seconds	Feet	SIMPRO3
CRIDIS	XCRIT	Critical distance	Feet	SIMPRO3
DELAY	AVRF	AVSF saved for the lane to the right	Feet per second squared	SIMPRO3
DELAY	AVRR	AVSR saved for the lane to the right	Feet per second squared	SIMPRO3
DELAY	PVRF	PVSF saved for the lane to the right	Feet	SIMPRO3
DELAY	PVRR	PVSR saved for the lane to the right	Feet	SIMPRO3
DELAY	QUEA	Queue length addition value added	Feet	SIMPRO3
DELAY	QUEC	Queue length for the current lane	Feet	SIMPRO3
DELAY	QUEL	Queue length for the lane to the left	Feet	SIMPRO3
DELAY	QUER	Queue length for the lane to the right	Feet	SIMPRO3
DELAY	SVRF	SVSF saved for the lane to the right	Feet per second cubed	SIMPRO3
DELAY	SVRR	SVSR saved for the lane to the right	Feet per second cubed	SIMPRO3
DELAY	VVRF	VVSF saved for the lane to the right	Feet per second	SIMPRO3
DELAY	VVRR	VVSR saved for the lane to the right	Feet per second	SIMPRO3
EXEC	POSPTH	Position on path	Feet	SIMPRO3
FLGNOR	JACC	Acceleration for the NORT vehicle	Feet per second squared	SIMPRO3
FNDXYA	ANGLNG	Longitudinal angle of the leg	Degrees	SIMPRO3
FNDXYA	ANGTRN	Transverse angle of the leg	Degrees	SIMPRO3
FNDXYA	POS	Position on lane	Feet	SIMPRO3
FNDXYA	POSBEG	Position relative to the beginning of the lane	Feet	SIMPRO3

FNDXYA	POSLAT	Position lateral on lane	Feet	SIMPRO3
FNDXYP	ANGLE	Angle on arc portion of path	Degrees	SIMPRO3
FNDXYP	PERLEN	Percent of segment used	None	SIMPRO3
FNDXYP	POS	Position on path	Feet	SIMPRO3
FNDXYP	POSBEG	Position relative to the beginning of the path segment	Feet	SIMPRO3
GAPACC	A	A/B/C for quadratic equation	Unknown	SIMPRO3
GAPACC	ACCVEH	Acceleration of the vehicle	Feet per second squared	SIMPRO3
GAPACC	ALAGAP	Acceptable lag gap	Feet	SIMPRO3
GAPACC	ALEGAP	Acceptable lead gap	Feet	SIMPRO3
GAPACC	AME	Acceleration me	Feet per second squared	SIMPRO3
GAPACC	AVR	Acceleration of the vehicle to the rear	Feet per second squared	SIMPRO3
GAPACC	B	A/B/C for quadratic equation	Unknown	SIMPRO3
GAPACC	C	A/B/C for quadratic equation	Unknown	SIMPRO3
GAPACC	DISLCH	Distance for a lane change	Feet	SIMPRO3
GAPACC	DISVEH	Distance for vehicle	Feet	SIMPRO3
GAPACC	FACT	Factor	None	SIMPRO3
GAPACC	FLENV	Four times the length of the vehicle	Feet	SIMPRO3
GAPACC	GAPLA	Gap lag	Feet	SIMPRO3
GAPACC	GAPLE	Gap lead	Feet	SIMPRO3
GAPACC	GAPMIN	Gap minimum	Feet	SIMPRO3
GAPACC	GAPNEW	Gap new	Feet	SIMPRO3
GAPACC	GAPOLD	Gap distance old	Feet	SIMPRO3
GAPACC	GFACT	Gap factor	Feet	SIMPRO3
GAPACC	PME	Position me	Feet	SIMPRO3
GAPACC	PVR	Position of the vehicle to the rear	Feet	SIMPRO3
GAPACC	RADICL	Radical for square root	Unknown	SIMPRO3
GAPACC	RESPLA	Relative speed lag	Feet per second	SIMPRO3
GAPACC	RESPLE	Relative speed lead	Feet per second	SIMPRO3
GAPACC	SLOPE	Slope	Feet per second cubed	SIMPRO3
GAPACC	SME	Slope me	Feet per second cubed	SIMPRO3
GAPACC	SVR	Slope of the vehicle to the rear	Feet per second cubed	SIMPRO3
GAPACC	TM	Time to stop for me	Seconds	SIMPRO3
GAPACC	TS	Time to stop for side vehicle	Seconds	SIMPRO3
GAPACC	VELMIN	Velocity minimum	Feet per second	SIMPRO3
GAPACC	VM	Velocity for me	Feet per second	SIMPRO3
GAPACC	VME	Velocity me	Feet per second	SIMPRO3
GAPACC	VMET	Velocity for me temporary	Feet per second	SIMPRO3
GAPACC	VS	Velocity of the vehicle on the side at	Feet per second	SIMPRO3

		TS		
GAPACC	VVR	Velocity of the vehicle to the rear	Feet per second	SIMPRO3
GAPACC	VVRT	Velocity of the vehicle to the rear temporary	Feet per second	SIMPRO3
GAPACC	XCRIT	Critical distance	feet	SIMPRO3
GAPACC	XM	Distance to stop for me	Feet	SIMPRO3
GAPACC	XS	Distance to stop for side vehicle	Feet	SIMPRO3
IBAP	DISEND	Distance to the end of the lane	Feet	SIMPRO3
IBAP	DISLCH	Distance for a lane change	Feet	SIMPRO3
IBAP	FLENV	Four times the length of the vehicle	Feet	SIMPRO3
IBAP	POSCHK	Position for checking if vehicle is in the queue	Feet	SIMPRO3
IBAP	POSLAT	Position lateral	Feet	SIMPRO3
IBAP	POSLC7	Lateral position for C7 statements	Feet	SIMPRO3
IBAP	POSLCS	Lateral position for CS statements	Feet	SIMPRO3
IBAP	POSR	Position of rear bumper	Feet	SIMPRO3
IBAP	PVILNI	Position of vehicle for lane and approach	Feet	SIMPRO3
IBAP	TESTLP	Test lateral position	Feet	SIMPRO3
INISTA	XMPH	XX of delay below XX miles per hour	Miles per hour	SIMPRO5
INTERP	POSLC7	Position lateral for C7 statements	Feet	SIMPRO3
INTERP	POSR	Position of rear bumper	Feet	SIMPRO3
INTLOG	DECMAX	Deceleration maximum	Feet per second squared	SIMPRO3
INTLOG	F3	4/3	None	SIMPRO3
INTLOG	XCRIT	Critical distance	Feet	SIMPRO3
INTSTA	AVGQ	Average queue length	Feet	SIMPRO3
INTSTA	OASD	Overall average stopped delay	Seconds	SIMPRO3
INTSTA	PDELAY	Percent stopped delay	None	SIMPRO3
INTSTA	PTURN	Percent of vehicle making turn movement	none	SIMPRO3
INTSTA	SUMDEL	Sum of total delay	Seconds	SIMPRO3
INTSTA	SUMVOL	Sum of total volume	Vehicles	SIMPRO3
INTSTA	TIMNOW	Time into the simulation since start-up time	Seconds	SIMPRO3
INTSTA	TMINT	Computer processing time used since last call	Seconds	SIMPRO3
INTSTA	TMSIM	Computer processing time used since startup time	Seconds	SIMPRO3
INTSTA	TOTDEL	Total stopped delay for leg	Seconds	SIMPRO3
INTSTA	TOTVOL	Total volume for leg	Vehicles	SIMPRO3
INTSTA	VOLUME	Equivalent hourly volume processed	Vehicles per hour	SIMPRO3
LCHDES	ALEGAP	Acceptable lead gap	Feet	SIMPRO3

LCHDES	ALEGAS	Acceptable lead gap for C) statements	Feet	SIMPRO3
LCHDES	CARDIS	Car following distance	Feet	SIMPRO3
LCHDES	CRISLP	Critical slope	Feet per second cubed	SIMPRO3
LCHDES	FACT	Factor	None	SIMPRO3
LCHDES	GAPMIN	Gap minimum	Feet	SIMPRO3
LCHDES	PREVM	Predicted velocity for me	Feet per second	SIMPRO3
LCHDES	PREVS	Predicted velocity of vehicle on side	Feet per second	SIMPRO3
LCHDES	RELDIS	Relative distance	Feet	SIMPRO3
LCHDES	RELSCD	Relative speed	Feet per second	SIMPRO3
LCHDES	RELSPD	Relative speed	Feet per second	SIMPRO3
LCHDES	VELMIN	Velocity minimum	Feet per second	SIMPRO3
LCHGEO	DVFACT	Driver and vehicle factor	None	SIMPRO3
LCHGEO	POSLAT	Position lateral	Feet	SIMPRO3
LCHGEO	TLDIST	Total lateral distance for a lane change	Feet	SIMPRO3
LCHGEO	VEHLEN	Vehicle length	Feet	SIMPRO3
LCHGEO	XNEW	New position on the cosine curve	Feet	SIMPRO3
LCHGEO	XOLD	Old position on the cosine curve	Feet	SIMPRO3
LCHGEO	XTOT	Total longitudinal distance for a lane change	Feet	SIMPRO3
LOGIBI	DTIME	Delta time for C(statements	Seconds	SIMPRO3
LOGIBI	POSTOT	Position total traveled for C(statements	Feet	SIMPRO3
LOGIIN	DPOS	Distance traveled on inbound lane	Feet	SIMPRO3
LOGIIN	VELAVG	Velocity average	Feet per second	SIMPRO3
LOGIN	CRISLP	Critical slope	Feet per second cubed	SIMPRO3
LOGIN	DECMAX	Deceleration maximum	Feet per second squared	SIMPRO3
LOGIN	DISEND	Distance to the end of the lane	Feet	SIMPRO3
LOGIN	DISLCH	Distance for a lane change	Feet	SIMPRO3
LOGIN	DIST	Distance from previous vehicle to start of lane	Feet	SIMPRO3
LOGIN	FACT	Factor	None	SIMPRO3
LOGIN	ONETRD	1/3	None	SIMPRO3
LOGIN	POSLC7	Lateral position for C7 statements	Feet	SIMPRO3
LOGIN	POSLCS	Lateral position for CS statements	Feet	SIMPRO3
LOGIN	SLPN	Slope new	Feet per second cubed	SIMPRO3
LOGIN	T	Time to stop or reduce speed	Seconds	SIMPRO3
LOGIN	TSTP	Time to stop	Seconds	SIMPRO3
LOGIN	V	Velocity after T seconds	Feet per second	SIMPRO3
LOGIN	XSTP	Position after TSTP seconds	Feet	SIMPRO3

LOGIN	XTIMEL	Extra time at login	Seconds	SIMPRO3
LOGOUT	ACMXV	Acceleration maximum for the vehicle	Feet per second squared	SIMPRO3
LOGOUT	AVGSPD	Average speed (total speed/total intervals)	Feet per second	SIMPRO3
LOGOUT	AVGVEL	Average speed (total distance/total time)	Feet per second	SIMPRO3
LOGOUT	DCMXV	Deceleration maximum for the vehicle	Feet per second squared	SIMPRO3
LOGOUT	DESPD	Average desired speed	Miles per hour	SIMPRO3
LOGOUT	XDISTL	Distance traveled during last DT	Feet	SIMPRO3
LOGOUT	XDMPH	Delay below XX miles per hour	Seconds	SIMPRO3
LOGOUT	XQD	Queue delay	Seconds	SIMPRO3
LOGOUT	XSD	Stopped delay	Seconds	SIMPRO3
LOGOUT	XSTIME	Total simulation time for the vehicle	Seconds	SIMPRO3
LOGOUT	XTD	Total delay	Seconds	SIMPRO3
LOGOUT	XVMT	Vehicle miles of travel	Feet	SIMPRO3
LOKFMR	DISEND	Distance to the end of the lane	Feet	SIMPRO3
LOKFMR	DISLCH	Distance for a lane change	Feet	SIMPRO3
LOKFMR	POSLT	Position of the end of the left turn bay	Feet	SIMPRO3
LOKFMR	POSRB	Position of the rear bumper of the vehicle	Feet	SIMPRO3
LOKFMR	POSRM	Position of the rear bumper of the look ahead vehicle	Feet	SIMPRO3
LOKIBI	POSRB	Position of the rear bumper of the vehicle	Feet	SIMPRO3
LOKIOB	POSRB	Position of the rear bumper of the vehicle	Feet	SIMPRO3
LSTOP	FHES	Hesitation factor based on number of vehicles stopped at the stop line at the intersection	none	SIMPRO3
LSTOP	GHES	Hesitation factor based on number of 50-foot (15.24-meter) sections within 200 feet (60.96 meters) of the intersection	None	SIMPRO3
LSTOP	HHES	Hesitation factor combining FHES and GHES	None	SIMPRO3
LSTOP	THES	Hesitation time using HHES	Seconds	SIMPRO3
NEMA8	TBIG	A very large time value	Seconds	SIMPRO2
NEMA8	TIMRMX	Maximum TCI+TAR for phases	Seconds	SIMPRO2
NEMA8	TIMRMX	Timer last value	Seconds	SIMPRO2
NEWVEL	CRISLP	Critical slope	Feet per second cubed	SIMPRO3
NEWVEL	DPOS	Delta position	Feet	SIMPRO3

NEWVEL	SLPTMP	Temporary slope	Feet per second cubed	SIMPRO3
NEWVEL	T	Time	Seconds	SIMPRO3
NEWVEL	TCU	Time cubed	Seconds cubed	SIMPRO3
NEWVEL	TSQ	Time squared	Seconds squared	SIMPRO3
OBAP	DISEND	Distance to the end of the lane	Feet	SIMPRO3
OBAP	DISLCH	Distance for a lane change	Feet	SIMPRO3
OBAP	POSLAT	Position lateral	Feet	SIMPRO3
OBAP	POSLC7	Position lateral for C7 statements	Feet	SIMPRO3
OBAP	POSLCA	Position lateral for CA statements	Feet	SIMPRO3
OBAP	POSR	Position of rear bumper	Feet	SIMPRO3
PREDTV	A	A/B/C for quadratic equation	Unknown	SIMPRO3
PREDTV	ACCM	Acceleration maximum for driver	Feet per second squared	SIMPRO3
PREDTV	ACCN	Acceleration new to get to desired speed	Feet per second squared	SIMPRO3
PREDTV	ACCV	Acceleration maximum for vehicle	Feet per second squared	SIMPRO3
PREDTV	AN	Acceleration new	Feet per second squared	SIMPRO3
PREDTV	AX	Acceleration at the conflict	Feet per second squared	SIMPRO3
PREDTV	B	A/B/C for quadratic equation	Unknown	SIMPRO3
PREDTV	C	A/B/C for quadratic equation	Unknown	SIMPRO3
PREDTV	CRISLP	Critical slope	Feet per second cubed	SIMPRO3
PREDTV	DV	Desired velocity	Feet per second	SIMPRO3
PREDTV	PN	Position new	Feet	SIMPRO3
PREDTV	RADICL	Radical for square root	Unknown	SIMPRO3
PREDTV	RELDIS	Relative distance to the end of the lane	Feet	SIMPRO3
PREDTV	SLOPE	37.5% of maximum slope for the driver	Feet per second cubed	SIMPRO3
PREDTV	SN	Slope new	Feet per second cubed	SIMPRO3
PREDTV	SPD	Desired speed for the intersection path	Feet per second	SIMPRO3
PREDTV	T	Time to the conflict	Seconds	SIMPRO3
PREDTV	TT	Time to reduce current speed to SPD	Seconds	SIMPRO3
PREDTV	VN	Velocity new	Feet per second	SIMPRO3
PREDTV	VTT	Velocity at time TT	Feet per second	SIMPRO3
PREDTV	VX	Velocity at the conflict	Feet per second	SIMPRO3
PREDTV	XCRIT	Critical distance	Feet	SIMPRO3
PREDTV	XPER	Percent of DT used to get to conflict	None	SIMPRO3

PREDTV	XT	Time the vehicle stopped	seconds	SIMPRO3
PRENEW	PAN	Predicted acceleration new	Feet per second squared	SIMPRO3
PRENEW	PAO	Predicted acceleration old	Feet per second squared	SIMPRO3
PRENEW	PPN	Predicted position new	Feet	SIMPRO3
PRENEW	PPO	Predicted position old	Feet	SIMPRO3
PRENEW	PSN	Predicted slope new	Feet per second cubed	SIMPRO3
PRENEW	PSO	Predicted slope old	Feet per second cubed	SIMPRO3
PRENEW	PVN	Predicted velocity new	Feet per second	SIMPRO3
PRENEW	PVO	Predicted velocity old	Feet per second	SIMPRO3
PRENEW	RADICL	Radical for square root	Unknown	SIMPRO3
PRENEW	T	Time for predicting pos/vel/acc/slp	Seconds	SIMPRO3
PRENEW	VELSTP	Velocity to be considered stopped	Feet per second	SIMPRO3
PREOLD	PAN	Predict acceleration new	Feet per second squared	SIMPRO3
PREOLD	PAO	Predict acceleration old	Feet per second squared	SIMPRO3
PREOLD	PPN	Predict position new	Feet	SIMPRO3
PREOLD	PPO	Predict position old	Feet	SIMPRO3
PREOLD	PSN	Predict slope new	Feet per second cubed	SIMPRO3
PREOLD	PSO	Predict slope old	Feet per second cubed	SIMPRO3
PREOLD	PVN	Predict velocity new	Feet per second	SIMPRO3
PREOLD	PVO	Predict velocity old	Feet per second	SIMPRO3
PREOLD	VELSTP	Velocity for stop	Feet per second	SIMPRO3
PRESIG	DTIME	Time at end of DT	Seconds	SIMPRO2
PREST1	DISEND	Distance to the end of the lane	Feet	SIMPRO3
PREST1	DISLCH	Distance for a lane change	Feet	SIMPRO3
PUNSTA	ANVSY	Average number of vehicle in the system	None	SIMPRO5
PUNSTA	XMPH	XX of delay below XX miles per hour	Miles per hour	SIMPRO5
RDVPRD	PIJRMI	PIJR minimum value	Seconds	SIMPRO2
RGEOPD	DISDT	Distance traveled in one DT	Feet	SIMPRO2
RGEOPD	DISLN	Length of the inbound lane at the beginning	Feet	SIMPRO2
RPHASD	TEST	Test value	Seconds	SIMPRO2
RUSERD	XMPH	XX for delay below XX miles per hour	Miles per hour	SIMPRO2
RUSERD	XTMAX	DT maximum value	Seconds	SIMPRO2
SETCON	IPOSCK	Position for checking	Feet	SIMPRO3

SETCON	JPOS	Position of the INOW vehicle	Feet	SIMPRO3
SETCON	POSLAT	Lateral position for a lane change	Feet	SIMPRO3
SIGRES	CRISLP	Critical slope	Feet per second cubed	SIMPRO3
SIGRES	DECMAX	Deceleration maximum	Feet per second squared	SIMPRO3
SIGRES	RADICL	Radical for square root	Unknown	SIMPRO3
SIGRES	SCRIT	Slope critical	Feet per second cubed	SIMPRO3
SIGRES	TCRIT	Time critical	Seconds	SIMPRO3
SIGRES	XCRIT	Distance critical	Feet	SIMPRO3
SLPCFS	ACCVEH	Acceleration of vehicle	Feet per second squared	SIMPRO3
SLPCFS	CARDEC	Car following deceleration	Feet per second	SIMPRO3
SLPCFS	CARDIS	Car following distance	Feet	SIMPRO3
SLPCFS	CRISLP	Critical slope	Feet per second cubed	SIMPRO3
SLPCFS	DECMAX	Deceleration maximum	Feet per second squared	SIMPRO3
SLPCFS	DENOM	Denominator	Feet per second squared	SIMPRO3
SLPCFS	FACT	Factor	None	SIMPRO3
SLPCFS	POSREL	Position relative	Feet	SIMPRO3
SLPCFS	POSSTP	Position where vehicle ahead stops	Feet	SIMPRO3
SLPCFS	POSVEH	Position of vehicle	Feet	SIMPRO3
SLPCFS	RADICL	Radical for square root	Unknown	SIMPRO3
SLPCFS	RADICL	Radical for square root	Unknown	SIMPRO3
SLPCFS	RANGE	Range for slope	Feet per second cubed	SIMPRO3
SLPCFS	SFACT	Slope factor	None	SIMPRO3
SLPCFS	SLPCAL	Slope calculated	Feet per second cubed	SIMPRO3
SLPCFS	SLPSTP	Slope of vehicle to stop	Feet per second cubed	SIMPRO3
SLPCFS	SLPVEH	Slope of vehicle	Feet per second cubed	SIMPRO3
SLPCFS	TCRIT	Time critical	Seconds	SIMPRO3
SLPCFS	TS	Time to stop	Seconds	SIMPRO3
SLPCFS	VELREL	Velocity relative	Feet per second	SIMPRO3
SLPCFS	VELVEH	Velocity of vehicle	Feet per second	SIMPRO3
SLPCFS	XCRIT	Critical distance	Feet	SIMPRO3
SSIBAP	POSCHK	Position to check for being in a queue	Feet	SIMPRO3
SSIBAP	SPFACT	Speed factor	None	SIMPRO3
SUMARY	DHESAV	Average of hesitation beyond current queue delay from LSTOP	Seconds	SIMPRO5

SUMARY	FHESAV	Average of FHES from LSTOP	Seconds	SIMPRO5
SUMARY	GHESAV	Average of GHES from LSTOP	Seconds	SIMPRO5
SUMARY	HHESAV	Average of HHES from LSTOP	Seconds	SIMPRO5
SUMARY	THESAV	Average of THES from LSTOP	Seconds	SIMPRO5
SVEHU	DPP	Distance from end of path	Feet	SIMPRO3
SVEHU	TPMIN	Minimum TPP	Seconds	SIMPRO3
SVEHU	TPP	Time to travel DPP	Seconds	SIMPRO3
SVEHU	VELMIN	Minimum velocity	Feet per second	SIMPRO3
TIMSTA	ANVSY	Average number of vehicles in the system during simulation time	Vehicles	SIMPRO5
TIMSTA	COSTIN	Computer time cost for initial time	Dollars	SIMPRO5
TIMSTA	COSTSI	Computer time cost for simulation time	Dollars	SIMPRO5
TIMSTA	COSTSS	Computer time cost for summary statistics time	Dollars	SIMPRO5
TIMSTA	COSTSU	Computer time cost for start up time	Dollars	SIMPRO5
TIMSTA	COSTTO	Computer time cost for total time	Dollars	SIMPRO5
TIMSTA	TMIN	Computer time for initial time	Seconds	SIMPRO5
TIMSTA	TMRAT	Vehicle-seconds of simulation per computer time	None	SIMPRO5
TIMSTA	TMRDT	Vehicle updates per computer time	None	SIMPRO5
TIMSTA	TMRSI	Simulation real time to computer time ratio	None	SIMPRO5
TIMSTA	TMRSU	Start up real time to computer time ratio	None	SIMPRO5
TIMSTA	TMSI	Computer time for simulation	Seconds	SIMPRO5
TIMSTA	TMSS	Computer time for summary statistics	Seconds	SIMPRO5
TIMSTA	TMSU	Computer time for start up	Seconds	SIMPRO5
TIMSTA	TMTO	Computer time total	Seconds	SIMPRO5
TX3467	TBIG	A very large time value	Seconds	SIMPRO2
UNBIAS	RADICL	Radical for square root	Unknown	SIMPRO3
UNBIAS	T	Time	Seconds	SIMPRO3

The following function calls were changed:

Old Function	New Function
ABS	DABS
ACOS	DACOS
AMAX1	DMAX1
AMIN1	DMIN1
ATAN	DATAN
COS	DCOS
FLOAT	DBLE

IFIX	IDINT
NINT	IDNINT
SIN	DSIN
SQRT	DSQRT

The PARAMETER VELSTP was moved from subroutine PRENEW, PREOLD, and ACDCP to file PARAMS, converted from type REAL to type DOUBLE PRECISION, and the value was reduced from 0.5 feet (0.1524 meters) to 0.1 feet (0.03048 meters) per second. For a vehicle to be considered to have come to a stop, the speed must be less than VELSTP and the acceleration/deceleration and slope must be negative. This definition was necessitated because when a vehicle was trying to accelerate from a stopped condition using a DT of 0.01 seconds, the new velocity would be very small, and thus the old logic would have considered the vehicle to have come to a stop; thus once the vehicle stopped, it could never move.

This page intentionally left blank to facilitate printing on 2 sides.

CHAPTER 4. MODEL DYNAMIC MESSAGE SIGNS

Task 3 was to model dynamic message signs. There are two elements to dynamic message signs. The first is the sign itself, its message, and how it may be "seen." The second is the driver's reaction to the sign. Both of these are critical to modeling Dynamic Message Signs (DMS). The sign may actually be an in-vehicle message. The exact physical nature of the sign is not specified; rather, the driver's reaction to the sign should be specified. Questions to be addressed during the design of the DMS were (1) Can a probability of being seen/heard be added? and (2) Can the messages be keyed based on a sensor value (e.g., sensor XX has detected a vehicle traveling at speed SPD with acceleration/deceleration ACC)? Task 3 consisted of 2 sub-tasks.

Task 3.1 involved inputting the data and modeling DMS and In-Vehicle Driver Messaging System (IVDMS) messages. The ability to model DMS and IVDMS messages was added to the TEXAS Model.

The data specified for a message are: (a) the message type; (b) the message; (c) the message starting time; (d) the message active time; (e) the approach or intersection path that can view/receive the message, the location where the message can first be seen/received, and the location where the message can last be seen/received for each lane on the approach or intersection path where the message is visible/transmitted; (f) whether the message is for all vehicles (all message types) or a single specified vehicle number (IVDMS message types only); and (g) the message reaction time, distribution name, message mean reaction time, and other parameters needed for the message reaction time distribution.

The message types are: (1) driver DMS (external visual message to the driver for all vehicles that can see the message), (2) driver IVDMS (internal visual or auditory message to the driver), and (3) vehicle IVDMS (internal computer message to the vehicle).

The messages are: [1] accelerate or decelerate to speed XX using normal acceleration or deceleration, [2] accelerate or decelerate to speed XX using maximum vehicle acceleration or deceleration, [3] stop at the intersection (forced stop logic from Task 1.1 of this project), [4] stop at location XX (see forced stop logic from Task 1.1 of this project), [5] stop immediately using maximum vehicle deceleration (see forced stop logic from Task 1.1 of this project), [6] stop immediately using a specified collision deceleration rate XX (see forced stop logic from Task 1.1 of this project), [7] change lanes to the left, [8] change lanes to the right, [9] go (see forced go logic from Task 1.1 of this project), and [10] run the red signal (see forced run red signal logic from Task 1.2 of this project). The messages are specified in the SIMDATA file and read by SIMPRO.

Task 3.2 involved modeling the driver response to Vehicle Message System (VMS) messages. The ability to model driver response to VMS messages was added to the TEXAS Model. Each driver-vehicle unit has a flag to indicate whether the vehicle has an operational IVDMS system. For this project, this flag was set true for all driver-vehicle units. A future project may implement a mechanism to establish the value of this flag for each driver-vehicle unit. Each driver-vehicle unit has a flag to indicate whether the driver can ignore a driver DMS message type. For this project, this flag was set false for all driver-vehicle units. A future project may implement a mechanism to establish the value of this flag for each driver-vehicle unit. Each driver-vehicle unit has a flag to indicate whether the driver can ignore a driver IVDMS message type. For this project, this flag was set false for all driver-vehicle units. A future project may implement a mechanism to establish the value of this flag for each driver-vehicle unit.

A driver-vehicle unit that has an operational IVDMS system cannot ignore a vehicle IVDMS message type. For both driver IVDMS and vehicle IVDMS message types, the message can be for all vehicles or a single specified vehicle number. The affected vehicles must have an operational IVDMS system and must be within the specified start time, active time, and approach or intersection path locations. For both driver DMS and driver IVDMS message types, all messages are allowed. For vehicle IVDMS message types, the only allowed messages are [1] accelerate or decelerate to speed XX using normal acceleration or deceleration, [2] accelerate or decelerate to speed XX using maximum vehicle acceleration or deceleration, [3] stop at the intersection (forced stop logic from Task 1.1 of this project), [4] stop at location XX (see forced stop logic from Task 1.1 of this project), [5] stop immediately using maximum vehicle deceleration, and [6] stop immediately using collision deceleration.

The driver's message reaction time for both driver DMS and driver IVDMS message types will be the driver's Perception Identification Judgment and Reaction (PIJR) time minus the average PIJR time plus the random variate message reaction time using the specified message reaction time distribution and parameters. The vehicle's message reaction time for vehicle IVDMS message types will be the random variate message reaction time using the specified message reaction time distribution and parameters. The only message reaction time distribution that was developed within this project was the Normal distribution having the parameters of mean and standard deviation. Both the mean and the standard deviation may be zero. Since this is the first stochastic event to be programmed into SIMPRO, a single random number seed was written to the driver-vehicle processor file by DVPRO, read by SIMPRO, and used to generate all random numbers within SIMPRO.

For message [1] accelerate or decelerate to speed XX using normal acceleration or deceleration, the desired speed of the driver-vehicle unit was changed to XX during the message event. This message used the normal accelerate to desired speed logic within the TEXAS Model.

For message [2] accelerate or decelerate to speed XX using maximum vehicle acceleration or deceleration, the desired speed of the driver-vehicle unit was changed to XX and the maximum vehicle acceleration or deceleration was used during the message event. This message used the normal accelerate to desired speed logic within the TEXAS Model.

For message [3] stop at the intersection, this message was implemented as message [4] stop at location XX where the location is the current lane stop line or the position of the rear bumper of the nearest stopped vehicle in the current lane, whichever is closer. This message used logic similar to the forced stop logic from Task 1.1 of this project during the message event.

For message [4] stop at location XX, the location XX was specified as a positive approach number or as a negative intersection path number and the location/distance on the approach or path to stop. If the vehicle was greater than the critical stopping distance from the specified location using normal deceleration, then normal deceleration was used. Otherwise, the vehicle used the jerk rate calculated to stop the vehicle at the desired location with a limit of -15 feet (-4.572 meters) per second cubed. This message used logic similar to the forced stop logic from Task 1.1 of this project during the message event.

For message [5] stop immediately using maximum vehicle deceleration, the vehicle began the stopping action immediately after any driver reaction time and used the maximum vehicle deceleration. This message used logic similar to the forced stop logic from Task 1.1 of this project during the message event.

For message [6] stop immediately using collision deceleration, the vehicle began the stopping action immediately after any driver reaction time and used a collision deceleration rate specified by the user. This message used logic similar to the forced stop logic from Task 1.1 of this project during the message event.

For message [7] change lanes to the left and message [8] change lanes to the right, the lane change was considered as a forced lane change as opposed to an optional lane change. This message used the normal lane change logic within the TEXAS Model.

For message [9] go, the driver ignored the vehicle ahead, accelerated to desired speed, ignored sight distance restrictions, ignored intersection conflicts, was ignored by other vehicles checking intersection conflicts, and treated any signal indication as a protected green. This message used logic similar to the forced go logic from Task 1.1 of this project during the message event.

For message [10] run the red signal, the driver ignored sight distance restrictions, ignored intersection conflicts, was ignored by other vehicles checking intersection conflicts, and treated any signal indication as a protected green. This message used logic similar to the forced run red signal logic from Task 1.2 of this project during the message event.

The following COMMON BLOCK and PARAMETER variables were added:

Variable Name	Description	Type	Units	File Subroutine	COMMON Block
CVMSDN	Vehicle Message System message reaction time distribution name	Character Array	None	USER	VMSMC
DECCOL	Vehicle Message System collision deceleration maximum	Double Precision	Feet per second squared	PARAMS	Parameter
DMSI	Dynamic message sign ignore flag	Logical Array	None	VEHF	VEHFL

DVMSAT	Vehicle Message System message active time	Double Precision Array	Seconds	USER	VMSMD
DVMSDM	Vehicle Message System message reaction time distribution mean	Double Precision Array	Seconds	USER	VMSMD
DVMSDP	Vehicle Message System message reaction time distribution parameter	Double Precision Array	Variable	USER	VMSMD
DVMSMP	Vehicle Message System message parameter (speed or position)	Double Precision Array	Variable	USER	VMSMD
DVMSPB	Vehicle Message System position begin	Double Precision Array	Feet	USER	VMSMD
DVMSPE	Vehicle Message System position end	Double Precision Array	Feet	USER	VMSMD
DVMSST	Vehicle Message System message start time	Double Precision Array	Seconds	USER	VMSMD
FCVMSC	Flag for cancel Vehicle Message System change lane message	Logical	None	SIMPRO3 PREST1	Local variable
FCVMSS	Flag for cancel Vehicle Message System speed message	Logical	None	SIMPRO3 PREST1	Local variable
FNVMSC	Flag for new Vehicle Message System change lane message	Logical	None	SIMPRO3 PREST1	Local variable
FNVMSS	Flag for new Vehicle Message System speed message	Logical	None	SIMPRO3 PREST1	Local variable
IVDMSI	In-vehicle driver messaging systems ignore flag	Logical Array	None	VEHF	VEHFL
IVDMSO	In-vehicle driver messaging systems operational flag	Logical Array	None	VEHF	VEHFL
IVMSAP	Vehicle Message System approach number (positive) or intersection path number (negative)	Integer Array	None	USER	VMSMI
IVMSLB	Vehicle Message System lane begin	Integer Array	None	USER	VMSMI
IVMSLE	Vehicle Message System lane end	Integer Array	None	USER	VMSMI

IVMSMB	Vehicle Message System message beginning entry	Integer Array	None	USER	USER2
IVMSMC	Vehicle Message System message category (speed or change lane)	Integer Array	None	USER	VMSMI
IVMSMG	Vehicle Message System message	Integer Array	None	USER	VMSMI
IVMSMT	Vehicle Message System message type	Integer Array	None	USER	VMSMI
IVMSPR	Vehicle Message System message priorities by message	Integer Array	None	USER	VMSMI
IVMSVN	Vehicle Message System vehicle number (0=all)	Integer Array	None	USER	VMSMI
IVPRTV	Vehicle number for PREDTV	Integer	None	CONCHK	CONCH2
MRTIME	Vehicle Message System message reaction time	Double Precision	Seconds	SIMPRO3 PREST1	Local variable
NVMSM	Number of Vehicle Message System messages	Integer	None	USER	USER2
NVMSMM	Number of Vehicle Message System messages maximum	Integer	None	PARAMS	Parameter
NVMSMV	Number of Vehicle Message System messages for a vehicle	Integer	None	PARAMS	Parameter
POSMAX	Position maximum	Double Precision	Feet	PARAMS	Parameter
SLPCOL	Maximum slope for collisions	Double Precision	Feet per second cubed	PARAMS	Parameter
VELMAX	Velocity maximum	Double Precision	Miles per hour	PARAMS	Parameter
VMSACM	Vehicle Message System active change lane message (0=none)	Integer	None	VEHD2	VEHD
VMSACN	Vehicle Message System next change lane message (0=none)	Integer	None	VEHD2	VEHD
VMSACT	Vehicle Message System active change lane response time timer	Integer	Number of DT's	VEHD2	VEHD
VMSASM	Vehicle Message System active speed message (0=none)	Integer	None	VEHD2	VEHD
VMSASN	Vehicle Message System next speed message (0=none)	Integer	None	VEHD2	VEHD
VMSAST	Vehicle Message System active speed response time	Integer	Number of DT's	VEHD2	VEHD

	timer				
VMSCAM	Vehicle Message System category for message accelerate or decelerate to speed xx using maximum vehicle acceleration or deceleration	Integer	None	PARAMS	Parameter
VMSCAN	Vehicle Message System category for message accelerate or decelerate to speed xx using normal acceleration or deceleration	Integer	None	PARAMS	Parameter
VMSCCL	Vehicle Message System category for message change lanes to the left	Integer	None	PARAMS	Parameter
VMSCCR	Vehicle Message System category for message change lanes to the right	Integer	None	PARAMS	Parameter
VMSCGO	Vehicle Message System category for message forced go	Integer	None	PARAMS	Parameter
VMSCRR	Vehicle Message System category for message forced run the red signal	Integer	None	PARAMS	Parameter
VMSCSC	Vehicle Message System category for message stop immediately using collision deceleration	Integer	None	PARAMS	Parameter
VMSCSI	Vehicle Message System category for message stop at the intersection stop line	Integer	None	PARAMS	Parameter
VMSCSL	Vehicle Message System category for message stop at location XX	Integer	None	PARAMS	Parameter
VMSCSM	Vehicle Message System category for message stop immediately using maximum vehicle deceleration	Integer	None	PARAMS	Parameter
VMSDD	Vehicle Message System message type driver DMS	Integer	None	PARAMS	Parameter
VMSDIV	Vehicle Message System message type Driver IVDMS	Integer	None	PARAMS	Parameter
VMSMAM	Vehicle Message System message accelerate or decelerate to speed xx using	Integer	None	PARAMS	Parameter

	maximum vehicle acceleration or deceleration				
VMSMAN	Vehicle Message System message accelerate or decelerate to speed xx using normal acceleration or deceleration	Integer	None	PARAMS	Parameter
VMSMCC	Vehicle Message System message category change lanes	Integer	None	PARAMS	Parameter
VMSMCL	Vehicle Message System message change lanes to the left	Integer	None	PARAMS	Parameter
VMSMCR	Vehicle Message System message change lanes to the right	Integer	None	PARAMS	Parameter
VMSMCS	Vehicle Message System message category speed	Integer	None	PARAMS	Parameter
VMSMGO	Vehicle Message System message forced go	Integer	None	PARAMS	Parameter
VMSMRR	Vehicle Message System message forced run the red signal	Integer	None	PARAMS	Parameter
VMSMSC	Vehicle Message System message stop immediately using collision deceleration	Integer	None	PARAMS	Parameter
VMSMSI	Vehicle Message System message stop at the intersection stop line	Integer	None	PARAMS	Parameter
VMSMSL	Vehicle Message System message stop at location XX	Integer	None	PARAMS	Parameter
VMSMSM	Vehicle Message System message stop immediately using maximum vehicle deceleration	Integer	None	PARAMS	Parameter
VMSPAM	Vehicle Message System priority for message accelerate or decelerate to speed xx using maximum vehicle acceleration or deceleration	Integer	None	PARAMS	Parameter
VMSPAN	Vehicle Message System priority for message accelerate or decelerate to speed xx using normal	Integer	None	PARAMS	Parameter

	acceleration or deceleration				
VMSPCL	Vehicle Message System priority for message change lanes to the left	Integer	None	PARAMS	Parameter
VMSPCR	Vehicle Message System priority for message change lanes to the right	Integer	None	PARAMS	Parameter
VMSPGO	Vehicle Message System priority for message forced go	Integer	None	PARAMS	Parameter
VMSPOS	Vehicle Message System position behind front bumper that messages can be detected	Double Precision	Feet	PARAMS	Parameter
VMSPRR	Vehicle Message System priority for message forced run the red signal	Integer	None	PARAMS	Parameter
VMSPSC	Vehicle Message System priority for message stop immediately using collision deceleration	Integer	None	PARAMS	Parameter
VMSPSI	Vehicle Message System priority for message stop at the intersection stop line	Integer	None	PARAMS	Parameter
VMSPSL	Vehicle Message System priority for message stop at location XX	Integer	None	PARAMS	Parameter
VMSPSM	Vehicle Message System priority for message stop immediately using maximum vehicle deceleration	Integer	None	PARAMS	Parameter
VMSPST	Vehicle Message System position for stopping	Double Precision	Feet	VEHDD	VEHD
VMSTDD	Vehicle Message System message type Driver DMS	Integer	None	PARAMS	Parameter
VMSTDI	Vehicle Message System message type Driver IVDMS	Integer	None	PARAMS	Parameter
VMSTVI	Vehicle Message System message type Vehicle IVDMS	Integer	None	PARAMS	Parameter

The first line of input directly to SIMPRO (the SIMDATA file) was modified in subroutine RUSERD to add the variable NVMSM to the end of the line (columns 89–92) as a 4-digit integer with a minimum value of 0 and a maximum value of NVMSMM. The value of NVMSM was checked and an appropriate error message printed if there was an error. The value of NVMSM was printed to the output file. The value of NVMSMM was set to 100 as a parameter in file PARAMS.

The Vehicle Message System message beginning entry IVMSMB was initialized to 1 and was incremented by 1 as the Vehicle Message System message start time plus active time passed. For the second and subsequent Vehicle Message System messages, the Vehicle Message System message start time must be greater than or equal to the previous Vehicle Message System message start time; i.e., the Vehicle Message System messages must be in chronological order by start time.

An additional parameter was added, VELMAX, which is the velocity maximum, and set to 120 miles (193.12128 kilometers) per hour in file PARAMS. Also added was parameter POSMAX, which is the position maximum and was set to 1000 feet (304.8 meters) in file PARAMS. These values are used to check for input errors. Finally, an additional parameter was added, VMSPOS, which is the position behind front bumper from which Vehicle Message System messages can be detected and was set to 10 feet (3.048 meters) in file PARAMS. From a position point of view, a vehicle can detect a Vehicle Message System message from the time the front bumper of the vehicle is greater than or equal to the beginning position for the message until the time the front bumper of the vehicle plus VMSPOS is greater than the ending position for the message. Another parameter added was SLPCOL, which is the maximum slope for collisions and was set to 9660.0 feet (2944.368 meters) per second cubed in file PARAMS. The final parameter added was DECCOL, which is the maximum deceleration for collisions and was set to 966.0 feet (294.4368 meters) per second squared in file PARAMS (30 G's).

Subroutine RVMSM (read VMS messages) was written and added to file simpro2.for. This subroutine read all VMS data from input, checked for input errors, and printed each VMS message read. A call to subroutine RVMSM was added to subroutine INITIAL in file simpro2.for after the last subroutine call to read data (RDVPRD) and before the subroutine return statement.

For reading data for VMS Message, the following format was used:

Field	Description	Start Col	Num Cols	Data Type	Units	Variable Name
1	Vehicle Message System message type	1	4	Integer	None	IVMSMT(I)
2	Vehicle Message System message	5	4	Integer	None	IVMSMG(I)
3	Vehicle Message System message parameter (speed or position)	9	7	Double Precision	Variable	DVMSMP(I)
4	Vehicle Message System message start time	16	6	Integer	Hundredths of a second	DVMSST(I)
5	Vehicle Message System message active time	22	6	Integer	Hundredths of a second	DVMSAT(I)
6	Vehicle Message System approach number (positive) or intersection path number (negative)	28	4	Integer	None	IVMSAP(I)

7	Vehicle Message System lane begin	32	2	Integer	None	IVMSLB(I)
8	Vehicle Message System lane end	34	2	Integer	None	IVMSLE(I)
9	Vehicle Message System position begin	36	7	Double Precision	Feet	DVMSPB(I)
10	Vehicle Message System position end	43	7	Double Precision	Feet	DVMSPE(I)
11	Vehicle Message System vehicle number (0=all)	50	6	Integer	None	IVMSVN(I)
12	Vehicle Message System message reaction time distribution name	56	7	Character	None	CVMSDN(I)
13	Vehicle Message System message reaction time distribution mean	63	6	Double Precision	Seconds	DVMSDM(I)
14	Vehicle Message System message reaction time distribution parameter	69	6	Double Precision	Variable	DVMSDP(I)

The tests performed on the input data follows:

1. NVMSM must be greater than or equal to 0 and less than or equal to NVMSMM; STOP 943.
2. IVMSMT(I) must be equal to VMSTDD, VMSTDI, or VMSTVI; STOP 945.
3. IVMSMG(I) must be equal to VMSMAN, VMSMAM, VMSMSI, VMSMSL, VMSMSM, VMSMSC, VMSMCL, VMSMCR, VMSMGO, or VMSMRR; STOP 946.
4. If IVMSMT(I) is equal to VMSTVI, then IVMSMG(I) must not be equal to VMSMCL, VMSMCR, VMSMGO, or VMSMRR; STOP 947.
5. If IVMSMG(I) is equal to VMSMSC, then DVMSMP(I) must be greater than or equal to -DECCOL and less than 0.0; STOP 948.
6. If IVMSMG(I) is equal to VMSMAN or VMSMAM, then DVMSMP(I) must be greater than or equal to 0.0 or less than or equal to VELMAX; STOP 949.
7. If IVMSMG(I) is equal to VMSMSL, then DVMSMP(I) must be greater than or equal to 0.0 and less than or equal to POSMAX; STOP 950.
8. DVMSST(I) must be greater than or equal to 0.0 and less than or equal to SIMTIM; STOP 951.
9. If I is greater than 1, then DVMSST(I) must be greater than or equal to DVMSST(I-1); STOP 952.
10. DVMSAT(I) must be greater than or equal to 0.0; STOP 953.
11. If IVMSAP(I) is greater than or equal to 0, then IVMSAP(I) must equal one of the approach numbers; STOP 954.
12. If IVMSAP(I) is less than 0, then IVMSMG(I) must not be VMSMCL or VMSMCR; STOP 955.

13. If IVMSAP(I) is less than 0, then -IVMSAP(I) must equal one of the intersection path numbers; STOP 956.
14. If IVMSAP(I) is greater than or equal to 0, then IVMSLB(I) must be greater than or equal to 1 and less than or equal to the number of lanes for the approach; STOP 957.
15. If IVMSAP(I) is greater than or equal to 0, then IVMSLE(I) must be greater than or equal to 1 and less than or equal to the number of lanes for the approach; STOP 958.
16. If IVMSAP(I) is greater than or equal to 0, then IVMSLE(I) must be greater than or equal to IVMSLB(I); STOP 959.
17. If IVMSAP(I) is greater than or equal to 0 and IVMSMG(I) is equal to VMSMCL or VMSMCR, then the number of lanes for the approach must be greater than or equal to 2; STOP 960.
18. DVMSPB(I) must be greater than or equal to 0.0 and less than or equal to the maximum lane length for the approach or the intersection path length; STOP 961.
19. DVMSPE(I) must be greater than or equal to DVMSPB(I) and less than or equal to the maximum lane length for the approach or the intersection path; STOP 962.
20. IVMSVN(I) must be greater than or equal to 0; STOP 963.
21. CVMSDN(I) must be equal to "NORMAL"; STOP 964.
22. DVMSDM(I) must be greater than or equal to 0.0; STOP 965.

The logical variable IVDMSO was added for each driver-vehicle unit to indicate whether the vehicle has an operational IVDMS system. For this project, this flag was set true for all driver-vehicle units in subroutine LOGIN. A future project may implement a mechanism to establish the value of this flag for each driver-vehicle unit. The logical variable DMSI was added for each driver-vehicle unit to indicate whether the driver can ignore a driver DMS message type. For this project, this flag was set false for all driver-vehicle units in subroutine LOGIN. A future project may implement a mechanism to establish the value of this flag for each driver-vehicle unit. The logical variable IVDMSI was added for each driver-vehicle unit to indicate whether the driver can ignore a driver IVDMS message type. For this project, this flag was set false for all driver-vehicle units in subroutine LOGIN. A future project may implement a mechanism to establish the value of this flag for each driver-vehicle unit.

The logic to determine whether a Vehicle Message System message is or should be active for a vehicle was added to subroutine PREST1 in file simpro3.for. This subroutine was called for each vehicle every DT very early on in the processing for all vehicles. The following is the general processing sequence added to PREST1:

1. If there were no VMS messages read (NVMSM equal to 0), then skip to the end.
2. Set the temporary active change lane message ACMVMS to the current active change lane message VMSACM(IV); if none, then set to the pending change lane message VMSACN(IV).
3. Set the temporary active speed message ASMVMS to the current active speed message VMSASM(IV); if none, then set to the pending speed message VMSASN(IV).
4. Set the temporary current change lane message CCMVMS to zero.
5. Set the temporary current speed message CSMVMS to zero.
6. Process all VMS messages starting with IVMSMB.

a. If the current time TIME is less than the Vehicle Message System message start time DVMSST(I), then ignore the message and ignore all remaining Vehicle Message System messages.
b. If the current time TIME is greater than the Vehicle Message System message start time plus active time (DVMSST(I)+DVMSAT(I)), then ignore the message and check the next message.
c. If the Vehicle Message System message type IVMSMT(I) is Driver DMS VMSTDD and the driver-vehicle unit can ignore DMS messages DMSI(IV), then ignore the message and check the next message.
d. If the Vehicle Message System message type IVMSMT(I) is Driver IVDMS VMSTDI and the driver-vehicle unit can ignore IVDMS messages IVDMSI(IV), then ignore the message and check the next message.
e. If the Vehicle Message System message type IVMSMT(I) is Driver IVDMS VMSTDI and the driver-vehicle unit does not have an operational IVDMS IVDMSO(IV), then ignore the message and check the next message.
f. If the Vehicle Message System message type IVMSMT(I) is Vehicle IVDMS VMSTVI and the driver-vehicle unit does not have an operational IVDMS IVDMSO(IV), then ignore the message and check the next message.
g. If the Vehicle Message System approach number or the intersection path number IVMSAP(I) is not equal to the current approach number IA or intersection path number IP, then ignore the message and check the next message.
h. If the Vehicle Message System intersection path number IVMSAP(I) is equal to the current intersection path number IP and if the Vehicle Message System message IVMSMG(I) is change lanes to the left VMSMCL or change lanes to the right VMSMCR, then ignore the message and check the next message.
i. If the Vehicle Message System approach number IVMSAP(I) is equal to the current approach number IA and if the current lane number ILN is not between the Vehicle Message System beginning lane number IVMSLB(I) and the ending lane number IVMSLE(I), then ignore the message and check the next message.
j. If the Vehicle Message System approach number IVMSAP(I) is equal to the current approach number IA and Vehicle Message System message is change lanes to the left VMSMCL and there is lane to the left, then ignore the message and check the next message.
k. If the Vehicle Message System approach number IVMSAP(I) is equal to the current approach number IA and Vehicle Message System message is change lanes to the right VMSMCR and there is no lane to the right, then ignore the message and check the next message.
l. If the current position of the front bumper of the vehicle IPOS(IV) is less than the Vehicle Message System beginning position DVMSPB(I), then ignore the message and check the next message.
m. If the current position of the position behind the front bumper of the vehicle that the Vehicle Message System messages can be detected (IPOS(IV)-VMSPOS) is greater than the Vehicle Message System ending position DVMSPE(I), then ignore the message and check the next message.
n. If (1) the Vehicle Message System message type IVMSMT(I) is Driver IVDMS VMSTDI or Vehicle IVDMS VMSTVI, (2) the Vehicle Message System message

vehicle number is greater than zero, and (3) the Vehicle Message System message vehicle number is not the current vehicle number, then ignore the message and check the next message.

   o. The Vehicle Message System message applies to this vehicle and thus set the temporary current change lane message CCMVMS to the highest-priority (1=highest) lane change message or set the temporary current speed message CSMVMS to the highest-priority (1=highest) speed message.

7. Based upon the priority of the current and active Vehicle Message System change lane message, determine whether to (a) initiate current message, (b) continue active message, (c) cancel active message, or (d) continue with no message.
8. Based upon the priority of the current and active Vehicle Message System speed message, determine whether to (a) initiate current message, (b) continue active message, (c) cancel active message, or (d) continue with no message.
9. If required, process cancel active Vehicle Message System change lane message.
10. If required, process initiate current Vehicle Message System change lane message, calculate message reaction time MRTIME, and set the change lane message reaction time counter VMSACT(IV) to the message reaction time MRTIME converted to number of DT's plus 1.
11. If Vehicle Message System change lane message reaction time counter VMSACT(IV) is greater than zero, then decrement change lane message reaction time counter VMSACT(IV), and if zero, then if the vehicle is currently changing lanes, then set the change lane message reaction time counter VMSACT(IV) to one and continue else start the current lane change message and set flags for a forced lane change to the left or the right depending on the Vehicle Message System message.
12. If required, process cancel active Vehicle Message System speed message.
13. If required, process initiate current Vehicle Message System speed message, calculate message reaction time MRTIME, and set the speed message reaction time counter VMSAST(IV) to the message reaction time MRTIME converted to number of DT's plus 1.
14. If Vehicle Message System speed message reaction time counter VMSAST(IV) is greater than zero, then decrement change speed reaction time counter VMSAST(IV) and if zero, then start the current speed message.
15. If there is a current Vehicle Message System speed message, then set vehicle parameters based upon the Vehicle Message System speed message.

There can be only one active lane-change-related Vehicle Message System message. Only the highest-priority lane-change-related Vehicle Message System message was processed in the priority order in the next table. If there was no existing lane-change-related Vehicle Message System message, the message reaction time for the new message was calculated and started from the beginning, no lane-change-related Vehicle Message System message was processed during the message reaction time for the new message, and the new message was activated upon completion of the message reaction time for the new message. If an existing lane-change-related Vehicle Message System message was preempted by a higher-priority lane-change-related Vehicle Message System message, the message reaction time for the new message was calculated and started from the beginning, the existing lane-change-related Vehicle Message System message was continued during the message reaction time for the new message, and the new message was activated upon completion of the message reaction time for the new message.

There can be only one active speed-related Vehicle Message System message. Only the highest-priority speed-related Vehicle Message System message was processed in the priority order in the next table. If there was no existing speed-related Vehicle Message System message, the message reaction time for the new message was calculated and started from the beginning, no speed-related Vehicle Message System message was processed during the message reaction time for the new message, and the new message was activated upon completion of the message reaction time for the new message. If an existing speed-related Vehicle Message System message was preempted by a higher-priority speed-related Vehicle Message System message, the message reaction time for the new message was calculated and started from the beginning, the existing speed-related Vehicle Message System message was continued during the message reaction time for the new message, and the new message was activated upon completion of the message reaction time for the new message.

Priority	Priority Variable Name	Message Variable Name	Description
1	VMSPSC	VMSMSC	Vehicle Message System message stop immediately using collision deceleration
2	VMSPSM	VMSMSM	Vehicle Message System message stop immediately using maximum vehicle deceleration
3	VMSPSL	VMSMSL	Vehicle Message System message stop at location XX
4	VMSPSI	VMSMSI	Vehicle Message System message stop at the intersection stop line
5	VMSPRR	VMSMRR	Vehicle Message System message forced run the red signal
6	VMSPGO	VMSMGO	Vehicle Message System message forced go
7	VMSPAM	VMSMAM	Vehicle Message System message accelerate or decelerate to speed xx using maximum vehicle acceleration or deceleration
8	VMSPAN	VMSMAN	Vehicle Message System message accelerate or decelerate to speed xx using normal acceleration or deceleration
9	VMSPCL	VMSMCL	Vehicle Message System message change lanes to the left
10	VMSPCR	VMSMCR	Vehicle Message System message change lanes to the right

The following changes were made in the TEXAS Model to correct errors discovered during the performance of Task 3. These errors were hindering the evaluation of the changes made for Task 3. These changes resulted in reducing the “intrusions” for the twenty example runs from approximately 120 to 5, reducing total delay, and slightly increasing queue and stopped delay. Vehicles generally approach the intersection at higher speeds (speeds closer to their desired speed) thus they arrive at the end of the queue sooner, and therefore they encounter more queue and stopped delay.

1. In all subroutines the test for whether a lane change is in progress was changed from LCHGE equal to 2 to ISET equal to 1. Previously LCHGE could be set to 3 (follow lane changing vehicle) while a lane change is in progress, thus invalidating the test. ISET is not modified while the lane change is in progress.
2. In all subroutines the test to determine whether it was the first DT for a lane change to the right for specifying the correct lateral position was not needed when the present link LPRES was used; therefore the code was removed.
3. In subroutine LOKIOB the variable IA in common block INDEX2 in file INDEX was set to the approach number for the linking inbound lane for the intersection path IP [ISNA(LIBL(IP))] after the fourth executable statement in the subroutine and immediately before the first use of IA. Previously, the subroutine used the last set value of IA, which was not correct.
4. In subroutine LOKFMR a check was added in the DO 1010 loop and in the DO 1020 loop to skip the KVPV vehicle if it was the current vehicle IV. Previously, the subroutine could have looked ahead and found itself.
5. In subroutine IBAP between DO 4010 and 1026 CONTINUE, a logical variable LTPACT was set true or false if a left turn pull out is allowed for this vehicle.
6. In subroutine IBAP between 1026 CONTINUE and 1030 CONTINUE, the condition that the previous vehicle position PVPOS be less than or equal to the current position of the vehicle IPOS(IV) minus the length of the vehicle LENV(IVEHCL(IV)) was imposed on incrementing the previous vehicle position PVPOS by the length of the current lane ENDLN minus 1.5 feet. Previously, it was incorrectly assumed that the previous vehicle was on an intersection path; thus the previous vehicle position needed to be incremented.
7. In subroutine IBAP between 1035 CONTINUE and 1040 CONTINUE, a vehicle was additionally allowed to look ahead into the linking intersection path by calling LOKIBI if (a) this vehicle may not proceed into the intersection, (b) a left turn pull out is allowed for this vehicle, (c) there is a vehicle on the linking intersection path for this vehicle, and (d) the position of the rear bumper of the last vehicle on the linking intersection path for this vehicle is less than the current end of the lane (the current end of the lane has been extended because a left turn pull out is allowed for this vehicle). Previously, this look ahead was not allowed and this vehicle could have a collision with the last vehicle on the linking intersection path for this vehicle because it did not know that it was there until it gained the right to enter the intersection.
8. In subroutine IBAP between 2020 CONTINUE and 2030 CONTINUE before LALT is used, if LALT is greater than or equal to 5, then a call was added to subroutine CKLALT to check the lane alternatives and set correct values for LALT. Previously, this was not done and therefore the statements testing for specific values of LALT did not work correctly.
9. In subroutine IBAP between 2050 CONTINUE and 2060 CONTINUE, the condition that the vehicle’s new velocity VELNEW be greater than the velocity considered to be stopped VELSTP and that the vehicle was not decelerating to a stop [MSFLG(IV) false] was added to the condition that the vehicle has the right to proceed into the intersection MPRO(IV) when deciding whether to call subroutine AVDCON.

Previously, a stopped vehicle that should remain stopped could be caused to accelerate by AVDCON and thus have a collision with the vehicle ahead.

10. In subroutine IBAP between 2080 CONTINUE and 2090 CONTINUE, if the current vehicle does have an intersection path [LNEXT(IV).NE.0], if there is a linking outbound lane for the intersection path [LOBL(LNEXT(IV)).NE.0], if there is a last vehicle on the linking outbound lane for the intersection path [ILVL(LOBL(LNEXT(IV))).NE.0], if the position of the rear bumper of the last vehicle on the linking outbound lane for the intersection path POSR is less than or equal to the new position of the front bumper of the current vehicle POSNEW and if the velocity of the current vehicle VELNEW is less than or equal to 0.0, then the current vehicle is not allowed to log out of the inbound approach and lane and into the linking intersection path. Previously, a right-turning vehicle on a short intersection path could have a collision with a long straight-through vehicle that just entered the linking outbound lane for the intersection path because the long vehicle cleared its intersection conflicts when it logged onto the linking outbound lane and was not visible in intersection conflict checking.
11. In subroutine CHKDSP before 1010 CONTINUE, the value of the time required to reduce the present velocity of the vehicle to the desired speed of the intersection path using SLOPE [T] was maxed with 0.001 seconds. Previously, a negative or zero value of time could have incorrectly been used to find the distance required to reduce the present velocity of the vehicle to the desired speed of the intersection path using SLOPE [XCRIT].
12. In subroutine LOGIBI the statement 2020 CONTINUE was moved from after the call to SETCON to before the call to SETCON, resulting in SETCON being called under all circumstances. Previously under certain circumstances, SETCON was not being called; therefore other vehicles did not see the vehicle when performing intersection conflict checking and intersection conflict avoidance.
13. In subroutine UNBIAS after 1030 CONTINUE, the maximum value for the time required to bring the vehicle to a stop within this DT [T] was set to 1.001*DT. Previously, the maximum value was 2.0*DT, which was incorrect.
14. In subroutine NEWVEL before 1010 CONTINUE, the logic for implementing the slope for intersection conflicts SLPCON was changed from simply adding SLPCON to SLPNEW to (a) if SLPNEW is greater than 0.0 and SLPCON is greater than 0.0, then add SLPCON to SLPNEW, (b) if SLPNEW is greater than 0.0 and SLPCON is less than or equal to 0.0, then set SLPNEW to SLPCON, (c) if SLPNEW is less than or equal to 0.0 and SLPCON is greater than 0.0, then add 0.25*SLPCON to SLPNEW, and (d) if SLPNEW is less than or equal to 0.0 and SLPCON is less than or equal to 0.0, then add SLPCON to SLPNEW. Previously, the simple addition did not work adequately in all situations.
15. In subroutine ENDLCH before LNEXT was set to 0, if LNEXT was not equal to 0, then a call was made to subroutine UNSETC. Previously under unique situations, intersection conflicts were not unset when a lane change was completed, causing other vehicles to check against an invalid vehicle or a vehicle that had logged out of the system.
16. In subroutine GAPACC between 5030 CONTINUE and 5040 CONTINUE before LALT is used, if LALT is greater than or equal to 5, then a call was added to

subroutine CKLALT to check the lane alternatives and set correct values for LALT. Previously, this was not done and therefore the statements testing for specific values of LALT did not work correctly.

17. In subroutine ACDCP between 7010 CONTINUE and 7020 CONTINUE, when calculating the time required to bring the vehicle to a stop within this DT, (a) the test for SLPNEW equal to 0.0 to GO TO 7020 was changed to DABS(SLPNEW) less than or equal to 0.001 to GO TO 7020, (b) a test for DABS(RADICL) less than or equal to 0.001 to GO TO 7020 was added, and (c) a test for T less than or equal to 0.0 or T greater than DT to GO TO 7020 was added. Previously, small values of SLPNEW could produce values of T, which were incorrect and cause the vehicle's position to be incorrectly calculated.
18. In subroutine ACDCP between 7020 CONTINUE and 7030 CONTINUE, when calculating the time required to bring the vehicle to a stop within this DT, a test for T less than or equal to 0.0 or T greater than DT to GO TO 7040 was added. Previously, small values of ACCOLD could produce values of T, which were incorrect and cause the vehicle's position to be incorrectly calculated.
19. In subroutine CARFOL between 2010 CONTINUE and 2020 CONTINUE, when there is a vehicle ahead, added that when the previous vehicle's slope PVSLP is greater than or equal to 0.0 and the previous vehicle's velocity PVVEL is less than or equal to 2.0, then set the position of the vehicle ahead when stopped PVSTP to the previous vehicle's current position PVPOS. Previously, when the vehicle ahead was stopped, causing the current vehicle to calculate a deceleration to a stop behind the vehicle ahead, and the current vehicle was almost stopped behind the vehicle ahead and then the vehicle ahead started accelerating, the car following logic could not decelerate adequately; thus a collision with the vehicle ahead occurred.
20. In subroutine CARFOL between 2010 CONTINUE and 2020 CONTINUE, when there is a vehicle ahead and a slope required to stop SLPSTP value has been calculated, changed the test from SLPSTP greater than 0.0 to GO TO 2020 to SLPSTP greater than or equal to 0.0 to GO TO 2020. Previously, a zero value of SLPSTP could cause SLPNEW to be set to zero incorrectly.
21. In subroutine CARFOL between 2010 CONTINUE and 2020 CONTINUE, when there is a vehicle ahead and a slope new SLPNEW value has been calculated based upon stopping behind where the vehicle ahead stops, change the minimum value of SLPNEW from 0.50*FACT*CRISLP to 1.3*FACT*CRISLP [FACT ranges from -1.0 to -2.0]. Previously, slope new SLPNEW could not be set to a high negative number; thus vehicles could not decelerate adequately and thus a collision with the vehicle ahead occurred.
22. In subroutine CARFOL between 2010 CONTINUE and 2020 CONTINUE, when calculating the time required to bring the vehicle ahead to a stop T, (a) added the calculation of the maximum time to stop TSMAX as 1.5 times the previous vehicle's velocity divided by 11.2 feet per second squared - recommended by the American Association of State Highway and Transportation Officials (AASHTO) - plus the PIJR time for the current driver, (b) set the minimum value for TSMAX to 3.0 seconds, and (c) added a check for T greater than or equal to TSMAX to GO TO 2020.

23. In subroutine CARFOL between 2010 CONTINUE and 2020 CONTINUE, when calculating the time required to bring the current vehicle to a stop TS, (a) added the calculation of the maximum time to stop TSMAX as 1.5 times the current vehicle's velocity divided by 11.2 feet per second squared (recommended by AASHTO) plus the PIJR time for the current driver, (b) set the minimum value for TSMAX to 3.0 seconds, and (c) added a check for TS greater than or equal to TSMAX to GO TO 2020.
24. In subroutine CARFOL between 2020 CONTINUE and 2025 CONTINUE, when a slope new SLPNEW value has been calculated by the traditional car following equation, change the minimum value of SLPNEW from 0.50*FACT*CRISLP to 0.65*FACT*CRISLP [FACT ranges from -1.0 to -2.0]. Previously, slope new SLPNEW could not be set to a high negative number; thus vehicles could not decelerate adequately and thus a collision with the vehicle ahead occurred.
25. In subroutine ACCEL after the IF statement with GO TO 4010 and before 1010 CONTINUE, made setting slope new SLPNEW to 0.0, acceleration/deceleration old ACCOLD to 0.0, and velocity old VELOLD to the desired velocity DESVEL conditional upon the absolute value of the velocity old VELOLD minus the desired velocity DESVEL being less than or equal to 0.1 feet per second or else setting the slope new SLPNEW to the acceleration/deceleration slope required to bring the vehicle's velocity to his desired speed in one DT [SLPNEW = (DESVEL-VELOLD-ACCOLD*DT)/(0.5*DTSQ)] and limiting the slope new SLPNEW to a minimum of -CRISLP and a maximum of CRISLP. Previously, when the speed was greater than the desired velocity minus 0.5 feet per second, the speed was less than or equal to the desired velocity plus 1.0 feet per second, and the absolute value of the acceleration/deceleration old ACCOLD was less than 0.5*CRISLP, the vehicle speed was set to the desired speed. This change tightened the criteria for taking this action. When these tightened criteria were not met, then the slope was set so the desired speed would be achieved the next DT. This action more accurately reflects the dynamics to reach desired speed.
26. In subroutine ACCEL between 2020 CONTINUE and 2030 CONTINUE, the logic to find the acceleration/deceleration slope new SLPNEW required to bring the acceleration/deceleration new ACCNEW to zero by the time the vehicle's velocity reaches his desired speed was modified as follows:

```
 2025 CONTINUE
C-----FIND THE ACC/DEC SLOPE REQUIRED TO BRING THE ACC/DEC TO ZERO BY
C-----THE TIME THE VEHICLES VELOCITY REACHES HIS DESIRED SPEED
      ACCTMP = ACCOLD + SLPNEW*DT
      VELTMP = VELOLD + ACCOLD*DT + 0.5D0*SLPNEW*DTSQ
      IF ( VELTMP . EQ . DESVEL )                 GO TO 2030
      SLOPE = -0.5D0*ACCTMP**2/(DESVEL-VELTMP)
      IF ( SLOPE . NE . 0.0D0 )                   THEN
        IF ( ( SLOPE . LT . -0.8D0*CRISLP ) . OR .
     *         ( SLOPE . GT .  0.8D0*CRISLP ) )   GO TO 2027
        T = -ACCTMP/SLOPE
        IF ( T . LT . DT )                        THEN
C-----     FIND THE ACC/DEC SLOPE REQUIRED TO BRING THE VEHICLES VELOCITY
C-----     TO HIS DESIRED SPEED IN ONE DT
          SLPNEW = (DESVEL-VELOLD-ACCOLD*DT)/(0.5*DTSQ)
          GO TO 2030
        END IF
C-----  IF THE TIME TO BRING THE ACC/DEC TO ZERO BY THE TIME THE
```

```
C-----  VEHICLES VELOCITY REACHES HIS DESIRED SPEED IS GREATER THAN THE
C-----  DRIVERS PIJR TIME THEN CONTINUE USING SLPNEW
          IF ( T . GT . PIJR(IDRICL(IV))+DT )      GO TO 2030
        END IF
 2027 CONTINUE
        IF ( VELOLD . EQ . DESVEL )                THEN
          SLPNEW = -ACCOLD/DT
          GO TO 2030
        END IF
        SLOPE = -0.5D0*ACCOLD**2/(DESVEL-VELOLD)
```

This new logic does a superior job of determining when the vehicle should reverse the slope to bring the acceleration/deceleration to zero when the speed reaches the desired speed, whether the vehicle is above or below its desired speed. Additionally, this new logic calculates the slope new SLPNEW required to bring the vehicle's velocity to the desired speed in one DT when the estimated time is less than one DT without regard to the final value of acceleration/deceleration. Previously, the acceleration/deceleration at the beginning of the DT [ACCOLD] and the velocity at the beginning of the DT [VELOLD] were used, and it was too late to start the reversal.

27. In subroutine ACCEL between 3020 CONTINUE and 4010 CONTINUE, the minimum value for the time to reduce the acceleration/deceleration old ACCOLD to zero using SLOPE [T] was changed to 0.001 second from 0.01 second, and if the time to reduce the acceleration/deceleration old ACCOLD to zero using SLOPE [T] is less than DT, then GO TO 2025. Previously, a larger minimum value was allowed, leading to inaccurate calculations.
28. In subroutine CRIDIS between 3010 CONTINUE and 3020 CONTINUE, (a) calculated the maximum time to stop TSMAX as 1.2 times the sum of the time to stop from the old velocity OLDVEL using 11.2 feet/second/second deceleration rate as recommended by AASHTO plus the driver's PIJR time; (b) if the calculated time to stop TS is greater than the maximum time to stop TSMAX, then call ACCEL to accelerate the vehicle to its desired speed; and (c) if the old acceleration/deceleration is not equal to 0.0 and a deceleration to a stop can not be calculated, then call ACCEL to accelerate the vehicle to its desired speed. Previously, the vehicle's acceleration/deceleration would be reduced to zero, which would mean the vehicle's speed would be constant and possibly low. This action allows the vehicle to speed up when it is not close enough to the object ahead, requiring a stop to calculate a deceleration to a stop.
29. In subroutine CHKCON between 1090 CONTINUE and 1100 CONTNUE, an additional test if NOFC was equal to 0 to GO TO 3010. Previously, NOFC could be equal to 0; thus the statement NORC(NOFC) resulted in an invalid array reference.
30. In subroutine PREDTV the code was updated to mimic code from subroutine CHKDSP, ACCEL, and ACDCP as modified.
31. In subroutine PATHF between 4010 CONTINUE and 4020 CONTINUE before LNEXT was set to LPATH, if LNEXT was not equal to 0, then a call was made to subroutine UNSETC. Previously under unique situations, intersection conflicts were not unset when a new intersection path was chosen causing other vehicle to check against an invalid vehicle or a vehicle that had logged out of the system.

32. In subroutine PATHF between 4010 CONTINUE and 4020 CONTINUE, variable INT2P in common block VEHF2 in file VEHF was set to zero after the intersection path for the vehicle was changed because it could not get into a lane that had an intersection path to its desired outbound approach. Previously, this variable was not re-set causing an error.
33. In subroutine BANGS the collision was dismissed if the previous vehicle IVPV was not the actual vehicle ahead of the current vehicle. Previously, a vehicle sometimes sets the vehicle ahead to a vehicle in the adjacent lane because it needs to car follow that vehicle and the vehicle in the adjacent lane is closer to the current vehicle than the actual vehicle ahead. In the process of responding to this vehicle in an adjacent lane, a collision might seem to occur, but the vehicles are in different lanes or intersection paths; therefore no real collision occurred, and therefore the collision can be dismissed.
34. In subroutine BANGS between 6010 CONTINUE and 9190 CONTINUE, after a collision, a vehicle's new position POSNEW is set to the previous vehicle's position PVPOS minus 0.1 feet rather than minus 2.0 feet. Previously, a vehicle's re-positioning after a collision with the vehicle ahead caused another collision with the vehicle behind the current vehicle and this effect could ripple through the entire queue of stopped vehicles waiting to enter the intersection.

CHAPTER 5. MODEL ENHANCED ITS SENSORS

Task 4 was to model enhanced ITS sensors. There are several elements to modeling ITS sensors. The first is the sensor itself and where it is located. The second is the field of view of the sensor - where on the roadway it can detect vehicles. The third is the nature of the field of view - can it detect presence, speed, vehicle class, acceleration, etc.? The fourth is the ability of the TEXAS Model to provide vehicle class information to the sensors. Task 4 consisted of two sub-tasks.

Task 4.1 was to modify the TEXAS Model to add additional vehicle types. The previous default number of vehicle types was twelve and the maximum number of vehicle types was fifteen. The parameters for each vehicle type is maximum acceleration IAMAX in feet per second squared, maximum deceleration IDMAX in feet per second squared, minimum turning radius IRMIN in feet, vehicle characteristic IVCHAR in percent, maximum velocity IVMAX in feet per second, vehicle length LENV in feet, and the percentages of each driver class in each vehicle class XPERD.

Task 4.2 was to implement additional sensor types. The TEXAS Model previously allowed a total of thirty detectors with a maximum of six detectors per lane. A single detector may be in one or more contiguous lanes. For each detector the user specified the type (pulse or presence), the approach number, the starting position, stopping position, the number of lanes, and the list of lanes. Each detector must be connected to at least one signal phase in a positive manner (not a NOT connection) and may be connected to as many signal phases as are necessary. Each vehicle on an inbound approach checked each DT to determine whether the front bumper of the vehicle crossed the start of the detector (LDCROS) and whether the vehicle had tripped the detector (LDTRIP). LDCROS and LDTRIP were logical variables; thus they were true or false and did not count the number of crosses and trips during a single DT. A pulse detector was only tripped for the DT when the front bumper of the vehicle crossed the start of the detector. A presence detector was tripped for each DT, starting when the front bumper of the vehicle crossed the start of the detector and continuing through when the rear bumper of the vehicle crossed the end of the detector.

Advanced ITS sensors generally have a starting and ending position on a lane or adjacent lanes where they measure presence and other traffic parameters. For the current vehicles detectors in presence mode, the detector shows presence from the time when the front bumper of the vehicle crosses the leading edge of the detector until the rear bumper of the vehicle crosses the trailing edge of the detector. Advanced ITS sensors could be based on SIMPRO's knowing the vehicle's position, velocity, acceleration/deceleration, jerk, length, lateral position if lane changing, vehicle type, driver type, signal indications, and similar information for the vehicle ahead of the vehicle, but this would give advanced information that detectors simply cannot measure as accurately. The mechanism of electronically or physically determining this information is generally less important. What seems important is that they do provide the number of vehicles, measure volume, measure speed and occupancy, measure density, measure headway, and classify vehicles. Many of the ITS sensors can measure both positive and negative acceleration. Video sensors have the theoretical capability of measuring heading, allowing sensing of when vehicles are making lane changes or turning movements. There is a desire to simulate the output of the ITS sensors using actual or estimated confusion matrices and variation distributions without going into the elaborate algorithms used by each company. This is particularly important since some companies may not have or may not be willing to provide these matrices and distributions. Part of the difference is that SIMPRO will know the exact speed at which a vehicle travels over the detector, whereas measuring speed using two sequential detectors has inaccuracies, and the information is not available until some time after the vehicle crosses the second detector.

The TEXAS Model was modified to incorporate an advanced ITS classification sensor. Several advanced ITS detectors were investigated to determine how they operated and the availability of their algorithms. No companies contacted would release their propriety algorithms.

To complete Task 4.1, the following actions were performed.

For programs DVPRO, GDVCONV, and GEOPRO in subroutine GDVCON from TEXASLIB:

1. Changed all occurrences of NDC to NDRICL.
2. Changed all occurrences of NVC to NVEHCL.
3. Added a parameter for the number of driver classes NDC.
4. Set the value of NDC to 9.
5. Added a parameter for the number of vehicle classes NVC.
6. Set the value of NVC to 99.
7. Changed all relevant values of 3 or 5 to NDC (there were none).
8. Changed all relevant values of 12 or 15 to NVC.
9. Modified the record number for reading the traffic destination data to allow for INT((NVEHCL+14)/15) number of traffic mix data records instead of just 1 traffic mix data record for an approach/leg.
10. Changed the format of the traffic mix data from "15F3.0" to "15F5.1" in the GDVDATA file.
11. Changed the reading of traffic mix data records from 1 record to INT((NVEHCL+14)/15) records while reading up to 15 values per record using the "15F5.1" format.

For program DVPRO:

1. Added a parameter for the number of driver classes for default NDCD in file PARAMS.
2. Set the value to 3 in file PARAMS.
3. Changed all relevant values of 3 or NDC to NDCD.

If the specified number of driver classes was 0 or 3, DVPRO set default values for NDCD driver parameters; otherwise the user must specify all driver parameters for the specified number of drivers. The user was not allowed to have fewer than NDCD drivers and was not allowed to change any of the parameters for the first NDCD drivers.

For program DVPRO:

1. Added a parameter for the number of vehicle classes for default NVCD in file PARAMS.
2. Set the value to 12 in file PARAMS.
3. Changed all relevant values of 12 or NVC to NVCD.

If the specified number of vehicle classes was 0 or 12, DVPRO set default values for NVCD vehicle parameters; otherwise the user must specify all vehicle parameters for the specified number of vehicles. The user was not allowed to have fewer than NVCD vehicles and was not allowed to change any of the parameters for the first NVCD vehicles because some programs use knowledge of the default vehicle classes' vehicle classification (auto, truck, etc.).

For programs DVPRO, GEOPRO (specified but not referenced), SIMPRO, and SIMSTA (specified but not referenced), the value of the parameter for the number of driver classes NDC was changed from 3 to 9. In programs DISPRE_J, DVPRO, EMPRO, GEOPRO (specified but not referenced), SIMPRO, and SIMSTA (specified but not referenced), the value of the parameter for the number of vehicle classes NVC was changed from 12 to 99. In addition, all source code files were checked to ensure that the number 3 was not referring to the number of driver classes and that the number 12 was not referring to the number of vehicle classes; no occurrences were found.

For program GEOPRO in subroutine READIO:

1. Added reading the number of vehicle classes NVEHCL and the number of driver classes NDRICL from the appropriate record.
2. Checked to ensure that NVEHCL was greater than or equal to NVCD and less than or equal to NVC (STOP 866).
3. Checked to ensure that NDRICL was greater than or equal to NDCD and less than or equal to NDC (STOP 867).

For program GEOPRO, in subroutine READAP, the reading of the percentage of each vehicle class making up the traffic stream was modified from reading 1 record to reading INT((NVEHCL+14)/15) records.

For program DISPRE_J, the reading of the vehicle lengths was changed from 1 record to approximately INT((NVEHCL+19)/20) records. The number of vehicle classes values and up to the first 19 vehicle length values were read from the first record using a "(I4,19F4.0)" format, then up to 20 vehicle lengths were read for the second and subsequent records as needed using a "(20F4.0)" format.

For Task 4.2, the following equations demonstrate the procedure to calculate velocity, acceleration/deceleration, and jerk from position.

Assume the basic equation of motion used in the TEXAS Model:
P(T+DT) = P(T) + V(T)*DT + A(T)*DT*DT/2 + J(T)*DT*DT*DT/6
where:
P is position in feet
V is velocity in feet per second (change in position over time)
A is acceleration/deceleration in feet per second squared (change in velocity over time)
J is jerk in feet per second cubed (change in acceleration/deceleration over time)
T is time in seconds
DT is delta time in seconds
and assume:
T4 > T3 > T2 > T1 and T4-T3 = T3-T2 = T2-T1 = DT
P1 is position in feet at T1
P2 is position in feet at T2 = T1+DT
P3 is position in feet at T3 = T2 + DT = T1+2*DT
P4 is position in feet at T4 = T3 + DT = T1+3*DT

```
|P1@T1     |P2@T2     |P3@T3     |P4@T4
|          |          |          |
------------------------------------------
     |V12       |V23       |V34
     |          |          |
     ---------------------------
          |A13       |A24
          |          |
          -------------
               |J14
               |
```

therefore:

```
V12  = (P2-P1)/(T2-T1) = (P2-P1)/DT
V23  = (P3-P2)/(T3-T2) = (P3-P2)/DT
V34  = (P4-P3)/(T4-T3) = (P4-P3)/DT

A13  = (V23-V12)/((T3-T1)/2) = (V23-V12)/DT
     = ((P3-P2)/DT - (P2-P1)/DT)/DT
     = (P3-2*P2+P1)/(DT*DT)
A24  = (V34-V23)/((T4-T2)/2) = (V34-V23)/DT
     = ((P4-P3)/DT - (P3-P2)/DT)/DT
     = (P4-2*P3+P2)/(DT*DT)

J14  = (A24-A13)/((T4-T1)/3) = (A24-A13)/DT
     = ((P4-2*P3+P2)/(DT*DT) - (P3-2*P2+P1)/(DT*DT))/DT
     = (P4-3*P3+3*P2-P1)/(DT*DT*DT)
```

The result of this analysis was that velocity can be calculated from two position values, acceleration/deceleration can be calculated from three position values, and jerk can be calculated from four position values.

To complete Task 4.2, the following actions were performed to add a classification detector.

In file PARAMS for program SIMPRO, a parameter for the number of classify detector vehicle classes LDC was added and the value was set to 13.

In file CHARAC for program SIMPRO:

1. Deleted the variable IENCE.
2. Changed the size of the character strings IPRES and IPULS from 4 to 8 characters.
3. Added the character strings ICLAS, IUNCLL, and IUNCLU with a size of 8 characters.

In file LOOPS for program SIMPRO, the following variables were added:

Variable Name	Description	Type	Units	COMMON Block
DETACC	Vehicle acceleration/deceleration calculated from DETP2, DETP3, and DETP4	Double Precision	Feet per second squared	LOOPSD
DETALV	Vehicle acceleration/deceleration last value (last value of DETACC)	Double Precision	Feet per second squared	LOOPSD
DETCLK	Number of vehicle classifications for detector	Integer	None	LOOPS2
DETCLL	List of detector classification minimum overall vehicle lengths (DETCLK values per detector)	Integer Array	Feet	LOOPSD
DETCLL	List of detector classification maximum overall vehicle lengths (DETCLK values per detector)	Integer Array	Feet	LOOPSD
DETCLN	List of detector classification names (DETCLK values per detector)	Character Array	None	LOOPSC
DETCLS	Detector classification	Character	None	LOOPSC
DETIV	Vehicle IV number currently crossing the detector	Integer	None	LOOPSD
DETLEN	Vehicle length	Double Precision	Feet	LOOPSD
DETP1	Vehicle position 1 (oldest value)	Double Precision	Feet	LOOPSD
DETP2	Vehicle position 2 (next to the oldest	Double	Feet	LOOPSD

	value)	Precision		
DETP3	Vehicle position 3 (previously the newest value)	Double Precision	Feet	LOOPSD
DETP4	Vehicle position 4 (newest value)	Double Precision	Feet	LOOPSD
DETSLP	Vehicle slope (jerk) calculated from DETP1, DETP2, DETP3, and DETP4	Double Precision	Feet per second cubed	LOOPSD
DETSLV	Vehicle slope (jerk) last value (last value of DETSLP)	Double Precision	Feet per second cubed	LOOPSD
DETTFB	Time the front bumper of the vehicle crossed the beginning location for the detector	Double Precision	Seconds	LOOPSD
DETTFE	Time the front bumper of the vehicle crossed the ending location for the detector	Double Precision	Seconds	LOOPSD
DETTRE	Time the rear bumper of the vehicle crossed the ending location for the detector	Double Precision	Seconds	LOOPSD
DETVBE	Average vehicle velocity from DETTFB to DETTFE	Double Precision	Feet per second	LOOPSD
DETVEL	Vehicle velocity calculated from DETP3 and DETP4	Double Precision	Feet per second	LOOPSD
DETVLV	Vehicle velocity last value (last value of DETVEL)	Double Precision	Feet per second	LOOPSD

In order to calculate vehicle length and classify a vehicle, SIMPRO calculated and stored a minimal amount of information that an actual detector in the field might be able to discover:

(1) The vehicle number crossing the detector DETIV.
(2) The time when the front bumper of the vehicle crossed the beginning location for the detector DETTFB.
(3) The time when the front bumper of the vehicle crossed the ending location for the detector DETTFE.
(4) The time when the rear bumper of the vehicle crossed the ending location for the detector DETTFE.
(5) The position of the front bumper while the front bumper was between the beginning and ending location for the detector DETP1 (oldest), DETP2, DETP3, and DETP4 (newest).

In file simpro1.for in the BLOCK DATA routine:

1. Initialized ICLAS to “CLASSIFY”.
2. Commented out the initialization for IENCE.
3. Changed the initialization of IPRES from “PRES” to “PRESENCE”.
4. Changed the initialization of IPULS from “PULS” to “PULSE”.
5. Added an initialization for IUNCLL to “UNCLAS-L”.

6. Added an initialization for IUNCLU to “UNCLAS-U”.
7. Changed the initialization for ITYPLD from “PRES” to “PRESENCE”.
8. Added initialization for all new detector integer and real data to zero.
9. Added initialization for all new detector character data to blanks.

In file simpro2.for in subroutine RLOOPD:

1. Commented out the definition and use of IT1X.
2. Changed the reading and printing of the detector type from two 4-character fields to one 8-character field.
3. Added reading the number of vehicle classifications for a detector DETCLK from the normal record containing the standard information for a detector in the next 4-character integer field following the list of lane numbers for the detector.
4. Added code to allow the detector type “CLASSIFY” in addition to “PRESENCE” and “PULSE”.
5. Added a check to ensure that a classification detector could not occupy more than 1 lane.
6. If the detector type was “CLASSIFY”, then
   a. Checked to ensure that the number of vehicle classifications for the detector (DETCLK) is greater than or equal to 1 and less than or equal to LDC.
   b. Read the detector classification data DETCLN, DETCLL, and DETCLU for each classification in the immediately following record(s) (five sets of (i) an 8-character classification name, (ii) a 4-digit minimum overall vehicle length, and (iii) a 4-digit maximum overall vehicle length).
   c. Checked to ensure that detector classification 1 minimum overall vehicle length is zero.
   d. Checked to ensure that detector classification DETCLK maximum overall vehicle length is 999.
   e. Checked to ensure that the minimum overall vehicle length is less than the maximum overall vehicle length.
   f. For detector classification 2 through DETCLK, checked to ensure the minimum overall vehicle length is equal to the maximum overall vehicle length for the previous detector classification.
   g. Printed out the vehicle classification number, the vehicle classification name, the vehicle classification minimum overall vehicle length, and the vehicle classification maximum overall vehicle length.
7. Added code to allow a classification detector to not have to be connected to a signal controller phase.

In file simpro3.for in subroutine INTERP, if (1) the intersection control was signalized, (2) the number of loop detectors was greater than 0, and (3) the old position of the rear bumper of a vehicle was negative (the rear bumper of the vehicle is still on the previous link), then code to call subroutine CHKLDT using the vehicle’s previous link, the old position of the front bumper adjusted for the previous link’s length was added, and the new position of the front bumper was adjusted for the previous link’s length. Previously, after the front bumper of a vehicle crossed the end of the inbound lane, no further checks were performed to determine whether the vehicle was still activating the detector. This change corrected the error.

In file simpro3.for in subroutine IBAP, the vehicle's current link, the old position of the front bumper on the current link, and the new position of the front bumper on the current link were added as parameters for the call to subroutine CHKLDT.

In file simpro3.for in subroutine CHKLDT:

1. Added parameters to the subroutine for:
    a. The link to check JL.
    b. The old position of the front bumper on the link to check POSFBO.
    c. The new position of the front bumper on the link to check POSFBN.
2. Added a double precision parameter for a value not set (NOTSET) (also referred to as an invalid number) and set the value to -99999.0.
3. Changed IL to JL, POSOLD to POSFBO, POSNEW to POSFBN, POSORB to POSRBO, and POSNRB to POSRBN.
4. If the detector type was "CLASSIFY" then,
    a. If the front bumper of the vehicle crossed the beginning location for the detector, then
        1) Set all data for the detector to invalid numbers (NOTSET).
        2) Store the vehicle ID number as:
           DETIV = IV
        3) Store the position of the front bumper as:
           DETP4 = POSFBN
        4) Calculate the time when the front bumper of the vehicle crossed the beginning location for the detector using the cubic equation solver and save the minimum non-negative time as DETTFB

       else if the current vehicle number IV matches the vehicle number for the detector DETIV and the front bumper of the vehicle was within the beginning and ending location for the detector, then
        1) Set DETP1 = DETP2.
        2) Set DETP2 = DETP3.
        3) Set DETP3 = DETP4.
        4) Set DETP4 = POSFBN.
        5) If DETP4 was equal to DETP3 (the vehicle is stopped), then
            a. Set DETVEL to NOTSET.
            b. Set DETACC to NOTSET.
            c. Set DETSLP to NOTSET.

           else
            a. Calculated the velocity from DETP3 to DETP4 as
               DETVEL = (DETP4-DETP3)/DT
            b. Saved the last value of DETVEL as
               DETVLV = DETVEL
            c. If DETP2 was set to a valid number, then

1. Calculated the acceleration/deceleration from DETP2 to DETP4 as DETACC = (DETP4-2*DETP3+DETP2)/(DT*DT)
2. Saved the last value of DETACC as DETALV = DETACC

d. If DETP1 was set to a valid number, then
1. Calculated the jerk from DETP1 to DETP4 as DETSLP = (DETP4-3*DETP3+3*DETP2-DETP1)/(DT*DT*DT)
2. Saved the last value of DETSLP as DETSLV = DETSLP

b. If the current vehicle number IV matched the vehicle number for the detector DETIV and the front bumper of the vehicle crossed the ending location for the detector, then
1) Set DETACC, DETP1, DETP2, DETP3, DETP4, DETSLP, and DETVEL to invalid numbers (NOTSET).
2) Calculated the time when the front bumper of the vehicle crossed the ending location for the detector using the cubic equation solver and saved the minimum non-negative time as DETTFE.
3) If the elapsed time from when the front bumper of the vehicle crossed the beginning location for the detector (time DETTFB and position DSTRT) until the front bumper of the vehicle crossed the ending location for the detector (time DETTFE and location DSTOP) was positive and then calculated the average velocity of the vehicle's front bumper over the detector as DETVBE = (DSTOP-DSTRT)/(DETTFE-DETTFB)

c. If the current vehicle number IV matched the vehicle number for the detector DETIV and the rear bumper of the vehicle crossed the ending location for the detector, then
1) Calculated the time when the rear bumper of the vehicle crossed the ending location for the detector using the cubic equation solver and saved the minimum non-negative time as DETTRE.
2) Calculated the elapsed time TT from when the front bumper of the vehicle crossed the ending location for the detector until the rear bumper of the vehicle crossed the ending location for the detector as:
TT = DETTRE - DETTFE
3) If the elapsed time TT was positive and the last saved velocity DETVLV was a valid number, then
a. Calculated the vehicle length as:
DETLEN = DETVLV*TT
b. If the last saved acceleration/deceleration was a valid number, then added the acceleration/deceleration component to the vehicle length as:
DETLEN = DETLEN + DETALV*TT*TT/2
c. If the last saved jerk was a valid number, then added the acceleration/deceleration component to the vehicle length as:
DETLEN = DETLEN + DETSLV*TT*TT*TT/6
d. If the vehicle length DETLEN was less than 0.0, then
1. Set DETLEN to 0.0.
2. Set DETCLS to IUNCLL.

        3. Skipped the normal classification process.
    e. If the vehicle length DETLEN was greater than 999.0, then
        1. Set DETLEN to 999.0.
        2. Set DETCLS to IUNCLU.
        3. Skipped the normal classification process.
    f. Searched the list of detector classifications and if DETLEN was greater than the minimum overall vehicle length for vehicle class I and if DETLEN was less than or equal to the maximum overall vehicle length for vehicle class I, then set the vehicle class name as:
       DETCLS = DETCLN(I)

In file simpro3.for, subroutine CUBIC was added to solve for X for A*X**3 + B*X**2 + C*X + D = 0. Subroutine CUBIC has parameters A, B, C, D, NX, X1, X2, and X3 where A, B, C, and D are the constants of the cubic equation, NX is the number of roots found, and X1, X2, and X3 are the roots of the equation. The following is the general logic for subroutine CUBIC:

1. Initialized NX to 0 and X1, X2, and X3 to 0.0.
2. If A was equal to zero, then
    a. If B was equal to zero, then
        1) If C was equal to zero, then
            a. A=0, B=0, and C=0
            b. Returned (D must be 0).
        2) Else
            a. A=0, B=0, and C<>0
            b. Solved the simple equation C*X + D = 0 and returned.
    b. Else
        1) A=0 and B<>0
        2) Solved the quadratic equation B*X**2 + C*X + D = 0 and returned.
3. Else
    a. A<>0
    b. Solved the cubic equation A*X**3 + B*X**2 + C*X + D = 0 and returned.

In subroutine SIMCON in file simcon.for for program SIMCONV:

1. Added a parameter for the number of classify detector vehicle classes LDC and set the value to 13.
2. Added a parameter for the number of loop detectors NLS, set the value to 30, and changed all values of 20 to NLS (the maximum number of loop detectors was incorrectly specified).
3. Added variables DETCLN, DETCLK, DETCLL, and DETCLU using the definitions for file LOOPS for program SIMPRO.
4. In do loop 4900,
    a. Added reading DETCLK after the last data value on the record as a 4-digit integer.
    b. If the detector type was “CL” for classify, then

1. Checked to ensure that DETCLK is greater than or equal to 1 and less than or equal to LDC.
2. Read up to 5 sets of DETCLN, DETCLL, and DETCLU per record using 5 sets of (a) an 8-character value, (b) a 4-digit integer, and (c) a 4-digit integer.

5. In do loop 5000,
   a. Added code to convert detector type "CL" to "CLASSIFY" in addition to "IN" for "INACTIVE," "PR" for "PRESENCE," and "PU" for "PULSE".
   b. Added writing DETCLK after the last data value on the record as a 4-digit integer.
   c. If the detector type was "CLASSIFY", then write up to 5 sets of DETCLN, DETCLL, and DETCLU per record using 5 sets of (a) an 8-character value, (b) a 4-digit integer, and (c) a 4-digit integer.

The following is the classification and an alternate classification for Jim Bonneson's application:

```
NUMBER OF VEHICLE CLASSES -- =    2
DETECTOR CLASS NAME  1 ----- = "Auto    " LENGTH GT   0 AND LE  25
DETECTOR CLASS NAME  2 ----- = "Truck   " LENGTH GT  25 AND LE 999

NUMBER OF VEHICLE CLASSES -- =    4
DETECTOR CLASS NAME  1 ----- = "UNCLAS-L" LENGTH GT   0 AND LE  10
DETECTOR CLASS NAME  2 ----- = "Auto    " LENGTH GT  10 AND LE  25
DETECTOR CLASS NAME  3 ----- = "Truck   " LENGTH GT  25 AND LE 150
DETECTOR CLASS NAME  4 ----- = "UNCLAS-U" LENGTH GT 150 AND LE 999
```

Referring to the AASHTO A Policy on Geometric Design of Highways and Streets (2001), Exhibit 2-1 - Design Vehicle Dimensions US Customary, page 17, the following classification and alternate classification was developed using overall vehicle length: (8)

```
NUMBER OF VEHICLE CLASSES -- =    8
DETECTOR CLASS NAME  1 ----- = "P&TR    " LENGTH GT   0 AND LE  20
DETECTOR CLASS NAME  2 ----- = "MH&SU   " LENGTH GT  20 AND LE  32
DETECTOR CLASS NAME  3 ----- = "CitySBUS" LENGTH GT  32 AND LE  40
DETECTOR CLASS NAME  4 ----- = "P/B&BU45" LENGTH GT  40 AND LE  45
DETECTOR CLASS NAME  5 ----- = "WB4050MH" LENGTH GT  45 AND LE  58
DETECTOR CLASS NAME  6 ----- = "ABUS&W62" LENGTH GT  58 AND LE  70
DETECTOR CLASS NAME  7 ----- = "WB65&67D" LENGTH GT  70 AND LE 100
DETECTOR CLASS NAME  8 ----- = "WB100109" LENGTH GT 100 AND LE 999
```

```
NUMBER OF VEHICLE CLASSES -- =   10
DETECTOR CLASS NAME  1 ----- = “UNCLAS-L” LENGTH GT   0 AND LE  10
DETECTOR CLASS NAME  2 ----- = “P&TR    “ LENGTH GT  10 AND LE  20
DETECTOR CLASS NAME  3 ----- = “MH&SU   “ LENGTH GT  20 AND LE  32
DETECTOR CLASS NAME  4 ----- = “CitySBUS” LENGTH GT  32 AND LE  40
DETECTOR CLASS NAME  5 ----- = “P/B&BU45” LENGTH GT  40 AND LE  45
DETECTOR CLASS NAME  6 ----- = “WB4050MH” LENGTH GT  45 AND LE  58
DETECTOR CLASS NAME  7 ----- = “ABUS&W62” LENGTH GT  58 AND LE  70
DETECTOR CLASS NAME  8 ----- = “WB65&67D” LENGTH GT  70 AND LE 100
DETECTOR CLASS NAME  9 ----- = “WB100109” LENGTH GT 100 AND LE 150
DETECTOR CLASS NAME 10 ----- = “UNCLAS-U” LENGTH GT 150 AND LE 999
```

FHWA defines thirteen vehicle classifications as follows (there are no overall vehicle lengths given) (http://www.fhwa.dot.gov/ohim/tmguide/tmg4.htm#app4c):

1. Motorcycles (Optional) - All two- or three-wheeled motorized vehicles. Typical vehicles in this category have saddle-type seats and are steered by handlebars rather than steering wheels. This category includes motorcycles, motor scooters, mopeds, motor-powered bicycles, and three-wheel motorcycles. This vehicle type may be reported at the option of the State.
2. Passenger Cars - All sedans, coupes, and station wagons manufactured primarily for the purpose of carrying passengers, including those passenger cars pulling recreational or other light trailers.
3. Other Two-Axle, Four-Tire, Single-Unit Vehicles - All two-axle, four-tire vehicles, other than passenger cars. Included in this classification are pickups, panels, vans, and other vehicles such as campers, motor homes, ambulances, hearses, carryalls, and minibuses. Other two-axle, four-tire, single-unit vehicles pulling recreational or other light trailers are included in this classification. Because automatic vehicle classifiers have difficulty distinguishing class 3 from class 2, these two classes may be combined into class 2.
4. Buses - All vehicles manufactured as traditional passenger-carrying buses with two axles and six tires or three or more axles. This category includes only traditional buses (including school buses) functioning as passenger-carrying vehicles. Modified buses should be considered to be a truck and should be appropriately classified.
5. Two-Axle, Six-Tire, Single-Unit Trucks - All vehicles on a single frame, including trucks, camping and recreational vehicles, motor homes, etc., with two axles and dual rear wheels.
6. Three-Axle, Single-Unit Trucks - All vehicles on a single frame including trucks, camping and recreational vehicles, motor homes, etc., with three axles.
7. Four-Axle (or more), Single-Unit Trucks - All trucks on a single frame with four or more axles.
8. Four-Axle (or fewer), Single-Trailer Trucks - All vehicles with four or fewer axles consisting of two units, one of which is a tractor or straight truck power unit.
9. Five-Axle, Single-Trailer Trucks - All five-axle vehicles consisting of two units, one of which is a tractor or straight truck power unit.
10. Six-Axle (or more), Single-Trailer Trucks - All vehicles with six or more axles consisting of two units, one of which is a tractor or straight truck power unit.

11. Five-Axle (or fewer), Multi-Trailer Trucks - All vehicles with five or fewer axles consisting of three or more units, one of which is a tractor or straight truck power unit.
12. Six-Axle, Multi-Trailer Trucks - All six-axle vehicles consisting of three or more units, one of which is a tractor or straight truck power unit.
13. Seven-Axle (or more), Multi-Trailer Trucks - All vehicles with seven or more axles consisting of three or more units, one of which is a tractor or straight truck power unit.

NOTE: In reporting information on trucks, the following criteria should be used:

a. Truck tractor units traveling without a trailer will be considered single-unit trucks.
b. A truck tractor unit pulling other such units in a "saddle mount" configuration will be considered one single-unit truck and will be defined only by the axles on the pulling unit.
c. Vehicles are defined by the number of axles in contact with the road. Therefore, "floating" axles are counted only when in the down position.
d. The term "trailer" includes both semi- and full trailers.

This page intentionally left blank to facilitate printing on 2 sides.

CHAPTER 6. MODIFY INPUT AND OUTPUT DATA PROGRAMS TO HANDLE NEW DATA FEATURES

Task 5 was to modify the input and output data programs to handle new data features. As each task was performed, the input and output data programs were modified with the exception of gdvsim.

GDVADTA and SIMDATA did not expose all possible input to the user, and therefore when gdvsim was written to replace GDVDATA and SIMDATA, it also did not expose all possible input to the user. GDVDATA and SIMDATA simply read and wrote records of data without breaking the records up into fields and did not allow the user to make any changes. Within gdvsim, the data structures were created, data were read into the data structures, and data were written out from the data structures within. Additionally, there was insufficient time and resources to modify gdvsim within the scope of this project.

The gdvdata and simdata files were manually edited to create the input for all the new features for testing. GDVCONV and SIMCONV were modified to read and write data for the new features and to print out information on the new features.

GEOPRO, DVPRO, and SIMPRO were modified to read data for the new features and to print out information on the new features. DISPRE_J was modified to accommodate the increased number of driver and vehicle classes. The Java version of GEOPLOT and DISPRO did not have to be modified because of the new features.

This page intentionally left blank to facilitate printing on 2 sides.

CHAPTER 7. SUMMARY

This project made significant enhancements to the TEXAS Model for Intersection Traffic. In addition, a large number of errors in the TEXAS Model were corrected.

As part of the contract, the final version of the source code and all files used to compile and create the installation files for the Windows Intel and Linux Intel platforms were delivered under the terms of the GNU General Public License as published by the Free Software Foundation. A copy of the GNU General Public License can be found in source\GNU_General_Public_License.txt. The following was added to all source code files:

```
*** ************************************************************ ***
*** *                                                          * ***
*** * This program is free software; you can redistribute it   * ***
*** * and/or modify it under the terms of the GNU General Public * ***
*** * License as published by the Free Software Foundation;     * ***
*** * either version 2 of the License, or (at your option) any  * ***
*** * later version.                                           * ***
*** *                                                          * ***
*** * This program is distributed in the hope that it will be   * ***
*** * useful, but WITHOUT ANY WARRANTY; without even the implied * ***
*** * warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR   * ***
*** * PURPOSE.  See the GNU General Public License for more     * ***
*** * details.                                                 * ***
*** *                                                          * ***
*** * You should have received a copy of the GNU General Public * ***
*** * License along with this program; if not, write to the Free * ***
*** * Software Foundation, Inc., 51 Franklin Street, Fifth      * ***
*** * Floor, Boston, MA 02110-1301, USA.                        * ***
*** *                                                          * ***
*** * For more information: http://www.gnu.org/licenses/gpl.html * ***
*** *                                                          * ***
*** ************************************************************ ***
```

These enhancements will allow other developers and users to simulate advanced ITS features.

This page intentionally left blank to facilitate printing on 2 sides.

REFERENCES

1. Lee, Clyde E., Rioux, Thomas W., and Copeland, C. R., "The Texas Model for Intersection Traffic-Development", Research Report No. 184-1, Project 3-18-72-184, Center for Highway Research, The University of Texas at Austin, December 1977, 408 pp
2. Lee, Clyde E., Rioux, Thomas W., Savur, V. S., and Copeland, C. R., "The Texas Model for Intersection Traffic - Programmer's Guide", Research Report No. 184-2, Project 3-18-72-184, Center for Highway Research, The University of Texas at Austin, December 1977, 346 pp
3. Lee, Clyde E., Grayson, G. E., Copeland, C. R., Miller, J. W., Rioux, Thomas W., and Savur, V. S., "The TEXAS Model for Intersection Traffic - User's Guide", Research Report No. 184-3, Project 3-18-72-184, Center for Highway Research, The University of Texas at Austin, July 1977, 89 pp
4. Lee, Clyde E., Savur, Vivek S., and Grayson, Glenn E., "Application of the TEXAS Model for Analysis of Intersection Capacity and Evaluation of Traffic Control Warrants", Research Report No. 184-4F, Project 3-18-72-184, Center for Highway Research, The University of Texas at Austin, July 1978, 102 pp
5. Lee, Clyde E., Inman, Robert F., and Sanders, Wylie M.., "User-Friendly TEXAS Model - Guide to Data Entry", Research Report No. 361-1F, Project 3-18-84-361, Center for Highway Research, The University of Texas at Austin, November 1985, 210 pp
6. Lee, Clyde E., Machemehl, Randy B.., and Sanders, Wylie M.., "TEXAS Model Version 3.0 (Diamond Interchanges)", Research Report No. 443-1F, Project 3-18-84-443, Center for Highway Research, The University of Texas at Austin, January 1989, 224 pp
7. Rioux, Tom, Inman, Robert, Machemehl, Randy B., and Lee, Clyde E., "TEXAS Model for Intersection Traffic -- Additional Features", Research Report No. 1258-1F, Project 3-18-91/2-1258, Center for Transportation Research, The University of Texas at Austin, Austin, Texas, January 1993, 115 pp
8. AASHTO A Policy on Geometric Design of Highways and Streets, 2001