OPTIMIZATION OF A TRAFFIC SIGNAL
SYSTEM THROUGH COMPUTER SIMULATION

JUNE 1972 - NUMBER 8

BY

GARY W. DAVIES

JHRP

JOINT HIGHWAY RESEARCH PROJECT
PURDUE UNIVERSITY AND
INDIANA STATE HIGHWAY COMMISSION

Final Report

OPTIMIZATION OF A TRAFFIC SIGNAL SYSTEM THROUGH COMPUTER SIMULATION

TO: J. F. McLaughlin, Director June 22, 1972
Joint Highway Research Project
FROM: H. L. Michael, Associate Director Project: C-36-17JJ
Joint Highway Research Project File: 8-4-36

The attached Final Report titled "Optimization of a Traffic Signal System Through Computer Simulation" is presented for acceptance as fulfillment of the objectives of the research proposed and approved as a JHRP study. Mr. Gary W. Davies, Graduate Assistant in Research on our staff, authored the Report under the direction of Professors W. L. Grecco and K. W. Heathington.

The Report introduces a simulation model, SIGNET, which was developed. The model is useful in signal timing studies and in evaluating many other traffic engineering measures, such as turn prohibitions, parking restrictions, etc. The research demonstrated that use of the model could make a signal system timing design more efficient and obtained more rapidly.

The Report is presented for acceptance.

Respectfully submitted,

Harold L. Michael
Associate Director

HLM:ms

cc: W. L. Dolch	M. E. Harr	M. B. Scott
R. L. Eskew	R. H. Harrell	J. A. Spooner
W. H. Goetz	M. L. Hayes	N. W. Steinkamp
W. L. Grecco	R. D. Miles	H. R. J. Walsh
M. J. Gutzwiller	J. W. Miller	K. B. Woods
G. K. Hallock	C. F. Scholer	E. J. Yoder

Final Report
OPTIMIZATION OF A TRAFFIC SIGNAL SYSTEM THROUGH
COMPUTER SIMULATION

by

Gary W. Davies
Graduate Assistant in Research

Joint Highway Research Project

Project No.: C-36-17JJ

File No.: 8-4-36

Conducted By
Joint Highway Research Project
Engineering Experiment Station
Purdue University

In Cooperation with
Indiana State Highway Commission

Purdue University
Lafayette, Indiana
June 22, 1972

Digitized by the Internet Archive
in 2011 with funding from

LYRASIS members and Sloan Foundation; Indiana Department of Transportation

ACKNOWLEDGMENTS

It would be a prodigious task to name and thank everyone who contributed to this research effort. My special thanks must go to a few, however, for without their guidance and assistance this work would have been impossible.

I would like to thank the Joint Highway Research Project for giving me the opportunity to do this research; Dr. William L. Grecco, for his guidance in choosing and developing the topic and performing the research; Dr. Kenneth W. Heathington, for his advice and encouragement throughout the project; Dr. I. W. Burr for his guidance in the statistical aspects of the project and his review of the manuscript; Professor H. L. Michael for his advice in traffic engineering; George Stafford for his invaluable assistance in collecting the needed traffic data; and Marian Sipes for her many favors while preparing the manuscript.

To all of these people go my most sincere thanks.

TABLE OF CONTENTS

	Page
LIST OF TABLES.....	v
LIST OF FIGURES.....	viii
ABSTRACT.....	xi
CHAPTER I. INTRODUCTION.....	1
CHAPTER II. DEVELOPMENT OF THE SIMULATION MODEL...	6
Mode of Representation.....	10
Mathematics of Vehicle Behavior.....	15
CHAPTER III. WORKING DETAILS OF THE SIMULATION MODEL.....	49
Geometric Configuration.....	53
SIGNET Program Structure	66
Simulation Output.....	93
CHAPTER IV. THE STREET NETWORK.....	96
CHAPTER V. MODEL TESTING AND VALIDATION.....	101
CHAPTER VI. SIGNAL TIMING STUDIES.....	128
CHAPTER VII. CONCLUSION.....	139
BIBLIOGRAPHY.....	141
APPENDICES	
Appendix A - PRESIG Program Details.....	145
Appendix B - SIGNET Program Details.....	168
Appendix C - Traffic Flow Data.....	236

LIST OF TABLES

Table		Page
1	Amber Decision Probability.....	28
2	Queue Discharge -- Seconds to Cross Stop Line.....	34
3	Lane Change Decision Probability.....	39
4	Left-Turn Decision Probability.....	45
5	Signal Indications.....	51
6	Analysis of Variance, Effect of Volume on Average Delay.....	110
7	Analysis of Variance, Effect of Volume on Average Travel Time.....	111
8	Analysis of Variance, Effect of Volume on Average Speed.....	112
9	Analysis of Variance, Effect of Volume on Average Queue Length (Link 57).....	113
10	Analysis of Variance, Effect of Target Velocity on Average Travel Time.....	114
11	Analysis of Variance, Effect of Target Velocity on Average Speed.....	115
12	Analysis of Variance, Effect of Effective Length on Average Travel Time.....	116
13	Analysis of Variance, Effect of Effective Length on Average Speed.....	117
14	Analysis of Variance, Effect of Trucks on Average Travel Time.....	118
15	Analysis of Variance, Effect of Trucks on Average Speed.....	119

LIST OF TABLES (Continued)

Table		page
16	Analysis of Variance, Effect of Maximum Desired Acceleration of Average Travel Time.....	120
17	Analysis of Variance, Effect of Maximum Desired Acceleration on Average Speed.....	121
18	Analysis of Variance, Effect of Characteristic Speed on Average Speed.....	122
19	Analysis of Variance, Effect of Characteristics Speed on Average Speed.....	123
20	Analysis of Variance, Effect of Proportionality Coefficient on Average Travel Time.....	124
21	Analysis of Variance, Effect of Proportionality Coefficient on Average Speed.....	125
22	Average Delay Caused by Signal Timing Alternatives, sec./veh.....	133
Appendix		
Table		
A-1	Default Radii Used In PRESIG.....	150
A-2	PRESIG Punched Card Input Format.....	151
A-3	PRESIG Program Code.....	155
B-1	SIGNET Punched Card Input Format.....	180
B-2	SIGNET Program Code.....	188
C-1	Network Traffic Signal Settings.....	236
C-2	Tabulation of Traffic Characteristics, Average Conditions, 4:00 to 4:15 P.M.....	242
C-3	Tabulation of Traffic Characteristics, Average Conditions, 4:15 to 4:30 P.M.....	245
C-4	Tabulation of Traffic Characteristics, Average Conditions, 4:30 to 4:45 P.M.....	248

LIST OF TABLES (Continued)

Table		Page
C-5	Tabulation of Traffic Characteristics, Average Conditions, 4:45 to 5:00 P.M.....	251
C-6	Tabulation of Traffic Characteristics, Average Conditions, 5:00 to 5:15 P.M.....	254
C-7	Tabulation of Traffic Characteristics, Average Conditions, 5:15 to 5:30 P.M.....	257

LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
1	Content of ICAR Word.....	14
2	Comparison of Negative Exponential and Translated Negative Exponential.....	17
3	General Queue Discharge Relationships.....	25
4	Amber Decision Probability Distribution....	27
5	Queue Discharge Comparison.....	35
6	Change Zone Definition.....	37
7	Lane Change Decision Probability Distribution.	40
8	Variables Used in Computing Adjusted Stopping Rate.....	42
9	Left-Turn Decision Probability Distributions.....	46
10	A simplified Link Pair.....	54
11	Compensation for Unequal Discharge Boundaries.....	54
12	Legal Turning Movements.....	56
13	Samples of Turning Movement Combinations...	56
14	Contributing Turning Movements.....	58
15	Computation of Discharge Boundary Offset...	59
16	Discharge Boundary Computations.....	63
17	PRESIG Flowchart.....	65
18	SIGNET Program Structure.....	68
19	Main Program Chain.....	69

LIST OF FIGURES (Continued)

Figure		Page
20	Vehicle Generation Routine (INPLINK).....	72
21	Vehicle Update Routine (UPDATE).....	73
22	Lane Change Routine (CHANGE).....	75
23	Signal Update Routine (SIGNAL).....	78
24	Amber Decision Routine (AMBER).....	80
25	General Behavior Routine (GENBHR).....	81
26	Red Approach Routine (REDAPP).....	83
27	Right Turn Routine (RIGHT).....	85
28	Left Turn Routine (LEFT).....	86
29	Left Turn Conflict Routine (CONLT).....	89
30	Cross-Street Conflict Routine (CONST).....	90
31	Lag and Gap Acceptance Routine (GAP).....	91
32	Street Map of the Test Network.....	97
33	Schematic Diagram of the Test Network.....	98
34	Comparison of Travel Times, Link 28.....	103
35	Comparison of Travel Times, Link 56.....	104
36	Comparison of Travel Times, Link 59.....	105
37	Effect of Volume on Average Delay.....	110
38	Effect of Volume on Average Travel Time....	111
39	Effect of Volume on Average Speed.....	112
40	Effect of Volume on Average Queue Length (Link 57).....	113
41	Effect of Target Velocity on Average Travel Time.....	114
42	Effect of Target Velocity on Average Speed.	115
43	Effect of Effective Length on Average Travel Time.....	116

LIST OF FIGURES (Continued)

Figure		Page
44	Effect of Effective Length on Average Speed.....	117
45	Effect of Trucks on Average Travel Time....	118
46	Effect of Trucks on Average Speed.....	119
47	Effect of Maximum Desired Acceleration on Average Travel Time.....	120
48	Effect of Maximum Desired Acceleration on Average Speed.....	121
49	Effect of Characteristic Speed (a_0) on Average Travel Time.....	122
50	Effect of Characteristic Speed (a_0) on Average Speed.....	123
51	Effect of Proportionality Coefficient (K) on Average Travel Time.....	124
52	Effect of Proportionality Coefficient (K) on Average Speed.....	125
Appendix		
Figure		
A-1	Intersection Types.....	146
B-1	Input Data Deck Structure.....	169
B-2	SIGNET Program Deck Structure.....	235

ABSTRACT

Davies, Gary William. M.S.C.E., Purdue University, June, 1972. Optimization of a Traffic Signal Network through Computer Simulation. Major Professors: William L. Grecco and Kenneth W. Heathington.

The objective of this study was to develop a simulation program which would be suitable for use in the design of traffic signal systems. In keeping with this, it was desired to develop, program, and test a conceptual model that would simulate the flow of vehicles on a street network controlled by traffic signals. Secondary goals were that the model be general enough to apply to any moderate sized network, and that the final program be such that the inputs and outputs could be easily understood by a non-computer oriented traffic engineer.

A simulation model, SIGNET, was developed to reproduce traffic under laboratory conditions. The program was fairly economical, achieving a 6.5 to 1 real-time to simulation ratio. The model was quite microscopic in nature, since each vehicle was individually processed through the network. A sensitivity analysis indicated that the model simulated traffic flows quite accurately.

The model was used to evaluate several alternative timing plans, and indicated by a significant difference in

mean delay between two of these plans that it is indeed sensitive to variations in signal timing. It was demonstrated that the incorporation of simulation into the design procedure could make that procedure quicker, at the same time resulting in the production of a more efficient timing plan.

The use of the SIGNET model is not restricted to signal timing studies. In fact, the effects of many other traffic engineering measures may be evaluated by varying the program inputs. Such measures could include turning movement prohibitions, parking restrictions, unbalanced lane operation, and one-way street operation. A comprehensive set of improvements to the network can thus be easily evaluated. This capability is important because traffic signal optimization cannot be considered to be the sole means of obtaining maximum network efficiency.

CHAPTER I

INTRODUCTION

Since its first appearance on the urban scene, the automobile has become a dynamic social force molding the structure of cities and the style of daily life. It represents unprecedented individual mobility, comfort, and convenience; consequently it has become enormously popular.

With the increased use of the automobile, however, have come many problems. Congestion and the ensuing delays are just short of intolerable in many cities. A more subtle but perhaps more dangerous problem is the deterioration of the total environment: deaths, injuries, and property damage caused by accidents continue to increase; exhaust fumes and noise provide very real sources of pollution; and automobiles seem to penetrate into every available clear space of land.

Increasing auto traffic and the attendant problems outlined above are threatening the viability of the central business districts of many cities more each year. There are currently in vogue four approaches to the solution of these problems:

1. Road construction and improvement: this is very expensive, tends to generate new demand, and necessitates the provision of additional parking facilities. It involves a net loss of scarce land which might be put to other use.

2. Mass transit construction and improvement: This line of attack is currently being pursued in several major cities. Model preferences seem to be based primarily on personal tastes rather than on personal finances. Therefore the comfort, convenience, and privacy of mass transit vehicles must be improved considerably to at least stabilize transit usage in many urban areas.

3. Auto user constraints: in this area are included special peak-hour tolls, the staggering of office hours, and deliberately increased parking lot charges to discourage auto commuters. Such schemes would certainly tend to reduce critical peak hour volumes, but it is quite possible that the viability of the central business district would also be threatened by such action.

4. Increasing the efficiency of the existing system: before any of the above more drastic measures are implemented there is a general trend to obtain the maximum efficiency out of the existing facilities. Such actions involve, at least in their simpler forms, relatively minor expenditures and little or no construction.

Measures of this kind include turning movement controls at intersections, parking restrictions, unbalanced lane operation, one-way street operation, improvement of signs and markings, and channelization. Another measure, improvement of traffic signal operation, can result in improved use of the intersections, thereby significantly increasing overall network efficiency.

The at-grade intersection is the most critical component of the urban traffic network. It is the focal point of congestion and the greatest source of delay. A traffic signal can allocate the usage of this facility between the conflicting streams of vehicles, and by coordinating the timing of adjacent signals, the flow of traffic can be markedly improved.

To date the theoretical treatment of signal controlled traffic networks has been limited. The determination of optimal signal settings at an isolated intersection was thoroughly covered in a monograph by F. V. Webster (38). The interactive effects of more closely spaced signals on arterials has been considered with varying degrees of success in many methods, including the Yardeni Time-Space Design Model (41), the Little Maximal Bandwidth Model (28), the Delay/Difference-of-Offset Method (15,23,36), the Eagle Signal Method (31), and the Raus Method (31). The problems which these interactive effects raise in closed networks, however,

have slowed progress in that area. One of the more significant efforts to date has been the development of SIGOP, a traffic signal optimization program, by Peat, Marwick, Livingston, and Co. Given a set of data describing the geometrics of a signalized street network and traffic characteristics for a given period of interest, the function of SIGOP is to determine a plan of cycle length, phase splits, and offsets for the traffic signals in the network. Other procedures such as the Webster Method, the Delay/Difference-of-Offset Method, the British Combination Method (36), the Volume Priority Method (36), and the Preferential Street Method (36) may be combined to produce efficient timing plans also. Under different conditions each of these methods could produce the most efficient timing plan; it is necessary to devise a means of choosing this optimal plan, therefore.

Probably the most accurate means of determining the proper timing scheme for a given network would be to develop several alternatives, implement each, and study each scheme's effect on the system. This approach poses several problems, however. It would be very expensive to sufficiently instrument the system to gather the required data; the time required to undertake such a design process would also be prohibitive; it could even be inconvenient to the motorist in extreme cases.

Another method, computer simulation, may be far less expensive than field experimentation, and it offers a much wider scope for innovation and experimentation with the complete control the researcher has over its inputs and operation. It is probable that a design procedure incorporating simulation would also be considerably faster than other procedures.

The objective of this study was to develop a simulation program which would be suitable for use in the design of traffic signal systems. In keeping with this, it was desired to develop, program, and test a conceptual model that would simulate the flow of vehicles on a street network controlled by traffic signals. Secondary goals were that the model be general enough to apply to any moderate sized network, and that the final program be such that the inputs and outputs could be easily understood by a non-computer oriented traffic engineer.

In the following chapters, the development of the simulation model is traced and the basic portions of the model defined (Chapter II), the working details of the model are described (Chapter III), the study network is described (Chapter IV), the testing and validation of the model are explained (Chapter V), and the use of the model in the design of a signal timing plan is shown by example (Chapter VI).

CHAPTER II

DEVELOPMENT OF THE SIMULATION MODEL

The use of computer simulation as a traffic research tool is virtually untapped. At a time when businessmen, defense analysts, scientists, corporate managers, and others are employing simulation to assist them in making better decisions and to gain more understanding of complex processes, many traffic engineers have failed to consider seriously the use of simulation.

Many significant efforts, some with varying degrees of success, have been made, however, at closing this breach. Production-type, general purpose computers were not readily available until about 1954. In 1956, three digital computer simulations were reported in the traffic engineering literature. Gerlough discussed the simulation of traffic on a freeway (11). Wong's paper described the simulation of a portion of a multi-lane boulevard (40). Goode, Pollmar, and Wright constructed a model of a signalized intersection (16). Two separate studies of intersection simulation performed by Benhard (4) and Lewis (27) dealt with the intersection of two streets, each with two lanes, and with actuated signal control. The models were greatly simplified in that turning and passing were

prohibited. These and other early simulations permitted only a limited and somewhat arbitrary action of vehicles. Later investigations incorporated several refinements. Research by Kell (25) involved the simulation of the intersection of two urban two-lane streets under various types of traffic control. The much larger problem of simulating a network was first attacked by Stark (33) and Schwartz (32). Besides being slow, the Stark model was very heavily network dependent, meaning it could not be easily applied to any other network. Additionally, the printed output from this model was quite inadequate, since it relied primarily on an on-line visual display device and could not trace individual vehicles nor compute their delays. Although the Schwartz model was quite general, it too relied heavily on deterministic relationships in describing vehicle behavior. Traffic flow theory has advanced to the point where distributions would be more desirable in most of these cases, however.

Probably the most sophisticated model to date is TRANS, developed by Planning Research Corporation (13,14, 15,37). The size and configuration of arterial networks to which TRANS can be applied is very flexible. Either open networks (i.e. arterials) or closed networks can be studied. Wherever available traffic flow data permit distributions are used to describe vehicle actions. The parameters of these distributions are required as input.

In simulating a large, complex network in the District of Columbia consisting of over 200 links, a simulated time to real time ratio of 1:4 was achieved.

A major drawback to the widespread use of TRANS is that it was written in FAP for operation on the IBM 7090 computer. This would tend to make the program highly machine dependent. Several concepts employed in the TRANS model, however, were utilized in developing the model in this study.

There are two basic areas which in the past have slowed the development of simulation as a useful and practical tool for the traffic engineer. The most obvious has been the state of hardware development. Fortunately this problem has been all but resolved with the development of today's high level computers.

A more subtle, but more critical problem is that not enough is known about traffic itself, especially in the area of defining distributions which can be used to predict vehicle actions. Especially critical in this regard are left-turn gap and lag acceptance probabilities, which relate directly to the amount of delay a vehicle encounters while making a left turn. Other distributions needing further definition are the amber acceptance distribution, lane change gap acceptance distributions, and lane distributions at intersections. If the distributions used as a basis for a mathematical model are incorrect, it is

then quite probable that the model will also be incorrect. This problem which has considerably hampered simulation development in the past has been somewhat of a problem in this research. Until much more progress is made in the area of traffic flow theory, it will probably continue to be one.

To be useful, traffic simulation must satisfy three basic considerations:

1. The results of the simulation must fit the facts; in other words, observations obtained as a result of simulation must agree with similar results obtained from observations of actual traffic flow.

2. The time required to simulate a problem must be reasonable; that is, the ratio of simulated time to real time must be such that computer simulation of a street network is economically feasible. Before embarking on a study the objectives of that study should be thoroughly reviewed and the alternative techniques for meeting these objectives compared. The technique which does the job effectively at the least cost should then be chosen. When viewed from this standpoint, the use of simulation as a study tool often becomes much more attractive economically.

3. The results of simulation must be accessible in a format that is meaningful to those using them. The actual simulation takes place within the computer and is, of course, unobservable to the user (neglecting the use of

some type of on-line visual display device). Thus it is necessary to devise some means of displaying simulation results in a form convenient to the user.

If each of the above considerations is satisfied, it is presumed that the simulation model is a useful tool in the study of signal timing.

Mode of Representation

There are various methods which may be used to represent the flow of traffic within the computer. Early traffic simulations employed a physical notation (11,16). Binary "1's" were used to represent vehicles and "0's" were used to indicate the spaces between vehicles. Groups of memory cells were figuratively placed end to end to represent the roadway. Algebraic manipulations caused the "1's" to change position, thereby simulating the flow of traffic. With this mode of representation the vehicles can occupy only certain specified locations (bit positions) along the roadway and individual vehicles have no identity as such.

The memorandum notation utilizes an entire word to represent a vehicle. Various parts of the word are used for such individual characteristics as its time of entry into the system, its desired velocity, et cetera. These parts may be extracted and interpreted as desired. This method is more versatile in that each vehicle's characteristics are identifiable as it moves through the

network. This makes it possible to compute the delays associated with the individual vehicle.

Most simulation programs which have used the memorandum notation have considered the roadway as being composed of a series of unit blocks (40). These blocks represent the various positions which a vehicle can occupy. Each block is one lane wide and has a length which is equivalent to some fraction or multiple of the unit vehicle length. Thus a vehicle may occupy only a limited number of discrete positions. Velocity and acceleration are step functions of the unit block and the time increment of scanning. This procedure is adequate for some models, but offers severe restrictions when realistic total delays are desired.

A third method of representation has been called a mathematical notation (11). This form of representation is similar to the memorandum notation except that, in addition to its other characteristics, each vehicle is associated with its own position indicator. Its position is therefore continuous within the accuracy of the computer. At any time a vehicle's new position can be computed as a function of its last position, its velocity, its acceleration, and the time increment. Spacings between vehicles are available from their respective coordinates and the vehicle length.

A fully mathematical notation has generally been avoided in the past since it requires a more complicated logic. Maneuvers, such as turns, which must be accomplished at a specified location were deemed more difficult when the vehicle could occupy any position at the start of the maneuver. Furthermore, the mathematical processing of the vehicles is more complex, thereby increasing the computer time required. On the other hand, the elimination of limitations on the position increment would allow some increase in the size of the time increment for the same model accuracy, and would provide increased versatility, thereby easing the further application of the model.

The SIGNET¹ model employs a fully mathematical notation. It was felt that the advantages to be realized from a virtually continuous position vector outweighed the additional execution time required. Furthermore, the high speed of the CDC 6500 computer somewhat offset this loss.

Each vehicle in the SIGNET model, then, is completely represented by the information contained in four computer words. Current position, current velocity, and current acceleration are each individual words (POSN, VEL, and ACCEL respectively). The fourth word, ICAR, contains

¹SIGNET is the name of the simulation model developed in this project.

information on ten variables, as shown in Figure 1. The meaning and significance of each of these will be fully explained in later sections. Each variable is easily accessed via an unpacking function.

In actuality all of the drivers of vehicles within the roadway system are continually and simultaneously making decisions and modifying their behavior. The computer, however, can make only one simple logical choice at a time. In order to control all the occurrences at any given instant, it must process all decisions sequentially. In other words, it must process each decision for every vehicle in every lane, and for each lane in the system. It must do this in accordance with a prescribed sequence for each instance of time to be considered.

The selection of a suitable time increment is most important. If the time increment chosen is too large, it will not be possible to simulate all the events that may occur. On the other hand, if the increment is too small, many additional computations will be required for each event, resulting in additional computer time. In SIGNET the increment must be specified with the input.

In most simulations a critical factor is the minimum headway for vehicles. That is, since vehicles may enter the network only at each time increment, their minimum spacing will be equal to, or some multiple of, this increment. A method was developed for this study which isolated vehicle generation from the time increment.

LNKARR = link arrival time

START = initial coordinate at arrival on link

TVEL = target velocity

JTURN = turning movement

IACC = acceleration type

1 = stopped

2 = stopping

3 = stopped, waiting for turn

4 = stopping for turn

5 = stopping for lane change

6 = stopped for lane change

7 = free behavior

8 = car following

9 = going in delayed left turn

10 = going in normal turn

11 = slowing for turn (no conflict)

12 = stopping for red signal

EL = effective length

DECMIN = minimum desired deceleration

MAXACC = maximum desired deceleration

VEHTYP = vehicle type

1 = car

2 = truck

TRACE = vehicle being traced on travel-time route

0 = no

1 = yes

FIGURE 1
CONTENT OF ICAR WORD

Mathematics of Vehicle Behavior

The philosophy on which the SIGNET model is based is relatively simple in principle, but involves complex programming for its implementation. The basic premise is that all drivers have a target velocity at which they would prefer to travel if conditions meet certain minimum requirements. Acting to limit the driver in the pursuit of his target velocity are obstructions generated by interactions with other vehicles and the physical environment. Limitations encountered include leading vehicles moving at a slower speed, red signal indications, turning movements, obstructions to lane changes, and conflicts with vehicles from other links at intersections.

The mathematical relationships describing vehicle behavior can be divided into 9 separate areas: vehicle generation, vehicle updating, car following, free behavior, amber acceptance, stopping performance, queue discharge, lane changing, and turning performance. These relationships are presented below in the order in which they might confront a vehicle being processed through the network.

Vehicle Generation. Vehicles are generated at the zero coordinate of each input link on a per-lane basis through the use of a translated negative-exponential distribution. Traditionally the unmodified negative exponential distribution has been used to obtain inter-vehicle headways. It is of the form

$$P(h \geq t) = e^{-\beta_1 t}$$

where

$P(h \geq t)$ = probability of headway being greater than or equal to t ; and

β_1 = vehicle flow rate (veh./sec.).

However, being distributed in $(0, \infty)$, the negative-exponential does not compensate for a minimum headway which of course exists for every vehicle. Therefore, as proposed by Gerlough (34), a better approximation would be a negative-exponential with a translated axis:

$$P(h \geq t) = e^{\frac{-(t - \tau)}{(1/\beta_1 - \tau)}}$$

where

τ = amount of translation; equivalently, in a physical sense, the minimum headway. (See Figure 2)

Work by Dawson and Chimini (10) in fitting the hyperlang probability distribution to intervehicular headways indicates a good value of τ (δ_1 in their study) to be 0.75 second. This value would apply for all flow rates.

In the SIGNET model the generation time is independent of the simulation scan cycle. For each lane in the network a variable is maintained to indicate the next arrival time of a vehicle on that lane. This tally is updated by the translated negative-exponential distribution at generation time to indicate the exact arrival time of the next vehicle in that lane;

VOLUME = 400 vph

NEGATIVE EXPONENTIAL = $P(h \geq t) = e^{-\beta t}$

TRANSLATED NEGATIVE EXPONENTIAL = $P(h \geq t) = e^{-(t-\tau)} / (1/B-\tau)$

$\tau = .75 \text{ sec.}$

$\beta = .1111 \text{ sec/veh}$

FIGURE 2

COMPARISON OF NEGATIVE EXPONENTIAL AND
TRANSLATED NEGATIVE EXPONENTIAL

$$TNEXT = TNEXT + NEGEXP$$

where

$TNEXT$ = time of generation of the next vehicle;

and

$NEGEXP$ = headway determined by translated negative-exponential distribution.

Once a vehicle is generated its position is determined relative to the link coordinates. If the rear of its leader is at a coordinate less than zero, the vehicle is positioned a safe distance behind it:

$$POSN(J) = POSN(J-1) - EL(J-1) - 20.$$

Otherwise the distance the vehicle will travel at its generated velocity between the exact current time and its generation time is calculated:

$$DX1 = (TNEXT - TIME) \cdot V$$

where

$DX1$ = distance traveled;

$TIME$ = exact current time (a multiple of the scan cycle); and

V = generated velocity.

The vehicle is then positioned this distance before the zero coordinate.

The new vehicle's velocity is determined in a way to provide safety from a collision condition. If the spacing between it and its leader is less than 100 feet, its velocity is a function of both the spacing and the relative velocities:

$$DX2 = POSN(J-1) - EL(J-1) - POSN(J)$$

$$VEL(J) = VEL(J-1) + \frac{DX}{100} \cdot (TVEL(J) - VEL(J-1))$$

where

$DX2$ = intervehicular spacing; and

$TVEL(J)$ = target velocity of the new vehicle.

Otherwise the velocity is set equal to the newly generated target velocity. Under normal volume conditions the previous model will receive little, if any, use. It is provided primarily to insure against collision conditions.

Several other descriptors of the vehicle's behavior are set at this time, some stochastically and others deterministically. The vehicle's target velocity ($TVEL$), minimum desired deceleration ($DECMIN$), maximum desired acceleration ($ACCMAX$), and effective length (EL) are generated probabilistically according to their respective distributions. Additionally the vehicle type ($IVEHTYP$) is randomly determined from the link truck percentage, input with the link traffic volume. The turning movement ($JTURN$) to be pursued at the link head is also determined randomly from the link turning probabilities. Finally, the link arrival time ($LNKARR$) is set to the current time, and the acceleration type ($ACCTYP$) is set to free behavior. All of the above are then packed into the ICAR word which was previously described (Figure 1.).

Vehicle Updating. The various behavior relationships to be described below yield only a negative or positive acceleration rate which begins after some reaction lag and continues for the rest of the scan cycle. In SIGNET both the reaction lag (REACT) and the scan cycle (CYCLE) are specified with the input data.

A theoretical analysis of driver reactions and highway events indicates the importance of driver reaction in safety and highway design. Therefore the inclusion of this parameter is important in achieving realism with the model. In practice a large variety of values are used for reaction time; the Traffic Engineering Manual (24), however, recommends a time between 0.75 and 1.0 second for design purposes in urban traffic. To maintain continuity with the Carstens data (8), used in validating the queue-discharge model, a time of 0.75 second is recommended for use in SIGNET.

Movements of vehicles between scans are computed by adaptations of the equations of motion:

$$\begin{aligned}
 V2 &= VEL(J,I) + ACCEL(J,I) \cdot REACT \\
 POSN(J,I+CYCLE) &= POSN(J,I) + VEL(J,I) \cdot REACT + .5 \\
 &\quad \cdot ACCEL(J,I) \cdot REACT^2 + V2 \cdot \\
 &\quad (CYCLE - REACT) + .5 \cdot ACCEL(J,I+ \\
 &\quad REACT) \cdot (CYCLE - REACT)^2 \\
 VEL(J,I+CYCLE) &= V2 + ACCEL(J,I+REACT) \cdot (CYCLE-REACT)
 \end{aligned}$$

where

V2 = the velocity of vehicle J after
 the reaction period;

REACT = the reaction time;

CYCLE = scan cycle;

POSN(J,I) and
POSN(J,I+CYCLE)= the position of vehicle J at time
 I and I+CYCLE, respectively;

VEL(J,I) and
VEL(J,I+CYCLE) = the velocity of vehicle J at time
 I and I+CYCLE, respectively; and

ACCEL(J,I) and
ACCEL(J,I+REACT)= acceleration of vehicle J at time
 I and I+REACT, respectively.

Car Following. At the heart of the simulation model are the free-behavior and car-following relationships. Much of the full spectrum of behavior at an intersection involves a tracking or following process. An especially important example of this behavior is seen in the case of queue discharge. Therefore the stimulus-response equations of car following theory which were developed by Herman and associates (21,22) have been used in this study to describe certain patterns of intersection performance.

Herman's works pertaining to car following theory are highly valuable and apply directly to the problem of processing vehicles in a digital simulation. With two basic exceptions (turning movements and stopping), one of

the alternative vehicle behavior equations can be used to describe the behavior of individual vehicles within an intersection system.

Herman's equations have the general form

response = sensitivity x stimulus .

The best specific equation of this form Herman found was (in the notation of this study):

$$ACCEL(J+1, I+T) = a_0 \frac{VEL(J, I) - VEL(J+1, I)}{POSN(J, I) - POSN(J+1, I)}$$

where

ACCEL(J+1, I+T) = acceleration of car J+1, the follower, initiated at time I+T;

T = the car/driver lag;

VEL(J, I) and VEL(J+1, I) = the velocities of the leader and follower, initiated at time I;

POSN(J, I) and POSN(J+1, I) = the positions of the leader and follower initiated at time I; and

a_0 = the characteristic speed.

This equation is termed the reciprocal spacing model.

A steady state equation which fits empirical data quite well takes the form

$$q = a_0 K \ln(K_j/K)$$

where

q = flow;

a_0 = characteristic speed;

K = concentration; and

K_j = jam concentration.

This relationship has been successfully applied to freeway and tunnel flow successfully in the past, but under a specific intersection condition the equation is especially pertinent. This condition occurs during that portion of the green phase when flow is maximum. By differentiating and setting $dq/dK = 0$, it is found that when q approaches $q_{\max}$, K_j/K approaches e . Thus by choosing typical values for K_j , related values of $q_{\max}$ and a_0 can be computed. (It should be noted that $q_{\max}$ is analogous to saturation flow; in fact, by definition, these two values are probably the same thing for a line of straight-through vehicles.)

This relationship is directly related to this work due to its compatibility with the model of individual vehicle behavior, the reciprocal spacing model. The characteristic speed, a_0 , in the steady-state relationship is identical to that in the car following equation. Consequently, for any intersection it would only be necessary to measure maximum flow past the entry point (for a series of straight through movements from a long queue). Then a_0 could then be calculated and used in the

car following equation. It should be understood that a_0 does not correspond to the target velocity around which free-flowing vehicles center, but is the typical speed past the entry point of vehicles that started as part of a long stationary queue once maximum flow has been attained. Generally speaking, passage through the green of undelayed free-flowing vehicles will not result in maximum flow.

Free Behavior. Not all vehicles in a real system act as followers, however. An example of such a vehicle is the leader of a queue being discharged from a signal. In such a case behavior can be described by

$$ACCEL(J, I+T) = K (TVEL(J) - VEL(J, I))$$

where

$ACCEL(J, I+T)$ = acceleration of car J initiated at time $I+T$;

K = proportionality coefficient;

$TVEL(J)$ = target velocity of car J; and

$VEL(J, I)$ = velocity of car J at time I.

This equation is termed the free behavior model. Applying the reciprocal spacing model to queue members and the free behavior model to queue leaders should result in the general relationships shown in Figure 3.

Response to Amber Signal. The decision to stop or go on amber is based on probability distributions developed for the TRANS model (14). Data were taken from the work of Olson and Rothery (29) and Blackman and Crawford (6).

FIGURE 3
GENERAL QUEUE DISCHARGE RELATIONSHIPS

Results of these studies, relating probability of stopping to either distance or travel time from the intersection, were converted to deceleration rates required to stop at the stop line. The data were plotted on normal probability paper and an excellent linear regression fit ($R = 0.967$) was determined by the probit method (Figure 4).

The deceleration rate needed to stop at the stop line is computed by

$$ACC = \frac{VEL(J)^2}{2 (STOP - POSN(J))}$$

where

ACC = required negative acceleration; and

STOP = target stopped position.

The decision to stop or go is made probabilistically by reference to an amber decision probability table (Table 1).

Stopping Performance. Two types of stops occur in the model: (1) stopping first in line at the intersection, and (2) stopping behind another stopped vehicle. Empirical work at Ohio State University (35) indicated that the use of a constant deceleration stopping model would be quite realistic. They found that, when given the choice, drivers tended to decelerate to a stop at an approximately constant rate.

In the SIGNET stopping model, the parameters of a minimum desired deceleration rate are supplied and a value $DECMIN(J)$ is randomly selected for each vehicle J at its

FIGURE 4
AMBER DECISION PROBABILITY DISTRIBUTION

TABLE 1
Amber Decision Probability

<u>Acceleration Required to Stop at Stop Line (ft/sec²)</u>	<u>Probability of Stopping</u>
-0 to -0.99	1.000
-1.00 to -1.99	0.994
-2.00 to -2.99	0.989
-3.00 to -3.99	0.982
-4.00 to -4.99	0.972
-5.00 to -5.99	0.956
-6.00 to -6.99	0.935
-7.00 to -7.99	0.905
-8.00 to -8.99	0.867
-9.00 to -9.99	0.820
-10.00 to -10.99	0.762
-11.00 to -11.99	0.700
-12.00 to -12.99	0.624
-13.00 to -13.99	0.548
-14.00 to -14.99	0.468
-15.00 to -15.99	0.390
-16.00 to -16.99	0.318
-17.00 to -17.99	0.250
-18.00 to -18.99	0.190
-19.00 to -19.99	0.140

generation time. For each scanning cycle the required stopping rate for the first vehicle on the approach is computed:

$$ACCEL(J,I) = \frac{VEL(J,I)^2}{2 (POSN(J,I) - (STOP + E1))}$$

where

STOP = the position of the stop line; and

E1 = random normal stopping error.

The other terms have been defined previously.

When $ACCEL(J,I) \leq DECMIN(J)$, the vehicle begins stopping at rate $ACCEL(J,I)$ and continues to do so until zero velocity is reached. This approach tends to give the model added stability.

A similar model is used for stopping behind another vehicle, with the main difference being in the computation of the target stopped position. Vehicles stop at the position of the effective rear of the previously stopped vehicle.

The stopping model is not used until the leader has begun to stop. At that time the projected stopped position of the leader is computed:

$$PSL = \frac{-VEL(J-1,I)^2}{2 \cdot ACCEL(J-1,I)} + POSN(J-1,I)$$

where

PSL = projected stopped position of
leader ahead of vehicle J;

VEL(J-1,I) = current velocity of leader ahead
of vehicle J;

ACCEL(J-1,I) = current stopping acceleration of
leader ahead of vehicle J; and

POSN(J-1,I) = current position of leader ahead
of vehicle J.

The target stopped position of the vehicle under
consideration is then found by:

$$PS = PSL - EL(J-1)$$

where

PS = target stopped position of vehicle
J; and

EL(J-1) = effective length of leader ahead of
vehicle J.

The effective length of a vehicle is equivalent to the
average stopped spacing of vehicles stopped in queue.
This average stopped spacing is measured from the front
bumper of the leading vehicle to the front bumper of the
following vehicle, and therefore includes the vehicle
length and a clear space. Field studies have shown that
EL has a value of approximately 22 feet for cars (3, 18,
26, 34). In SIGNET a vehicle's effective length is
determined probabilistically at generation time from the
parameters supplied as input.

The stopping acceleration is found similarly to the case of the first vehicle on the approach, discussed above. A secondary check is then performed if the stopping acceleration is rejected, in which the projected stopping acceleration at the next scan cycle is calculated:

$$ACC = \frac{VEL(J, I+C)^2}{2 (POSN(J, I+C) - PS)}$$

where

ACC = projected stopping acceleration
at time I+C;
C = time interval;
VEL(J, I+C) = projected velocity of vehicle J
at time I+C;
POSN(J, I+C) = projected position of vehicle J
at time I+C; and
PS = target stopped position of vehicle
(same value as was computed
previously).

Studies have shown that deceleration rates of 8 to 9 feet per second are comfortable while rates up to 16 feet per second per second can be used without severe discomfort to the driver (19,39). For use in this stopping model a maximum desired deceleration rate of 10 feet per second per second was chosen. The projected stopping acceleration rate calculated above is compared to this value; if it is more severe, the previously calculated stopping acceleration is accepted and the vehicle begins

to stop. Otherwise it continues with zero acceleration.

Queue Discharge Performance. As in the stopping model, two types of vehicle action occur during queue discharge: (1) starting from first in line at the intersection, and (2) starting behind another vehicle. The first vehicle in line uses a modification of the free-behavior equation

$$ACCEL(J,I+T) = K (TVEL(J) - VEL(J,I))$$

The terms of the equation have been defined previously.

Work by Carstens (8) in the area of queue discharge revealed the average starting delay for the first vehicle in queue to be 0.35 seconds. Accordingly this value is used for T (the car/driver lag) in the above free behavior equation. When the first vehicle discharging from the queue is not being considered or when the vehicle is already underway T reverts to the original value found in the input data.

The original starting model for following vehicles used a simple application of the reciprocal spacing model

$$ACCEL(J+1,I+T) = a_o \frac{VEL(J,I) - VEL(J+1,I)}{POSN(J,I) - POSN(J+1,I)}$$

This version of the model was very dependent on the simulation time interval, however, since the state of the queue could only be checked once every scan cycle.

In reality, the driver is continuously analyzing his relationship to his leader. To approximate this behavior in the queue discharge model the leader-follower relationship is examined 1 second after the leader begins moving, and the follower's movement is begun. In this way a 1 second delay results which would adequately account for the car/driver lag, and the starting model is kept independent of the simulation time interval.

Independent studies by Greenshields (17,18), Carstens (8), and Bartle et al (2) have resulted in estimates of both starting delay and succeeding time spacings between vehicles discharging from a queue (Table 2). The variation in these estimates can be at least partially accounted for by the known influence of intersection geometrics on capacity. The SIGNET starting model, shown in Figure 5 with 95% confidence intervals for 50 iterations, shows good agreement with the other studies, even though confidence intervals were not reported with them.

Lane Changing. Lane change maneuvers associated with passing are not allowed in the SIGNET model. This restriction is not unrealistic since it is presumed that in most urban situations traffic volumes are high enough to greatly reduce the desire for such maneuvers. Lane changes associated with turning movements are permitted, however, since quite often the lane from which a vehicle wishes to turn is different from that in which it entered the link.

TABLE 2

Queue Discharge - - Seconds to Cross Stop Line

<u>Vehicle Number</u>	<u>Greenshields Data</u>	<u>Bartle Data</u>	<u>Carstens Data</u>	<u>SIGNET Model</u>	<u>Confidence Interval</u>
1	3.8	(3.1)	2.64	1.03	1.01 - 1.05
2	6.9	5.8	5.13	4.66	4.58 - 4.75
3	9.6	7.9	7.62	7.66	7.54 - 7.77
4	12.0	10.0	9.91	9.90	9.74 - 10.07
5	14.2	12.1	12.20	12.68	12.56 - 12.81
6	16.3	14.2	14.49	14.95	14.79 - 15.11
7	18.4	16.3	16.78	17.30	17.08 - 17.52
8	20.5	18.4	19.07	19.58	19.34 - 19.82
9	22.6	20.5	21.36	21.84	21.52 - 22.15
10	24.7	22.6	23.65	24.10	23.67 - 24.52

FIGURE 5
QUEUE DISCHARGE COMPARISON

Once it is determined that geometric conditions (i.e. existence of the new lane, a consideration with turning pockets) allow a change, a changing zone is defined in the new lane. The front coordinate of this zone is calculated by:

$$XF = POSN(J) + \overline{EL} + SPACE$$

where

XF = the front coordinate of the change zone;

$POSN(J)$ = the current position of vehicle J, the vehicle under consideration;

$\overline{EL}$ = the average effective length; and

$SPACE$ = the minimum allowable spacing, equal to one half of the vehicle's velocity (in m.p.h.).

The rear coordinate of the change zone is defined by:

$$XR = POSN(J) - EL(J)$$

where

XR = the rear coordinate of the change zone;
and

$EL(J)$ = the effective length of vehicle J.

(See Figure 6).

If another vehicle is in this change zone, the change is rejected.

FIGURE 6.
CHANGE ZONE DEFINITION

In the absence of such conflict the potential lane changer determines the expected travel time to the change zone of the next vehicle in the new lane, based on its current position and velocity. This travel time is defined as the lane change lag. The decision to change or not change is then made probabilistically by reference to a lane change lag acceptance table, Table 3. This distribution is taken from the TRANS model (14). (See Figure 7).

If the lane change is rejected the vehicle begins slowing at a rate equal to twice the vehicle's minimum desired acceleration. Occasionally, especially under high volume conditions, a hangup could result when two potential lane changers stop in each other's change zone. To avoid this, the vehicle in the left lane adjusts its stopping rate to stop behind the other's change zone through the following series of computations:

$$PFADJ = POSN(K) - \frac{VEL(K)^2}{2(ACCEL(K))}$$

$$PFLZN = PFADJ - 10. - EL(K)$$

$$ACCEL(J) = - \frac{VEL(J)^2}{2(PFLZN - POSN(J))}$$

where

J = vehicle in left lane;

K = conflicting vehicle in right lane;

PFADJ = projected final position of vehicle K;

and

TABLE 3
Lane Change Decision Probability

<u>Lane Change Lag (Sec.)</u>	<u>Probability of Acceptance</u>
0 to 0.5	0
0.5 to 1.0	0
1.0 to 1.5	0
1.5 to 2.0	0
2.0 to 2.5	0.090
2.5 to 3.0	0.180
3.0 to 3.5	0.310
3.5 to 4.0	0.490
4.0 to 4.5	0.660
4.5 to 5.0	0.810
5.0 to 5.5	0.900
5.5 to 6.0	0.960
6.0 to 6.5	0.985
6.5 to 7.0	0.995
7.0 to 7.5	0.999
7.5 to 8.0	1.000
8.0 to 8.5	1.000
8.5 to 9.0	1.000
9.0 to 9.5	1.000
9.5 to 10.0	1.000

FIGURE 7
LANE CHANGE DECISION PROBABILITY DISTRIBUTION

PFLZN = projected final rear coordinate of vehicle K's change zone; equivalently, the target stopped position of vehicle J. (See Figure 8).

The other terms have been defined previously.

Turning Performance. Vehicles which desire to turn left or right at the intersection must at some point cease operating under the stimulus-response model and undertake an independent fixed turning schedule.

The principal requirement is that vehicles must not exceed a certain maximum speed during the turn. Maximum turning velocity is related to turning radius and side friction by the equation

$$V_{TURN} = \sqrt{fgr}$$

where

V_{TURN} = maximum turning, ft./sec.;

f = coefficient of friction;

r = turning radius; and

g = acceleration of gravity.

The AASHO Policy on Geometric Design of Rural Highways (1) indicates that the 95-percentile turning speed is associated with side friction $f = 0.3$ for medium to low-speed turns. Therefore,

$$V_{TURN} = \sqrt{0.3 (32.2) r} = \sqrt{9.66 r}$$

where V_{TURN} is in feet per second and r is in feet. Thus

FIGURE 8

VARIABLES USED IN COMPUTING ADJUSTED
STOPPING RATE

right-turn, free left-turn, and delayed left-turn radii are supplied as input data and the maximum speeds associated with these radii are computed.

As a turning car approaches the intersection it is scanned at each simulation cycle, and if the current velocity is greater than maximum turning velocity, the deceleration rate required to reach maximum turning velocity exactly at the start-turn point is computed. For example, for a right turner,

$$ACCEL(J,I+T) = \frac{VTURN^2 - VEL(J,I+T)^2}{2 (BEGINPT(RT) - POSN(J,I+T))}$$

where

BEGINPT(RT) = start right turn position; and

VTURN = maximum right-turn velocity.

The other terms have been defined previously.

When $ACCEL(J,I) \leq DECMIN(J)$, the vehicle begins slowing to maximum turning velocity at rate $ACCEL(J,I)$, as in the stopping model. It continues to do so until reaching the start turn position, unless affected by more stringent conditions. Maximum turning speed is maintained throughout the turn, whereupon the vehicle resumes behavior under one of the stimulus-response models (car-following or free-behavior), whichever is more stringent.

The foregoing discussion does not imply that all vehicles will make their turns at maximum speed. Some will be affected by other vehicles in queue or vehicles

on the receiving link so that their turns will be made at considerably lower speeds. However, none will exceed maximum turning speed.

Left-turns are subject to further complications because they must be made across the path of opposing traffic. The driver must decide if gaps and lags in the opposing approach traffic are sufficiently large to be accepted. In this model, when a left-turning driver crosses the start-free-left-turn position, a decision is made to either accept or reject a time lag (i.e., the time required for the first straight-through vehicle on an opposing approach to reach the center of intersection conflict point). This lag is computed by

$$V = VEL(K)^2 + 2 \cdot ACCEL(L) \cdot (CENTER - POSN(K))$$

$$DT = \frac{(VEL(K) - V)}{ACCEL(K)}$$

where

- K = opposing vehicle;
- CENTER = coordinate of center of intersection;
- V = projected velocity of vehicle K at
CENTER; and
- DT = time lag.

The other terms have been defined previously.

The decision to accept or reject the time lag is made probabilistically by reference to a left-turn lag acceptance table (Table 4 and Figure 9). If the lag is

TABLE 4
Left-Turn Decision Probability

<u>Lag or Gap Size (Sec.)</u>	<u>Probability of Accepting Lag</u>	<u>Probability of Accepting Gap</u>
0 to 0.5	0	0
0.5 to 1.0	0	0
1.0 to 1.5	0	0
1.5 to 2.0	0	0
2.0 to 2.5	0	0
2.5 to 3.0	0	0
3.0 to 3.5	0.030	0.150
3.5 to 4.0	0.124	0.320
4.0 to 4.5	0.300	0.520
4.5 to 5.0	0.530	0.690
5.0 to 5.5	0.730	0.820
5.5 to 6.0	0.860	0.900
6.0 to 6.5	0.940	0.950
6.5 to 7.0	0.970	0.970
7.0 to 7.5	0.990	0.986
7.5 to 8.0	0.996	0.993
8.0 to 8.5	0.999	0.997
8.5 to 9.0	1.000	0.998
9.0 to 9.5	1.000	0.999
9.5 to 10.0	1.000	1.000

FIGURE 9

LEFT-TURN DECISION PROBABILITY DISTRIBUTIONS

acceptable the vehicle continues through the turn, never exceeding maximum free-turning velocity. If the lag is rejected, the vehicle immediately begins stopping at the constant rate required to stop at the left-turn waiting point, one quarter of the way through the delayed turn. Thereafter, the driver views each time gap appearing in the opposing traffic and determines its acceptability by reference to the left-turn gap acceptance probability table. If acceptable, the vehicle negotiates a delayed left-turn, never exceeding maximum delayed-turning velocity. It is possible, due to the use of random numbers in the model structure, that a waiting vehicle will at some point accept a short gap which it would have previously rejected. Such behavior is not unrealistic, since in actuality a degree of impatience often bears upon such a decision.

The log-normal probability distributions for gap and lag acceptance were taken from the TRANS model (14). They were developed from data supplied by James H. Kell.

The exact logic employed in processing left-turn vehicles is considerably more complex than would be inferred from the above discussion, principally due to the fact that other behavior conditions (such as stopping in queue) may supercede the standard left-turn schedule and decisions.

In the above sections the basic mathematical relationships governing an individual vehicle's behavior were presented. The next chapter, Chapter III, is concerned with the logic which employs these relationships in processing a vehicle through the network.

CHAPTER III

WORKING DETAILS OF THE SIMULATION MODEL

The original purpose of SIGNET was to evaluate the effect of traffic signal settings on traffic flow in a region of a city. Thus SIGNET is a tool which helps to optimize the timing of traffic signals in a street network. Alternatively SIGNET may be used to study the effect of other network modifications including: geometric design changes, such as additional turning lanes; traffic regulation changes, such as turning movement and parking prohibitions; and changes in the magnitudes and patterns of traffic demands.

In the SIGNET model a traffic network is considered to consist of a set of links. Each link carries traffic in one direction between two signalized intersections. Links are of three types: input, network, and output. An input link feeds traffic into the network; a network link carries traffic from one node (i.e. signalized intersection) in the interior of the network to another; and an output link removes traffic from the network.

For identification purposes each link is assigned a positive integer. Input links are numbered

$1, 2, 3, \dots, n_I$; network links are numbered $(n_I+1), (n_I+2), \dots, (n_I+n_N)$; and output links are numbered $(n_I+n_N+1), (n_I+n_N+2), \dots, (n_I+n_N+n_0)$. Because a digital computer can perform only one operation at a time, it is necessary for the simulation program to process each vehicle sequentially. In the SIGNET model this sequence is ordered by link number, with the vehicles within the link being processed from the link head to tail.

Special care should be taken in numbering the links so that all links feed lower numbered links (i.e. links which have already been processed). This is quite easy to accomplish in open networks, such as single arterials. Closed networks are more difficult, however, and in most cases the number of links feeding higher numbered links can only be minimized. These cases should, wherever possible, be confined to low volume locations to minimize their effect. Additionally, the SIGNET model employs supplementary logic to make the processing of such cases more accurate.

Each link (input type or network type) has a traffic signal associated with it. The signal appears at the head of the link, and may take on any of the 14 indications shown in Table 5. Only one of these may be on at any given time.

The characteristics of each link in the network must be supplied to the program as input. Physical

TABLE 5
Signal Indications

<u>Number</u>	<u>Indication</u>	<u>Movements Permitted</u>
1	Red	None
2	Green	all
3	Green Arrow Straight	#3, #4
4	Green Arrow Left	#6
5	Green Arrow Right	#1
6	Green Arrow Diagonal Left	#5
7	Green Arrow Diagonal Right	#2
8	Green Arrow Straight, Green Arrow Left	#3, #4, #6
9	Green Arrow Straight, Green Arrow Right	#1, #3, #4
10	Green Arrow Straight, Green Arrow Diagonal Left	#3, #4, #5
11	Green Arrow Straight, Green Arrow Diagonal Right	#2, #3, #4
12	Green Arrow Left, Green Arrow Diagonal Left	#5, #6
13	Green Arrow Right Green Arrow Diagonal Right	#1, #2
14	Amber	all

characteristics are defined through the use of PRESIG, the preliminary geometry determination program, which converts easily obtained geometry data to the required inputs of SIGNET. A full description of this program follows. Other link inputs include probabilistic characteristics pertaining to arrival and departure of vehicles and traffic signal characteristics which define the timing for each of the signal modes. Additionally, vehicular arrival rates are required for input links.

A simulation run is divided into two major parts: fill time and observe time. During fill time traffic is generated on the input links and pumped into the network until a stable condition is reached; this serves to create an initial network population. Generally a 15 minute fill time should be adequate, although this should be somewhat dependent on network size and configuration and magnitude of traffic flow. At the end of fill time appropriate initializations are performed and observation begins.

Simulation is carried out in discrete time intervals. During each simulation cycle four major operations are performed: (1) vehicles are generated on the input links; (2) vehicles are advanced within each link as conditions allow; (3) vehicles are moved from the front of their link to the rear of the receiving link at signalized intersections; and (4) bookkeeping operations are

performed. The bookkeeping operation records or summarizes all movements and delays. These data provide the various statistics that are of significance to the traffic engineer, such as vehicle-miles travelled, average attained speeds, densities, and volumes.

A variety of options are available in the use of SIGNET. For purposes of travel time testing or for more detailed analysis, a route may be designated through the network. A frequency analysis of travel times is then printed for each link on this route whenever a normal summary occurs. Similarly, a frequency analysis of queue lengths may be specified on up to 15 links. Turning probabilities, link input volumes, and truck percentages may each be reset up to 10 times in the course of one run. Primarily for debugging purposes, a full printout of all vehicles' positions, velocities, accelerations, and other pertinent data may be requested for any period of time. Finally, up to 15 complete summaries of flow statistics may be specified at any time during the observation time.

Geometric Configuration

The roadway is not represented physically, as such, in the computer. However, it is necessary to specify certain roadway references in order to make meaningful the positions of vehicles contained in the POSN word. Thus there is a need for a coordinate system in which vehicles operate. Consider the sketch (Figure 10) of a

FIGURE 10
A SIMPLIFIED LINK PAIR

FIGURE 11
COMPENSATION FOR UNEQUAL DISCHARGE BOUNDARIES

simplified pair of links. The zero coordinate of each link is taken to be the point where all contributing turning movements are complete. In turn, the end of the link, or discharge boundary, is the point where all other contributing movements to the receiving link are complete. In reality there could be several contributing movements at the tail (i.e. zero coordinate) of the link, each ending at a different point. It would be necessary to figuratively add a tangent section to each except the longest, thereby making all end at the same point (Figure 11).

There are a total of 8 possible movements at the head of a link (Figure 12). Of these, any 6 may be used for each link within the structure of SIGNET. These six movements are numbered in a counter-clockwise order, as they would confront a vehicle approaching the intersection. Movements to the right of straight-through are denoted 1 and 2; if only one exists, it is always denoted 1. Straight-through movements in lane 1 and lane 2 are denoted 3 and 4, respectively. The distinction is necessary to facilitate the use of lane distribution probabilities. Movements to the left of straight-through are numbered 5 and 6; if only one exists, it is always denoted 6. Several common intersection types are shown in Figure 13 with movement numbers illustrating this convention.

FIGURE 12
LEGAL TURNING MOVEMENTS

FIGURE 13
SAMPLES OF TURNING MOVEMENT COMBINATIONS

Two types of left turn are possible: free flowing, with radius $RADIUS(FLT)$, and delayed, with radius $RADIUS(DLT)$. Thus there nine possible contributing movements which must be considered in the determination of the zero coordinate (see Figure 14). Straight-through movements are not considered in this respect because their length is not critical.

The first step in tying down the network geometry is to determine for each link the location of the zero coordinate relative to the center of the contributing, or tail, intersection. Referring to Figures 11, 14, and 15, the following equation can be written:

$$ZERO = MAX (DBO(1), DBO(2), \dots, DBO(9))$$

where

$ZERO$ = distance from the center of the tail intersection to the zero coordinate;
 MAX = the maximum of the following terms; and
 $DBO(I)$ = the required discharge boundary offset for each contributing movement I , as derived in Figure 15.

If a movement does not exist within a given intersection, DBO for that movement is automatically zero.

Once the offsets have been calculated for all links, the stop-line and the center of the receiving intersection may be redefined relative to the zero coordinate, leaving the way clear to compute the beginning of turn points and

FIGURE 14.
CONTRIBUTING TURNING MOVEMENTS

$$\begin{aligned} \text{DBO}(2) &= \text{RADIUS} + A \\ &= \text{RADIUS} + W(L)/2 - \text{WIDTH}/2 - \text{PARKW}(L) \end{aligned}$$

FIGURE 15a.

$$\begin{aligned} X_1 &= (W(L)/2 - \text{WIDTH}/2 - \text{PARKW}(L)) \cdot \sqrt{2} \\ X_2 &= (W(N)/2 - \text{WIDTH}/2 - \text{PARKW}(N)) \\ X_3 &= \text{RADIUS} / \tan(22.5^\circ) \end{aligned}$$

$$\begin{aligned} \text{DBO}(1) &= X_1 + X_2 + X_3 \\ &= (W(L)/2 - \text{WIDTH}/2 - \text{PARKW}(L)) \cdot \sqrt{2} \\ &\quad + (W(N)/2 - \text{WIDTH}/2 - \text{PARKW}(N)) \\ &\quad + \text{RADIUS} / \tan(22.5^\circ) \end{aligned}$$

FIGURE 15b.

FIGURE 15.

COMPUTATION OF DISCHARGE BOUNDARY OFFSET

$$DBO(6,7) = RADIUS - WIDTH/2$$

$$DBO(3) = X_1 + X_2 \\ = RADIUS(TAN(22.5^\circ)) \\ - (W(N)/2 - WIDTH/2 - PARKW(N)/2)TAN(22.5^\circ)$$

FIGURE 15c.

FIGURE 15d.

FIGURE 15, cont.

COMPUTATION OF DISCHARGE BOUNDARY OFFSET

$$\begin{aligned} \text{DBO}(4,5) &= X_1 - X_2 \\ &= \text{RADIUS}(\tan(22.5^\circ)) \\ &\quad - (W(N)/2 - \text{PARKW}(N)/2) \tan(22.5^\circ) \end{aligned}$$

FIGURE 15e.

$$\begin{aligned} \text{DBO}(8,9) &= (\text{RADIUS} - W(N) + \text{WIDTH} + \text{PARKW})/2 \\ &\quad + (\text{RADIUS} + \text{WIDTH}(\text{NOLANE}(L) - /)) \\ &\quad - W(L)/2 + \text{PARKW}(L)/\sqrt{2} - \text{RADIUS} \\ &\quad + W(N) - \text{WIDTH} - \text{PARKW}(N) \end{aligned}$$

FIGURE 15f.

FIGURE 15, cont.

COMPUTATION OF DISCHARGE BOUNDARY OFFSET

discharge boundaries for each link. Referring to Figure 16, the beginning of turn point is computed by the general equation

$$\text{BEGINPT} = \text{COI} - \text{RADFACT}$$

where

BEGINPT = beginning of turn point;

COI = redefined center of intersection; and

RADFACT = a radius factor, dependent on the radius of the turning movement, link widths, and parking lane width. It is defined in Figure 16 for the various turning movement types.

The accompanying discharge boundary may also be computed, using the general equation

$$\text{DBNDRY} = \text{BEGINPT} + \text{LTURN} + (\text{ZERO}(N) - \text{DBO}(NT))$$

where

DBNDRY = discharge boundary for turning movement;

LTURN = length of turn, calculated from radius and angle of turn;

ZERO(N) = maximum discharge boundary offset of receiving link N; and

DBO(NT) = discharge boundary offset of turn under consideration, NT.

$$\begin{aligned} \text{BEGINPT} &= \text{COI} - \text{DBO}(1) \\ \text{DBNDRY} &= \text{BEGINPT} + 3/4 \pi (\text{RADIUS}) + \\ &\quad (\text{ZERO} - \text{DBO}(1)) \end{aligned}$$

(a)

$$\begin{aligned} A &= \text{PARKW}(N) + \text{WIDTH}/2 \\ \text{BEGINPT} &= \text{COI} - \text{RADIUS} - (\text{W}(N)/2 - A) \\ \text{DBNDRY} &= \text{BEGINPT} + \pi/2 (\text{RADIUS}) + (\text{ZERO} - \\ &\quad \text{DBO}(2)) \end{aligned}$$

(b)

$$\begin{aligned} A &= \text{W}(N)/2 - \text{WIDTH}/2 \\ &\quad - \text{PARKW}(N) \\ \text{BEGINPT} &= \text{COI} - \\ &\quad \text{RADIUS}(\text{TAN}(22.5^\circ)) \\ &\quad - A(\text{TAN}(22.5^\circ)) \end{aligned}$$

$$\text{DBNDRY} = \text{BEGINPT} + \pi/4 (\text{RADIUS}) + (\text{ZERO} - \text{DBO}(3))$$

(c)

$$\begin{aligned} A &= \text{W}(N)/2 - \text{PARKW}(N) \\ &\quad - \text{WIDTH}/2 \\ \text{BEGINPT} &= \text{COI} - \\ &\quad \text{RADIUS}(\text{TAN}(22.5^\circ)) \\ &\quad + \sqrt{2} A \\ \text{DBNDRY} &= \text{BEGINPT} + \pi/4 (\text{RADIUS}) \\ &\quad + (\text{ZERO} - \text{DBO}(4,5)) \end{aligned}$$

(d)

$$\begin{aligned} A &= (\text{W}(N)/2 - \text{WIDTH}/2 - \text{PARKW}(N)) \\ \text{BEGINPT} &= \text{COI} - \text{RADIUS} + A \\ \text{DBNDRY} &= \text{BEGINPT} + 3\pi/4 (\text{RADIUS}) \\ &\quad + (\text{ZERO} - \text{DBO}(6,7)) \end{aligned}$$

(e)

$$\begin{aligned} A &= \text{W}(N)/2 - \text{WIDTH}/2 - \text{PARKW}(N) \\ \text{BEGINPT} &= \text{COI} - \text{DBO}(8,9) + \sqrt{2} A \\ \text{DBNDRY} &= \text{BEGINPT} + 3\pi/4 (\text{RADIUS}) \\ &\quad + (\text{ZERO} - \text{DBO}(8,9)) \end{aligned}$$

(f)

FIGURE 16
DISCHARGE BOUNDARY COMPUTATIONS

Individually the above computations are quite simple. When performed for an entire network, however, they can become quite tedious and awkward. To resolve this difficulty the program PRESIG was developed as a companion to SIGNET. Its function is to compute a complete set of geometry data ready for input to SIGNET, based on easily acquired measurements from the network. This is accomplished by following the general computational procedure outlined above.

For purposes of discussion the logic of PRESIG may be divided into 3 areas as shown in Figure 17. Region I contains the primary input and initialization functions. Included in the input block is the verification of the input data, primarily achieved through checking card types and link numbers. Additional means for verification are obtained by printing out all input data.

Region II is concerned with the computations demonstrated on pages 57 through 63. The final product of this series is a tentative set of discharge boundaries and begin-turn points for each link.

In Region III this tentative set is compared to revised discharge boundaries and begin-turn points which are read from the input data. Any differences are changed to agree with the revised values. This option is provided to enable the user to easily correct any discrepancies which may arise between the PRESIG output

FIGURE 17
PRESIG FLOWCHART

and the actual conditions. The final function of PRESIG is to output, both on the line printer and card punch, all the geometry data needed for input to SIGNET.

A full description of the inputs to PRESIG and a listing of the PRESIG program code is provided in Appendix A.

The output from PRESIG is in two forms; printed and punched. Each form includes all the geometry data required for input to SIGNET; the punched output therefore may be used directly as a portion of the SIGNET input deck.

The following variables are output for each link:

STPLINE = coordinate of the stop line;

XLINK_{j,j=1,3} = length of each lane, j;

DBNDRY_{j,j=1,6} = discharge boundary for each
turning movement, j;

BEGINPT_{j,j=1,6} = beginning point of each turn,
j; and

RAD_{j,j=1,6} = radius of each turn, j.

SIGNET Program Structure

The SIGNET program consists of a set of 33 nested, closed subroutines. The program is modular in construction, making it easy to insert new routines or modify existing ones.

The structure of the SIGNET program is shown in Figure 18. It is seen that 28 routines occupy a fixed place in the program structure, while 5 routines serve auxiliary functions which are called upon at various places in the program. All routines are written in FORTRAN IV for operation under the MACE operating system on the CDC 6500 computer. A complete listing of the SIGNET program code and a full description of all inputs are given in Appendix B.

Tasks performed by the main program SIGNET include:

1. Input of all program data, program instructions, and link data, followed by formatting and writing of these data.
2. Primary initializing tasks, which include initializing all variables not connected with the statistical summary.
3. Activation of the traffic data input and initialization routines (INITRN and INIVOL) at the proper times as expressed on the program instruction cards.
4. Maintenance of the time loop, which includes control over the simulation time cycle and proper calling of the vehicle generation, vehicle update, and interlink transfer routines.
5. Program termination, which entails calling the final summary routine and writing final counts.

The Main Program Chain is shown in Figure 19.

FIGURE 18
SIGNET PROGRAM STRUCTURE

FIGURE 19
MAIN PROGRAM CHAIN

In the following paragraphs descriptions are given of the major routines. The subroutine flow charts are, for the most part, verbal, and in some instances have been generalized to a minor degree to enhance rapid understanding.

Secondary Initialization (INIT). This routine performs the initialization tasks necessary after fill-time has been completed in order to prepare for observe time. It is also used when reinitialization is called for at other times in the program (see Appendix B). Bookkeeping minimum values are reset to large numbers and counters, sums, sums of squares, and maximum values are cleared.

Turning Probability Input and Initialization (INITRN). In addition to reading the turning probabilities for each network and input link this routine initializes various variables associated with them. A full description of the inputs is given in Appendix B.

Traffic Volumes Input and Initialization (INIVOL). After reading the input volumes and truck percentages for each input link the lane flow rates, in vehicles per second, are set. Flow rates for short lanes (i.e. turning pockets) are set to zero.

Vehicle Printout (DEFINE). As detailed in Appendix B, a complete printout of each vehicle's position, velocity, acceleration, acceleration type, and turning

movement during each cycle may be specified. This routine unpacks the ICAR word, prepares the applicable variables, and writes them on the output file when called by the main program SIGNET.

Vehicle Generation (INPLINK). This routine generates, on a per-lane basis, each vehicle and its associated parameters; the generation time of the next vehicle on the lane is then determined. The flowchart is shown in Figure 20.

Vehicle Update (UPDATE). This routine (see Figure 21) determines the applicability of, and controls the transfer to, the subsidiary behavior routines; namely, lane change, general behavior, amber decision, red approach, right-turn, and left-turn. Additionally it controls the use of the signal update and bookkeeping routines. Based on the acceleration determined in the applicable behavior routine, it calculates the vehicle's velocity and position.

Lane Change (CHANGE). This routine (see Figure 22) determines for each lane if a potential lane changer exists, computes the size of the lane-change lag, and, by a random process, determines its acceptability. As appropriate, the vehicle is moved to the new lane and list modifications performed.

Traffic Signal Update (SIGNAL). This routine (see Figure 23) returns the current signal indication for the

FIGURE 20
VEHICLE GENERATION ROUTINE (INLINK)

FIGURE 21
VEHICLE UPDATE ROUTINE (UPDATE)

FIGURE 21, cont.

FIGURE 22 (cont.)

FIGURE 22 (cont.)

FIGURE 23

SIGNAL UPDATE ROUTINE (SIGNAL)

lane under consideration. It is therefore called once for each lane in the network each scan cycle. In its current implementation the routine will operate only in the fixed-time mode.

Amber Decision (AMBER). This routine (see Figure 24) determines if a vehicle approaching the intersection will stop or go through the intersection on amber. If it decides to go, control is transferred either to the general behavior routine or a turning routine, as determined by the vehicle's turning movement. Otherwise control is transferred to the red-approach routine. Decisions are made by a random process related to the stopping acceleration required.

Behavior Routines (GENBHR, REDAPP, RIGHT, LEFT). The four following routines determine the acceleration initiated in the current cycle for each vehicle. Transfer to these routines is controlled by the vehicle-update routine (UPDATE), and is related to both the traffic signal indication and the vehicle's turning movement. A limited amount of transfer occurs among the routines, notably to facilitate queue discharge.

The general-behavior routine GENBHR (see Figure 25) is applied to all vehicles not within the influence of an intersection and to those which are not turning. Contained within it is the logic related to queue discharge.

FIGURE 24

AMBER DECISION ROUTINE (AMBER)

FIGURE 25
GENERAL BEHAVIOR ROUTINE (GENBHR)

All vehicles confronted by a red signal indication or one not conforming to their turning movement are processed by the red-approach routine REDAPP (see Figure 26).

When a turning vehicle is within the influence of the intersection it is processed by either the right or left turn routine (RIGHT or LEFT), respectively. See Figures 27 and 28.

Left-Turn Conflict (CONLT). This routine (see Figure 29) determines if a vehicle on an opposing link is in conflict with the desired path of a left-turning vehicle. If yes, appropriate preventive action is taken after control has been transferred back to the left-turn behavior routine.

Cross-Street Conflict (CONST). This routine (see Figure 30) determines if a vehicle on a crossing link is in conflict with the path of a vehicle entering the intersection. This is most likely to occur immediately after the signal turns green. If such a conflict exists, appropriate preventive action is taken after control has been transferred back to the calling routine.

Left-Turn Lag and Gap Acceptance (GAP). This routine (see Figure 31) determines the size and acceptability of the time lag in traffic on the opposing approach for a left-turn vehicle. If the left-turning vehicle has rejected the lag, the size and acceptability

FIGURE 26
RED APPROACH ROUTINE (REDAPP)

FIGURE 26, cont

FIGURE 27
RIGHT TURN ROUTINE (RIGHT)

FIGURE 28

LEFT TURN ROUTINE (LEFT)

FIGURE 29
LEFT-TURN CONFLICT ROUTINE (CONLT)

FIGURE 30

CROSS STREET CONFLICT ROUTINE (CONST)

FIGURE 31

LAG & GAP ACCEPTANCE ROUTINE (GAP)

of successive time gaps in traffic on the opposing approach are determined. Decisions are made by a random process in relation to probability distributions of gap and lag acceptance.

Statistical Bookkeeping (BOOK). Each time a vehicle leaves its link this routine updates the applicable counters and sums, including total vehicles processed, total distance travelled, total travel time, total delay encountered, and sums of squares associated with each. Additionally the vehicle is marked for link transfer, which is accomplished in the routine OUTPT.

Queue Growth (QUEUE). This routine determines if a new vehicle has joined the queue on any of the links under consideration. Only those links specified on the Program Instruction Card (see Appendix B) are considered. A frequency analysis of queue lengths on these links is printed whenever a full statistical summary occurs.

Interlink Transfer (OUTPT). Whenever a vehicle crosses the discharge boundary of its link it is necessary to either process it out of the network or onto another link (the receiving link). This routine does so by resetting the vehicle's position as a function of its old-link position and the discharge boundary, and modifying the appropriate link lists. Additionally some link-dependent vehicle variables, such as turning movement and target velocity, are reset randomly.

Statistical Summary (SUMARY). This routine controls the computation, formatting, and writing on the output file of the summary statistics of network performance. It is executed at the end of the simulation run and at intermediate intervals as specified on the Program Instruction Card (see Appendix B). The routines STIX3 and STIX 6 are called by SUMARY to compute means and standard deviations and to compute, format, and write frequency tables of travel times and queue lengths.

Simulation Output

The simulation program output is a detailed statistical tabulation of traffic characteristics in the network. The first section of output is a listing of input data supplied to the simulation program. Included are program parameters, link parameters, link geometry descriptors, traffic signal settings, turning probabilities, and traffic volumes.

The second section of output presents traffic operation data. Traffic magnitude data related to generated volumes, link exit volumes, and volumes traced along the specified route are reported, followed by statistics computed from the individual vehicles' performance. Values are presented for the overall system and for individual links. The following performance measurements are reported:

Total Vehicle Miles: the total distance travelled on the link by all vehicles during the study period. Bookkeeping occurs only when a vehicle leaves its link; the distance travelled by vehicles still on the link at the time of the summary, therefore, is not included in this tally.

Total Delay (Given in seconds): A vehicle's undelayed travel time is based on its target velocity and distance travelled. The delay encountered by the vehicle, then, is the difference between its actual and undelayed travel time. Total delay for a link is the sum of all vehicles' delays on that link.

Average Delay (Given in seconds): the total delay divided by the number of vehicles leaving the link. In this study Average System Delay is taken to be the primary measurement of system performance.

Delay Standard Deviation: the standard deviation associated with the above average delay.

Average Delay (Given in seconds/vehicle-mile): an extended form of the average delay. Its purpose is to facilitate the comparison of average delays among links.

Total Travel Time (Given in seconds): the travel time for all vehicles.

Average Travel Time (Given in seconds/vehicle): the average time required to traverse the link under consideration.

Average Speed (Given in miles per hour): the average speed achieved by all vehicles over the entire link or network.

The final sections of the output report contain frequency tables of queue lengths on the specified links and travel-times on the route links.

CHAPTER IV

THE STREET NETWORK

A portion of the street network of West Lafayette, Indiana, was chosen as the test network. It consists primarily of the streets surrounding the Purdue University campus.

The major elements of the network are one major east-west arterial (S.R. 26), one major north-south arterial (U.S. 231), and three other north-south cross streets. The network thus defined includes sixteen signalized intersections. A street map of the network is shown in Figure 32.

A schematic diagram of the network is shown in Figure 33 with links numbered as required for use in the simulation model. The network consists of three types of links:

1. Input links that feed traffic into the network (links 60 through 84);
2. Network links that carry traffic from intersection to intersection within the network (links 26 through 59); and
3. Output links that remove traffic from the network (links 1 through 25).

FIGURE 32
STREET MAP OF THE TEST NETWORK

The SIGNET model does not accommodate minor streets not controlled by traffic signals. Two unusual characteristics of the network should be noted, however. First, the intersection of Northwestern Avenue and Wiggins Street is in actuality unsignalized. Because a major division of flow occurs at this point, however, a signal was assumed for use in the model which always indicated green. Thus no delay was caused to traffic but the required division was achieved.

Second, the intersection of Grant Street and North Street is such that the two east-west legs (North Street) are offset from each other. This situation is represented by employing two dummy links which form the offset. As in the intersection of Northwestern and Wiggins, the signals on these dummy links are always green.

Traffic signals in the network can be divided into three distinct systems. Those on State Street are interconnected and use a 65 second cycle length. The signal at Grant Street and Wood Street is also interconnected with this group and shares its cycle length. The signals on Northwestern Avenue, Wiggins Street, and Fowler Avenue are interconnected and use a 60 second cycle length. Finally, those city-operated signals at Stadium Avenue and University Street, Third Street and University Street, Grant Street and Central Avenue, and Grant Street and North Street operate on a variety of cycle lengths,

ranging from 55 to 70 seconds. There is no interconnection for these signals. All of the above cycle lengths are in effect during the study period, 4:00 p.m. to 5:30 p.m. The traffic signal settings are summarized in Appendix C.

For purposes of both simulation and signalization this network is considered to be independent of any other signals.

Information on the traffic characteristics of the network was required for two distinct phases of this study: (1) the validation of the simulation model; and (2) the application of the simulation program in testing alternative signal timing schemes. Common to both phases was the need for turning movement and lane distribution data. Traffic volumes were measured separately for each phase. The collection of travel time data is dealt with in Chapter 5, Model Testing and Validation.

Turning probabilities and lane distributions associated with each input and network link and input volumes associated with each input link are shown in Appendix C for average conditions during the study period, 4:00 to 5:30 p.m.

CHAPTER V

MODEL TESTING AND VALIDATION

The final phase in the development of the SIGNET model was testing and validation. It was desired to determine if the model represents real-world conditions with sufficient accuracy and if it is properly sensitive to changes in various input parameters.

The first phase of the sensitivity analysis was to compare travel times produced by the simulation model with equivalent travel times obtained in the field. It was presumed that a close correspondence between these times would indicate that the overall model was accurate. Individual portions of the model could then be tested in subsequent phases of the sensitivity analysis.

Actual travel times were obtained for each link on a route through the network between 4:00 and 4:45 p.m. using the floating-car method of driving. Input volumes were determined on a 15-minute basis for the same period of time. Due to limited resources it was possible to travel the route only 3 times during each 15-minute period, thereby yielding only 3 sets of travel time data for each set of input volume data.

Four simulation runs were made using different seeds for the random number generator and the input volumes described above. Histograms were made for each link in the route relating simulated travel times of vehicles traveling the route to the average frequency of occurrence. The actual travel times were then superimposed on these histograms to facilitate comparison. Several histograms are shown in Figures 34, 35, and 36.

Ideally the distribution of actual travel times should match the distribution of simulated travel times; if these distributions did match closely, it would be difficult to reject the simulation results as invalid. If the number of observations used in estimating both distributions were large it would be possible to apply standard statistical techniques in analyzing this goodness of fit. In this case, however, only 3 observations of actual travel times were available for each comparison. Standard statistical techniques break down under these conditions, making a qualitative analysis necessary.

Examination of the histograms reveals several general characteristics of the distributions. In all cases, the actual travel times fall within the distributions of simulated travel times, showing that the simulated travel times approximate actual travel times. At this point two cases arise in the comparison, both being dependent upon the relationship of demand volume and capacity. For those links on which demand volume does not approach capacity

(a)

(b)

(c)

FIGURE 34
COMPARISON OF TRAVEL TIMES, LINK 28

(a)

(b)

(c)

FIGURE 35

COMPARISON OF TRAVEL TIMES, LINK 56

FIGURE 36
COMPARISON OF TRAVEL TIMES, LINK 59

(i.e. Link 28, Figure 34), only one peak appears in the histogram, implying that no vehicles are forced to wait through more than one red indication. In this case it would be expected that an average vehicle would achieve an average travel time over the link; therefore, since the floating car used in the travel time collection is assumed to be an average vehicle, it would be expected that the actual travel times would fall approximately on the mean of the simulated travel times. It can be seen in Figure 34 that the actual travel times did approximate the mean. The overall model would seem to be performing relatively well in this instance.

On some links (i.e. Links 56 and 59, Figures 35 and 36), demand volume does approach capacity. In this case, some vehicles are forced to wait through more than one red indication, resulting in the two peaks shown. Under these circumstances it would be unrealistic to expect the average actual travel time to approximate the mean, since the mean could fall during the red indication. Instead, it is necessary to examine the relationship of the actual travel times to the peaks. It can be seen that, in general, the peaks are fairly well approximated by the average travel times. It was therefore felt that the simulation model performed well in this case.

Based on the above analysis, it was decided that the overall model was an acceptable one. That is, the entire

model working as a whole produced acceptable results. It remained to be shown, however, that individual portions of the model were also working. This phase of the testing is described below.

The objective of the second phase of the testing was to determine the sensitivity of the model output to changes in various input parameters. An inherent characteristic of computer simulation, the ability to control all input parameters precisely, enables the investigator to change only one parameter and thereby to determine its influence on the output. This phase of the sensitivity analysis had practical usefulness for two reasons:

1. It provided additional data which could be used in confirming the reasonableness of the model. The direction of change in the output caused by modifying a parameter could be examined to see if it was logical, and the magnitude of the change could be checked for reasonableness.

2. It indicated whether particular portions of the model logic were indeed operating. If a parameter modification produced no change where one was expected, there was good evidence of logic flaws in the model.

Sensitivity tests were made on 9 separate effects involving 7 input parameters. For all of the effects an attempt was made to test throughout the range of magnitudes

which might reasonably occur in real life. The simulation run of 4:00 to 4:15 p.m. was used throughout the sensitivity analysis as a base for comparison purposes. The following effects were tested:

1. The effect of variation in input volume on average delay. Greater volumes were expected to increase delay.

2. The effect of variation in input volume on average travel time and average speed. Greater volumes were expected to increase average travel time and thus decrease average speed.

3. The effect of variation in input volume on average queue length. Greater volumes were expected to increase average queue length.

4. The effect of variation in target velocity on average travel time and average speed. Greater target velocity was expected to decrease average travel time and thus increase average speed.

5. The effect of variation in effective length on average travel time and average speed. Greater effective length was expected to increase average travel time and thus decrease average speed.

6. The effect of variation in the truck percentage on average travel time and average speed. A higher truck percentage was expected to increase travel time and thus decrease average speed.

7. The effect of variation in the maximum desired acceleration on average travel time and average speed. A higher maximum desired acceleration was expected to decrease average travel time and thus increase average speed.

8. The effect of variation in the characteristic speed (a_0) on average travel time and average speed. A higher characteristic speed was expected to reduce average travel time and thus increase average speed.

9. The effect of variation in the proportionality coefficient (K) on average travel time and average speed. A higher proportionality coefficient was expected to result in decreased travel time and thus increased average speed.

Detailed results of the sensitivity tests are exhibited in Figures 37 through 52 and Tables 6 through 21. Four runs were made for each value of the input parameters. These provided a good data base upon which to build the Analysis of Variance for each effect.

It was desirable to know not only that the simulation output varied with changes in the input parameters, but also what form the variation took (i.e., if the response curve was linear, quadratic, or cubic in nature). If the output variation levelled out at some point this response could then be compared with expectations, thereby providing added evidence of the

FIGURE 37

EFFECT OF VOLUME ON AVERAGE DELAY

TABLE 6
ANALYSIS OF VARIANCE
EFFECT OF VOLUME ON AVERAGE DELAY

SOURCE OF VARIATION	SUM OF SQUARES	D.F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	43631.71	3	14543.90	78.96 *
LINEAR EFFECT	39856.58	1	39856.58	216.3 *
QUADRATIC EFFECT	3747.26	1	3747.26	20.4 *
CUBIC EFFECT	27.86	1	27.86	0.161
EXPERIMENTAL ERROR	3683.79	20	184.19	
TOTAL	47315.50	23		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$.
CRITICAL REGION: $F > 2.97$

FIGURE 38

EFFECT OF VOLUME ON AVERAGE TRAVEL TIME

TABLE 7
ANALYSIS OF VARIANCE
EFFECT OF VOLUME ON AVERAGE TRAVEL TIME

SOURCE OF VARIATION	SUM OF SQUARES	D.F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	44707.17	3	14902.37	77.98 *
LINEAR EFFECT	40990.93	1	40990.93	213.9 *
QUADRATIC EFFECT	3715.69	1	3715.69	19.4 *
CUBIC EFFECT	0.501	1	0.501	0.003
EXPERIMENTAL ERROR	3822.04	20	191.10	
TOTAL	48529.16	23		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$.CRITICAL REGION: $F > 2.97$

FIGURE 39
EFFECT OF VOLUME ON AVERAGE SPEED

TABLE 8
ANALYSIS OF VARIANCE
EFFECT OF VOLUME ON AVERAGE SPEED

SOURCE OF VARIATION	SUM OF SQUARES	D.F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	107.62	3	35.87	132.8 *
LINEAR EFFECT	103.66	1	103.66	383.3 *
QUADRATIC EFFECT	3.85	1	3.85	14.3 *
CUBIC EFFECT	0.10	1	0.10	0.38
EXPERIMENTAL ERROR	5.40	20	0.270	
TOTAL	113.02	23		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$.
CRITICAL REGION: $F > 2.97$

FIGURE 40

EFFECT OF VOLUME ON AVERAGE QUEUE LENGTH (LINK 57)

TABLE 9
ANALYSIS OF VARIANCE

EFFECT OF VOLUME ON AVERAGE QUEUE LENGTH (LINK 57)

SOURCE OF VARIATION	SUM OF SQUARES	D.F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	12.152	3	4.051	128.0 *
LINEAR EFFECT	12.142	1	12.142	384.5 *
QUADRATIC EFFECT	0.010	1	0.010	0.326
CUBIC EFFECT	0.0002	1	0.0002	0.005
EXPERIMENTAL ERROR	0.538	17	0.0317	
TOTAL	12.690	20		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$.
CRITICAL REGION: $F > 3.03$

FIGURE 41

EFFECT OF TARGET VELOCITY ON AVERAGE TRAVEL TIME

TABLE 10
ANALYSIS OF VARIANCE

EFFECT OF TARGET VELOCITY ON AVERAGE TRAVEL TIME

SOURCE OF VARIATION	SUM OF SQUARES	D.F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	638599.4	3	212866.5	423.6 *
LINEAR EFFECT	435057.6	1	435057.6	865.8 *
QUADRATIC EFFECT	174491.2	1	174491.2	347.3 *
CUBIC EFFECT	29050.6	1	29050.6	57.8 *
EXPERIMENTAL ERROR	9547.2	19	502.5	
TOTAL	648146.6	22		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$.
CRITICAL REGION: $F > 2.99$

FIGURE 42

EFFECT OF TARGET VELOCITY ON AVERAGE SPEED

TABLE II
ANALYSIS OF VARIANCE

EFFECT OF TARGET VELOCITY ON AVERAGE SPEED

SOURCE OF VARIATION	SUM OF SQUARES	D. F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	422.56	3	140.85	233.6 *
LINEAR EFFECT	377.64	1	377.64	626.2 *
QUADRATIC EFFECT	44.91	1	44.91	74.5 *
CUBIC EFFECT	0.008	1	0.008	0.014
EXPERIMENTAL ERROR	11.46	19	0.603	
TOTAL	434.02	22		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$.CRITICAL REGION: $F > 2.99$

FIGURE 43

EFFECT OF EFFECTIVE LENGTH ON AVERAGE TRAVEL TIME

TABLE 12

ANALYSIS OF VARIANCE

EFFECT OF EFFECTIVE LENGTH ON AVERAGE TRAVEL TIME

SOURCE OF VARIATION	SUM OF SQUARES	D. F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	13631.18	2	6815.59	30.5 *
LINEAR EFFECT	13416.86	1	13416.86	60.2 *
QUADRATIC EFFECT	214.32	1	214.32	0.96
EXPERIMENTAL ERROR	2008.72	9	223.19	
TOTAL	15639.90	11		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$.CRITICAL REGION: $F > 3.23$

FIGURE 44

EFFECT OF EFFECTIVE LENGTH ON AVERAGE SPEED

TABLE 13

ANALYSIS OF VARIANCE

EFFECT OF EFFECTIVE LENGTH ON AVERAGE SPEED

SOURCE OF VARIATION	SUM OF SQUARES	D.F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	35.363	2	17.681	43.9 *
LINEAR EFFECT	35.322	1	35.322	87.7 *
QUADRATIC EFFECT	0.041	1	0.041	0.101
EXPERIMENTAL ERROR	3.625	9	0.403	
TOTAL	38.988	11		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$.
 CRITICAL REGION: $F > 3.36$

FIGURE 45
EFFECT OF TRUCKS ON AVERAGE TRAVEL TIME

TABLE 14
ANALYSIS OF VARIANCE
EFFECT OF TRUCKS ON AVERAGE TRAVEL TIME

SOURCE OF VARIATION	SUM OF SQUARES	D.F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	9406.99	2	4703.49	23.4 *
LINEAR EFFECT	8851.82	1	8851.82	44.1 *
QUADRATIC EFFECT	555.17	1	555.17	2.76
EXPERIMENTAL ERROR	1810.03	9	201.11	
TOTAL	11217.02	11		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$.
CRITICAL REGION: $F > 3.36$

FIGURE 46

EFFECT OF TRUCKS ON AVERAGE SPEED

TABLE 15
ANALYSIS OF VARIANCE
EFFECT OF TRUCKS ON AVERAGE SPEED

SOURCE OF VARIATION	SUM OF SQUARES	D.F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	16.815	2	8.408	24.5 *
LINEAR EFFECT	16.188	1	16.188	47.2 *
QUADRATIC EFFECT	0.627	1	0.627	1.83
EXPERIMENTAL ERROR	3.084	9	0.343	
TOTAL	19.899	11		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$.
CRITICAL REGION: $F > 3.36$

FIGURE 47

EFFECT OF MAXIMUM DESIRED ACCELERATION
ON AVERAGE TRAVEL TIME

TABLE 16
ANALYSIS OF VARIANCE
EFFECT OF MAXIMUM DESIRED ACCELERATION
ON AVERAGE TRAVEL TIME

SOURCE OF VARIATION	SUM OF SQUARES	DEGREES OF FREEDOM	MEAN SQUARE	F RATIO
AMONG TREATMENTS	32086.9	3	10695.6	19.1 *
LINEAR EFFECT	22890.4	1	22890.4	40.9 *
QUADRATIC EFFECT	7399.0	1	1399.0	13.2 *
CUBIC EFFECT	1797.5	1	1797.5	3.21 *
EXPERIMENTAL ERROR	6714.8	12	559.6	
TOTAL	38801.7	15		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$.

CRITICAL REGION: $F > 3.18$

FIGURE 48
EFFECT OF MAXIMUM DESIRED ACCELERATION ON
AVERAGE SPEED

TABLE 17
ANALYSIS OF VARIANCE,
EFFECT OF MAXIMUM DESIRED ACCELERATION ON
AVERAGE SPEED

SOURCE OF VARIATION	SUM OF SQUARES	D.F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	36.93	3	12.31	14.3 *
LINEAR EFFECT	16.52	1	16.52	19.2 *
QUADRATIC EFFECT	19.56	1	19.56	22.8 *
CUBIC EFFECT	.85	1	.85	.99
EXPERIMENTAL ERROR	10.31	12	.859	
TOTAL	47.24	15		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$
CRITICAL REGION: $F > 3.18$

FIGURE 49

EFFECT OF CHARACTERISTIC SPEED (a_0) ON AVERAGE TRAVEL TIME

TABLE 18

ANALYSIS OF VARIANCE

EFFECT OF CHARACTERISTIC SPEED (a_0) ON AVERAGE TRAVEL TIME

SOURCE OF VARIATION	SUM OF SQUARES	D.F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	8787.66	2	4393.03	15.9 *
LINEAR EFFECT	8432.46	1	8432.46	30.4 *
QUADRATIC EFFECT	355.20	1	355.20	1.28
EXPERIMENTAL ERROR	2488.82	9	276.54	
TOTAL	11276.48	11		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$.
 CRITICAL REGION: $F > 3.23$

FIGURE 50

EFFECT OF CHARACTERISTIC SPEED (A_o)
ON AVERAGE SPEED

TABLE 19
ANALYSIS OF VARIANCE,
EFFECT OF CHARACTERISTIC SPEED ON AVERAGE SPEED

SOURCE OF VARIATION	SUM OF SQUARES	D. F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	18.909	2	9.455	16.8 *
LINEAR EFFECT	18.301	1	18.301	32.5 *
QUADRATIC EFFECT	.608	1	.608	1.08
EXPERIMENTAL ERROR	5.073	9		
TOTAL	23.982	11		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$

CRITICAL REGION: $F > 3.36$

FIGURE 51
EFFECT OF PROPORTIONALITY COEFFICIENT (K) ON
AVERAGE TRAVEL TIME

TABLE 20
ANALYSIS OF VARIANCE,
EFFECT OF PROPORTIONALITY COEFFICIENT ON AVERAGE TRAVEL TIME

SOURCE OF VARIATION	SUM OF SQUARES	D.F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	9601.90	3	3200.63	31.20 *
LINEAR EFFECT	9031.83	1	9031.83	88.00 *
QUADRATIC EFFECT	437.65	1	437.65	4.27 *
CUBIC EFFECT	132.42	1	132.42	1.29
EXPERIMENTAL ERROR	1231.04	12	102.59	
TOTAL	10832.93	15		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$

CRITICAL REGION: $F > 3.18$

FIGURE 52
EFFECT OF PROPORTIONALITY COEFFICIENT (K) ON
AVERAGE SPEED

TABLE 21
ANALYSIS OF VARIANCE ,
EFFECT OF PROPORTIONALITY COEFFICIENT ON AVERAGE SPEED

SOURCE OF VARIATION	SUM OF SQUARES	D.F.	MEAN SQUARE	F RATIO
AMONG TREATMENTS	29.268	3	9.756	26.6 *
LINEAR EFFECT	28.375	1	28.375	77.3 *
QUADRATIC EFFECT	.518	1	.518	1.41
CUBIC EFFECT	.374	1	.374	1.02
EXPERIMENTAL ERROR	4.403	12		
TOTAL	33.671	15		

* EFFECT IS SIGNIFICANT AT $\alpha = 0.10$
CRITICAL REGION: $F > 3.18$

model's acceptability. For this reason the treatment sum of squares was broken down into sums of squares associated with linear, quadratic, and cubic effects using the orthogonal polynomials method. The results are shown for each test in Tables 6 through 21.

It can be seen from Figures 37, 38 and 39 that volume changes resulted in significantly less variation in average delay, average travel time, and average speed under low volume conditions than they did under high volumes. This flattening of the response curve was not unexpected, since under these low volume conditions delays caused by signals become much more significant than those caused by other vehicles.

Target velocity, the characteristics speed (a_0), the proportionality coefficient (K), and effective length are the primary parameters affecting queue discharge and thereby the total network performance. Decreases in target velocity cause slower discharge and lower speeds. (See Figures 41 and 42). As expected, increases are somewhat negated by the maximum desired acceleration (See Figures 47 and 48), since it puts a figurative ceiling on acceleration rates, and therefore causes a flattening of the response curve. Similarly, increases in the proportionality coefficient and the characteristic speed (Figures 49 through 52) cause faster queue discharge, while increased effective length (Figures 43 and 44) slows queue discharge

as was expected. Decreases in the maximum desired acceleration also slow queue discharge as expected. Studies by Greenshields (17) indicate that nearly all vehicles' starting accelerations are less than 6 ft/sec^2 . Higher values for the maximum desired acceleration would therefore have little effect in slowing queue discharge. This expectation is paralleled in the response curve.

The way in which each of the above sensitivity tests performed as expected provided strong evidence that the individual portions of the SIGNET model were valid. These results, combined with the results of the travel time tests, served to substantially increase the author's confidence in the model.

CHAPTER VI

SIGNAL TIMING STUDIES

As stated in Chapter I, the ultimate objective of this research was to develop a tool which could be used to compare alternative traffic signal control schemes. The foregoing chapters have related to the development of this tool. This chapter is concerned with the application of the simulation program to testing these various schemes.

Several methods for optimizing signal systems were studied and reported on by Wagner, Barnes, and Gerlough (36). Among them were SIGOP, the British Combination Method, the Volume Priority Method, the Preferential Street Method, the Webster Method, and the Delay/Difference-of-Offset Method. Their findings indicate that application of any of these methods will produce significant upgrading of network performance.

Three of these techniques were used to develop alternative control schemes for this research. The Webster Method was used to determine optimal cycle lengths and splits for each intersection; the Delay/Difference-of-Offset Method was used to determine optimal offsets for each link; and the Preferential Street Method was used to determine the final offset plan for the network. Because full

descriptions of each of these methods appear elsewhere only a summary of each is presented below.

First published in a monograph in 1958 (38), the Webster method of determining optimum cycle lengths and splits for fixed time signals is rapidly becoming a standard method. Webster used a combination of theory and computer simulation to deduce a formula for average delay per vehicle at a signalized intersection. From that formula expressions were derived for the cycle lengths and splits which minimize total delay at the intersection. For a two-phase intersection:

$$C_0 = \frac{1.5L + 5}{1 - Y_A - Y_B}$$

where

C_0 = optimum cycle length, in seconds;

L = lost time per cycle, in seconds; and

Y_A and Y_B = the maximum ratios of single lane flow to saturation flow for phases A and B.

The splits should be established such that

$$\frac{GE_A}{GE_B} = \frac{Y_A}{Y_B}$$

where

GE_A and GE_B = effective green times of phases A and B, in seconds.

In the normal approach to strategic timing of signal networks the maximum cycle length determined above would be

taken as the network cycle length. This approach was used in the determination of one alternative, Alternative A, of this study. The difficulty with this practice is, however, that excessive cycle lengths may be used for large, virtually unrelated areas of the network. Wagner, Barnes, and Gerlough (36) found that under some conditions it is possible to mix cycle lengths and thereby optimize operation in two or more areas. Alternative B of this study was therefore designed to employ different cycle lengths on each of the two primary arterials, State Street and Northwestern Avenue. Simulation testing would be used to determine if the lack of fixed relationships between some adjacent signals adversely affected system performance.

First developed at the Road Research Laboratory in England for strategic optimization of offsets, the Delay/Difference-of-Offset Method was refined by Wagner, Gerlough, and Barnes (15). The traffic flows, the common cycle length, and the apportionment of green at each signal are given, and it is assumed that the delay to traffic along any link depends solely on the difference between the offsets of the signals at each end of the link. This offset optimization technique has perhaps the soundest logical base of the methods developed to date, because delay caused by queueing is taken under consideration and systematically reduced. A complete description of the technique may be found in N.C.H.R.P. Report 124 (36).

The Preferential Street Method is a relatively straightforward approach to combining the optimal offsets determined above into the final offset plan. It is one of the most widely used means for developing network timing plans, since the engineer uses his judgement and the information at hand to decide which streets are to be favored. The preferential streets must be selected in a manner which avoids closure of any portion of the network; in a grid network, this would entail selecting any number of parallel streets and one cross-street. The intersections along each preferential street are then assigned offsets as determined from some systematic offset determination technique, such as the Delay/Difference-of-Offset technique described above.

Before evaluating the network performance under each of the various signal control alternatives it was necessary to settle upon some criterion upon which to base this evaluation. Many criteria are available to the researcher, such as total travel time, mean travel time, maximum travel time, total system delay, mean system delay, maximum delay, total stopped delay, mean stopped delay, maximum stopped delay, mean queue length, maximum queue length, mean travel speed, and minimum travel speed.

Total or mean system delay has been widely accepted in theory as an appropriate measure of effectiveness. System delay is considered to be equal to the difference

between actual travel time and some theoretical optimum or undelayed travel time. Berry (5) assumed an undelayed trip to be one made at the average free flow speed. This assumption could be in error, however, since free flow speed does not necessarily correspond exactly with a vehicle's target velocity.

In the simulation model mean system delay is used as the primary measure of effectiveness of network performance. Undelayed travel time is calculated uniquely for each vehicle, based on the vehicle's target velocity, and delay is calculated as the difference between this undelayed travel time and the vehicle's actual travel time. Mean system delay is the average of all vehicles' delay.

A total of three different signal timing plans were tested for their effects on network performance. The existing condition was used as a basis for comparison, while Alternatives A and B were tested for any improvement in operation which they might produce. The simulation program was run three times for each condition using a different random number seed each time. The results are exhibited in Table 22.

Two statistical tests were performed which compared the operational effectiveness under Alternatives A and B with the effectiveness under the existing timing plan:

1. The variances of Alternatives A and B were tested for equality with the variance of the existing

TABLE 22

Average Delay Caused by Signal Timing Alternatives
sec./veh.¹

	<u>Existing Condition</u>	<u>Alternative A</u>	<u>Alternative B</u>
Run No. 1	19.47	22.65	19.21
Run No. 2	19.32	23.73	21.17
Run No. 3	20.07	21.09	21.26
Mean	19.62	22.49	20.55

¹Delays are computed on a link basis. Therefore the delays shown are the average delays encountered in negotiating the length of one link.

condition; i.e., the hypotheses $H_0: \sigma_A^2 = \sigma_E^2$ and $H_0: \sigma_B^2 = \sigma_E^2$ were tested, as opposed to the alternatives $H_1: \sigma_A^2 \neq \sigma_E^2$ and $H_1: \sigma_B^2 \neq \sigma_E^2$ respectively. As defined by Ostle (30), the hypothesis H_0 can be accepted if

$$\chi^2_{(\alpha/2, n-1)} < \frac{\sum (X - \bar{X})^2}{\sigma_0^2} < \chi^2_{(1-\alpha/2, n-1)}.$$

Otherwise, H_0 must be rejected. In this case,

$$\frac{\sum (X - \bar{X})_A^2}{\sigma_E^2} = \frac{3.5145}{.1666} = 21.1$$

and

$$\frac{\sum (X - \bar{X})_B^2}{\sigma_E^2} = \frac{2.686}{.1666} = 16.1.$$

At the $\alpha = 0.10$ level, both values were greater than the critical χ^2 , so the hypotheses were rejected. The variances, then, could not be considered equal.

2. The means of Alternatives A and B were tested for equality with the mean of the existing condition; i.e., the hypotheses $H_0: \mu_A = \mu_B$ and $H_0: \mu_B = \mu_E$ were tested, as opposed to the alternatives $H_1: \mu_A \neq \mu_E$ and $H_1: \mu_B \neq \mu_E$. A reasonably good approximate procedure for cases involving unequal variances is defined by Ostle. For the sample hypotheses $H_0: \mu_1 = \mu_2$, compute

$$t' = (\bar{X}_1 - \bar{X}_2) / (s_1^2/n_1 + s_2^2/n_2)^{1/2}$$

and reject if

$$t' \leq -(w_1 t_1 + w_2 t_2) / (w_1 + w_2)$$

or if

$$t' \geq (w_1 t_1 + w_2 t_2) / (w_1 + w_2)$$

where

$$w_1 = s_1^2/n_1;$$

$$w_2 = s_2^2/n_2;$$

$$t_1 = t_{(1-\alpha/2, n_1-1)}; \text{ and}$$

$$t_2 = t_{(1-\alpha/2, n_2-1)}.$$

In this case,

$$t'_A = \frac{(22.488 - 19.619)}{(\frac{1.757}{3} + \frac{.1666}{3})^{1/2}} = 3.58$$

and

$$t'_B = \frac{(20.546 - 19.619)}{(\frac{1.343}{3} + \frac{1.666}{3})^{1/2}} = 1.31.$$

The weighted average of t_1 and t_2 was 2.920. Therefore $H_0: \mu_A = \mu_E$ was rejected while $H_0: \mu_B = \mu_E$ could not be rejected. In other words, the mean delay associated with Alternative A could not be considered equal to the mean delay associated with the existing condition. The mean delay of Alternative B, however, could not be said to be unequal to the mean delay of the existing condition.

It is not surprising that no significant improvement could be made on the existing condition, for that timing plan was derived over a period of time using both

analytical and empirical (i. e., field adjustment) techniques. It cannot be said that a design procedure incorporating simulation will consistently produce a better plan than a procedure using other techniques, for that is not the point of using simulation. Instead, simulation can help arrive at the optimal timing plan more quickly, primarily by providing an easily accessible medium for study and experimentation. This aspect of the project was demonstrated by the fact that it took only a few hours to develop, implement, and test the two alternatives in this study. The benefits of using simulation were further demonstrated by the fact that a significant difference was demonstrated between two plans, thereby indicating which should be implemented.

There is also a possibility that the techniques used to determine the timing plans were inadequate in some respects. The detection of such inadequacies was not in the scope of this project, nor would it be under normal design conditions. Inadequacies in the design techniques could contribute to the lack of improvement encountered in this instance, however; since simulation is merely a tool for choosing among the alternative timing plans, there is no way of overcoming such shortcomings using just simulation.

The foregoing was intended to give a somewhat detailed description of the use of the simulation program in the

design of signal timing plans. The steps required in this type of design may be summarized as follows:

1. Comprehensive data must be gathered on both the network geometry and traffic characteristics. Turning movements and input volumes are especially critical, and should be determined as precisely as resources allow.

2. The simulation program should be calibrated to the network by means of a travel time test and at least a cursory sensitivity analysis.

3. Alternative signal timing plans should be developed using systematic study and analysis techniques.

4. These alternative plans should be tested, using several simulation runs and standard statistical techniques to determine the significance of the results.

5. The final selection of one alternative should be made on the basis of one or more predefined criteria.

The above steps are predicated on the assumption that the goals, objectives, and constraints of the study have already been determined and that simulation is the most effective means available to achieve these objectives.

It is difficult at this point to determine how much time the use of simulation could save the traffic engineer, since it is unknown how long the field adjustment of a typical improvement would take. In this project preparation and coding of the input data took between 15 and 20 hours, although much of this data would also

normally be required for designing the timing plans. It is felt, therefore, that this figure compares favorably with the time required to undertake field adjustment. The possibility of using the simulation in studying other phases of traffic control, such as turning movement prohibition, parking regulation, and unbalanced lane operation, also tends to make the use of simulation more attractive. Such studies were not, however, attempted in this research.

CHAPTER VII

CONCLUSION

The overall accomplishments of this project conformed well with the original research plan. A simulation model, SIGNET, was developed to reproduce traffic flow under laboratory conditions. The program was fairly economical, achieving a 6.5 to 1 real-time to simulation-time ratio. A sensitivity analysis indicated that the model simulated traffic flows quite accurately.

The model was used to evaluate several alternative timing plans, and indicated by a significant difference in mean delay between two of these plans that it is indeed sensitive to variations in signal timing. No improvement could be made over the operation of the existing network. This does not necessarily indicate that the model itself is weak, however, because the traffic control theory used to derive the alternatives could be inadequate; under such circumstances no improvement could be expected. It was demonstrated that the incorporation of simulation into the design procedure could make that procedure quicker, at the same time resulting in the production of a more efficient timing plan. In effect this was the overall goal of the project.

The use of the SIGNET model is not restricted to signal timing studies. In fact, the effects of many other traffic engineering measures may be evaluated by varying the program inputs. Such measures could include turning movement prohibitions, parking restrictions, unbalanced lane operation, and one-way street operation. A comprehensive set of improvements to the network can thus be easily evaluated. This capability is important because traffic signal optimization cannot be considered to be the sole means of obtaining maximum network efficiency.

Of primary importance in future work involving SIGNET should be more intensive validation studies. The model was quite accurate in reproducing traffic conditions during the validation phase of this study; it would be desirable, however, to test the model much more if more comprehensive traffic data could be obtained. A constant monitoring of input volumes during collection of travel data would be especially beneficial in this regard, as would well defined turning probabilities.

The model itself could be somewhat improved also. Provision for pedestrian conflicts, right-turns on red, and traffic actuated signals would result in increased versatility. It is doubtful, however, that such additions would affect overall model performance substantially.

BIBLIOGRAPHY

1. AMERICAN SOCIETY OF CIVIL ENGINEERS, Highway Engineering, 3rd ed., McGraw-Hill, New York, 1954.
2. AMERICAN SOCIETY OF CIVIL ENGINEERS, Highway Engineering, 4th ed., McGraw-Hill, New York, 1964.
3. AMERICAN SOCIETY OF CIVIL ENGINEERS, Highway Engineering, 5th ed., McGraw-Hill, New York, 1974.
4. AMERICAN SOCIETY OF CIVIL ENGINEERS, Highway Engineering, 6th ed., McGraw-Hill, New York, 1984.
5. AMERICAN SOCIETY OF CIVIL ENGINEERS, Highway Engineering, 7th ed., McGraw-Hill, New York, 1994.
6. AMERICAN SOCIETY OF CIVIL ENGINEERS, Highway Engineering, 8th ed., McGraw-Hill, New York, 2004.
7. AMERICAN SOCIETY OF CIVIL ENGINEERS, Highway Engineering, 9th ed., McGraw-Hill, New York, 2014.
8. AMERICAN SOCIETY OF CIVIL ENGINEERS, Highway Engineering, 10th ed., McGraw-Hill, New York, 2024.
9. AMERICAN SOCIETY OF CIVIL ENGINEERS, Highway Engineering, 11th ed., McGraw-Hill, New York, 2034.
10. AMERICAN SOCIETY OF CIVIL ENGINEERS, Highway Engineering, 12th ed., McGraw-Hill, New York, 2044.

BIBLIOGRAPHY

1. American Association of State Highway Officials, A Policy on Geometric Design of Rural Highways, Washington, D. C., 1965.
2. Bartle, R. M., Val Skoro, and Gerlough, D. L., "Starting Delay and Time Spacing of Vehicles Entering Signalized Intersection," Effects of Traffic Control on Street Capacity, Bulletin 112, Highway Research Board, pp. 33-41, 1956.
3. Bellis, W. R., "Capacity of Traffic Signals and Traffic Signal Timing," Increasing Traffic Capacity of Arterial Streets, Bulletin 271, Highway Research Board, pp. 45-67, 1960.
4. Benhard, F. G., "Simulation of a Traffic Intersection on a Digital Computer," Master of Science Thesis, University of California, Los Angeles, June, 1959.
5. Berry, D. S., "Field Measurement of Delay at Signalized Intersections," Proceedings, Highway Research Board, Vol. 35, pp. 505-522, 1956.
6. Blackman, A. R., and Crawford, A., "Driver Behavior During Amber Period of Traffic Lights: 6. Road Observations," Unpublished Research Note LN/106/ARB, AC, British Road Research Laboratory, May, 1962.
7. Brooks, W. D., Designing Arterial Progressions Using a Digital Computer, International Business Machines Corporation.
8. Carstens, R. L., "Some Traffic Parameters at Signalized Intersections," Traffic Engineering, Vol. 41, pp. 33-36, August, 1971.
9. Chang, A., Synchronization of Traffic Signals in Grid Networks, International Business Machines Corporation, San Jose Research Laboratory.
10. Dawson, R. F., and Chimini, L. A., "The Hyperlang Probability Distribution - A Generalized Traffic Headway Model," Record 230, Highway Research Board, pp. 1-14, 1968.

11. Gerlough, D. L., "Simulation of Freeway Traffic by an Electronic Computer," Proceedings, Highway Research Board, Vol. 35, pp. 543-547, 1956.
12. Gerlough, D. L., "Simulation of Traffic Flow," An Introduction to Traffic Flow Theory, Special Report 79, Highway Research Board, pp. 97-118, 1964.
13. Gerlough, D. L., and Wagner, F. A., "Improved Criteria for Traffic Signals at Individual Intersections, Interim Report," Report No. 3, National Cooperative Highway Research Program, 1964.
14. Gerlough, D. L., and Wagner, F. A., "Improved Criteria for Traffic Signals at Individual Intersections," Report No. 32, National Cooperative Highway Research Program, 1967.
15. Gerlough, D. L., Wagner, F. A., and Barnes, F. C., "Improved Criteria for Traffic Signal Systems on Urban Arterials," Report No. 73, National Cooperative Highway Research Program, 1969.
16. Goode, H. H., Polmar, C. H., and Wright, J. B., "The Use of a Digital Computer to Model a Signalized Intersection," Proceedings, Highway Research Board, Vol. 35, pp. 548-557, 1956.
17. Greenshields, B. D., "Some Time Space Relationships of Traffic in Urban Areas," Proceedings, Institute of Traffic Engineers, pp. 114-134, 1946.
18. Greenshields, B. D., Schapiro, D., and Erickson, E. L., Traffic Performance at Urban Street Intersections, Technical Report No. 1, Bureau of Highway Traffic, Yale University, 1947.
19. Hammond, H. F., "Report of Committee on Safe Approach Speeds at Intersections," Proceedings, Highway Research Board, Vol. 20, p. 657, 1940.
20. Heathington, K. W., and Rath, G. J., "Computer Simulation for Transportation Problems," Traffic Quarterly, April, 1968, pp. 271-281.
21. Herman, R., and Potts, R. B., "Single Lane Traffic Theory and Experiment," Theory of Traffic Flow, Elsevier, pp. 120-146, 1961.

22. Herman, R., and Rothery, R. W., "Car Following and Steady State Flow," Proceedings of the International Symposium on Theory of Traffic Flow, London, pp. 1-11, 1963.
23. Hillier, J. A., "Glasgow's Experiment in Area Traffic Control," Traffic Engineering and Control, Vol. 7, No. 8, pp. 502-509, 1965. Appendix, Vol. 7, No. 9, pp. 569-571, 1966.
24. Institute of Traffic Engineers, Traffic Engineering Handbook, Washington, D. C., 1965.
25. Kell, J. H., "Analyzing Vehicular Delay at Intersections through Simulation," Theory of Traffic Flow, Bulletin 356, Highway Research Board, 1962.
26. Lewis, R. M., and Michael, H. L., "The Simulation of Traffic Flow to Obtain Volume Warrants for Intersection Control," Ph.D. Thesis, Purdue University, Lafayette, Indiana, 1962.
27. Lewis, R. M., "The Simulation of Vehicular Traffic at an Intersection on a Digital Electronic Computer," MCE Thesis, Rensselaer Polytechnic Institute, Troy, New York, May, 1956.
28. Little, J. D. C., Martin, B. V., and Morgan, J. T., "Synchronizing Signals for Maximal Bandwidth," Record 118, Highway Research Board, pp. 21-46, 1966.
29. Olson, P. L., and Rothery, R. W., "Driver Response to the Amber Phase of Traffic Signals," Traffic Engineering, Vol. 32, No. 5, pp. 17-21, 29, 1962.
30. Ostle, Bernard, Statistics in Research, Second Edition, The Iowa State University Press, Ames, Iowa, 1963.
31. "Progression," Unpublished Handout from CE 565, Traffic Engineering Operations and Controls, Purdue University, 1971.
32. Schwartz, Jesse G., An Urban Street Network Computer Simulation Model, Massachusetts Institute of Technology, Department of Civil Engineering Report R66-52, September, 1966.
33. Stark, M. C., "Computer Simulation of Traffic on Nine Blocks of a City Street," Theory of Traffic Flow, Bulletin 356, Highway Research Board, 1962.

34. Strickland, R. I., "Traffic Operation at Vehicle Tunnels," Proceedings, Highway Research Board, Vol. 33, p. 398, 1956.
35. Transportation Engineering Center, "Study of Electronic Devices as Traffic Aids," Report No. 202-1, Ohio State University, 1962.
36. Wagner, F. A., Barnes, F. C., and Gerlough, D. L., "Improved Criteria for Traffic Signal Systems in Urban Networks," Report No. 124, National Cooperative Highway Research Program, 1971.
37. Wagner, F. A., Barnes, F. C., Stirling, D. P., and Gerlough, D. L., "Urban Arterial and Network Simulation," Planning Research Corporation Report R-926, prepared for U. S. Dept. of Commerce, Bureau of Public Roads, Office of Research and Development (PB 174629).
38. Webster, F. V., "Traffic Signal Settings," Road Research Technical Paper No. 39, London, Her Majesty's Stationery Office, 1958.
39. Wilson, E. E., "Deceleration Distances for High Speed Vehicles," Proceedings, Highway Research Board, Vol. 20, pp. 393-398, 1940.
40. Wong, S. Y., "Traffic Simulation with a Digital Computer," Proceedings of the Western Joint Computer Conference, San Francisco, California, pp. 92-94, February 7-9, 1956.
41. Yardeni, L. A., "Algorithms for Traffic Signal Control," IBM Systems Journal, Vol. 4, No. 2, 1965.

APPENDICES

Appendix A

PRESIG Program Details

The following is a complete description of the PRESIG (preliminary geometry determination) program inputs. All data are directly related to the geometry of the network, and are easily measured in the field. The function of PRESIG is to convert these data to data needed by the simulation program SIGNET, using the procedure described in Chapter III. A summary of the required variables and formats is given in Table A-2.

The physical characteristics required for each link are as follows:

$ITYPE_i$ = intersection type, numerical class.

$JTYPE_i$ = intersection type, alphabetical subclass.

The intersection types recognized by PRESIG are shown in Figure A-1 with their class (number) and subclass (letter) identification. Care must be taken that the intersection is considered from the same orientation as it would be by an approaching driver on the link. For the purposes of PRESIG, one-way streets toward an intersection are not considered by the other links of that intersection. Therefore when the intersection type is chosen, one-way legs toward the intersection are not included.

FIGURE A-1
INTERSECTION TYPES

FIGURE A-1, cont.

N_i = number of lanes in link i . $N_{mzx} = 3$.

$LLAN_i$ = length of each lane, in feet, from center of tail intersection to center of head intersection. Left and right turning pockets are measured from the center of the head intersection back to the beginning of the lane.

$CURBD_i$ = distance from center of head intersection back to stop line.

$IPARKR_i$ = parking on right side of link, yes or no.

$IPARKL_i$ = parking on left side of link, yes or no.

$IOPP_{i,j,j=1,5}$ = opposing links beginning with rightmost and proceeding counter-clockwise. The directly opposite link (i.e. opposing left turns) is indexed by $j = 3$. Opposing links on the right are indexed by $j = 1$ and $j = 2$; if only one exists, it is indexed by $j = 1$. Opposing links on the left are indexed by $j = 4$ and $j = 5$; if only one exists, it is indexed by $j = 5$.

$NREC_{i,j,j=1,6}$ = receiving link for each of six possible turning movements, beginning with the rightmost and proceeding counter-clockwise. Straight-through movements in lane 1 and lane 2 are indexed by $j = 3$ and $j = 4$ respectively. One right turn is indexed by $j = 1$; two are indexed by $j = 1$ and $j = 2$. One left turn is indexed by $j = 6$; two are indexed by $j = 5$ and $j = 6$.

Turning radii are not required for all links, since a table of default radii is kept in PRESIG. These default values were developed from criteria found in the AASHO

Policy on Geometric Design of Rural Highways (1), and are shown in Table A-1. For those links with radii differing from the default values it is necessary to supply the proper values with the rest of the input data.

Revised discharge boundary values may be input prior to the final output similarly to the turning radii values. In both cases an input value of zero is assumed to mean no change is required. In the case of turning radii, the default value is used. In the case of discharge boundaries, the calculated value is used.

The PRESIG input deck consists of the cards given in Table A-2 in the specified order.

The PRESIG program code is given in Table A-3.

TABLE A-1
Default Radii Used in PRESIG

<u>Angle of Turn¹</u>	<u>Turn Type</u>	<u>Default Radius</u>
135°	Right	27 ft.
90°	Right	30 ft.
45°	Right	70 ft.
45°	Free Left	70 ft.
45°	Delayed Left	50 ft.
90°	Free Left	40 ft.
90°	Delayed Left	30 ft.
135°	Free Left	27 ft.
135°	Delayed Left	27 ft.

¹Angle of turn is measured from the extension of the centerline tangent of the approach link to the corresponding tangent on the receiving link.

TABLE A-2
PRESIG Punched Card Input Format

<u>Card Designation</u>	<u>Card Description</u>	<u>Column</u>	<u>Symbol</u>	<u>Meaning</u>	<u>Units</u>	<u>Format</u>
A	Program Parameters	1	A	Card Type	---	A
		6-7	NOLINK	No. of links in network	---	XX
		8-12	WIDTH	Lane width	ft.	XXXX.
		13-17	PARKW	Parking lane width	ft.	XXXX.
		18-21	NOUTPT	Number of output links	---	XXXX
B	Link Variables ^a	1	B	Card Type	---	B
		3-4	L	Link number	---	XX
		6	ITYPE	Intersection type, numerical	---	X
		7	JTYPE	Intersection type, alphabetical	---	X
		8	N	Number of lanes	---	X
		9-12	LLAN(1)	Lane length, lane 1	ft.	XXXX
		13-16	LLAN(2)	Lane length, lane 2	ft.	XXXX
		17-20	LLAN(3)	Lane length, lane 3	ft.	XXXX
		21-23	CURBD	Distance from center of head intersection back to stop line.	ft.	XXX

TABLE A-2, Cont.

<u>Card Designation</u>	<u>Card Description</u>	<u>Column</u>	<u>Symbol</u>	<u>Meaning</u>	<u>Units</u>	<u>Format</u>
		24-26	IPARKR	Parking on right (yes or no)	---	XXX ^b
		27-29	IPARKL	Parking on left (yes or no)	---	XXX ^b
		30-31	IOPP(1)	First opposing link number	---	XX
		.	.	.		
		.	.	.		
		38-39	IOPP(5)	Fifth opposing link number	---	XX
		40-41	LNKADJ	Adjacent link	---	XX
		42-43	NREC(1)	Receiving link, turn #1	---	XX
		.	.	.		
		.	.	.		
		52-53	NREC(6)	Receiving link, turn #6	---	XX
C	Radius Card ^e	1	C	Card type	---	C
		4-5	LHOLD	Link number	---	XX
		7-10	RAD(1)	Preset radius, turn number 1	ft.	XXX. ^c
		.	.	.		.
		.	.	.		.
		27-30	RAD(6)	Preset radius, turn number 6	ft.	XXX. ^c

TABLE A-2, Contd.

<u>Card Designation</u>	<u>Card Description</u>	<u>Column</u>	<u>Symbol</u>	<u>Meaning</u>	<u>Units</u>	<u>Format</u>
D	Revised Discharge Boundary Card ^e	1	D	Card type	---	D
		4-5	L	Link number	---	XX
		6-9	DNEW(1)	Revised Discharge Boundary for turn number 1	ft.	XXXX ^d
		.	.	.		.
		.	.	.		.
E	Revised Begin-Turn Point Card ^e	.	.	.		.
		26-29	DNEW(6)	Revised Discharge Boundary for turn number 6	ft.	XXX ^d
		1	F	Card Type	---	F
		4-5	L	Link number	---	XX
		6-9	BGNNEW(1)	Revised Begin-Turn point for turn no. 1	---	XXXX ^d
		.	.	.		.
		.	.	.		.
		.	.	.		.
		26-29	BGNNEW(6)	Revised Begin-Turn point for turn no. 6	---	XXXX ^d
		.	.	.		.

TABLE A-2, Cont.

^aOne card for each link.

^bWord must be left-justified in the field.

^cBlank field does not cause radius to be preset.

^dBlank field causes no revision to that turn.

^eOne card required for each link being revised. Series must be concluded with an END (Col. 1-3) card. END card must be present even if no revisions are made.

TABLE A-3, cont

```

      READ (5,1) TYPE,NOLINK,WIDTH,PARKW,NOUTPT
1  FORMAT (A1,4X,I2,F5.0,F5.0,I4)

C
C  PROGRAM PARAMETERS CARD
C
C  TYPE= CARD TYPE (A)
C  NOLINK= NO. OF LINKS IN NETWORK
C  WIDTH= NETWORK-WIDE LANE WIDTH
C  PARKW=NETWORK-WIDE PARKING LANE WIDTH
C  NOUTPT= NUMBER OF OUTPUT LINKS
C
      IF (TYPE.NE.1HA) CALL DIAGN(1,KNTCARD)
      WRITE (6,2) NOLINK,WIDTH,PARKW
2  FORMAT (1H1,4X,125(1H*),/5X,125(1H*),///51X,*SYSTEM GEOMETRY DEVER
      1MTNATION*,///5X,125(1H*),/5X,125(1H*),///20X,*NUMBER OF LINKS =
      2*,I4,///25X,*LANE WIDTH = *,F4.0,* FEET*,//17X,*PARKING LANE WIDTH
      3= *,F4.0,* FEET*,///20X,*INT. TYPE*,5X,*LANES*,5X,*LANE LENGTH
      4(1-3)*,9X,*OPPOSING LINKS (1-5)*,14X,*PARKING*,5X,*ADJ. LINK*///)
      ISTART=NOUTPT+1
      DO 120 LINK=1,NOLINK

C
      READ (5,3) TYPE,L,ITYPE(LINK),JTYPE(LINK),N,(LLAN(LINK,I),I=1,3),
      1CURPC(LINK),IPARKP,IPARKL,(IOPP(LINK,I),I=1,5),LNKADJ(LINK),(NREC(
      2LINK,I),I=1,6)
3  FORMAT (A1,1X,I2,1X,I1,A1,I1,3F4.0,F3.0,2A3,13I2)

C
C  LINK VARIABLES CARDS
C
C  TYPE= CARD TYPE (B)
C  L= LINK NUMBER
C  ITYPE= INTERSECTION TYPE, NUMERICAL
C  JTYPE= INTERSECTION TYPE, ALPHABETICAL
C  N= NUMBER OF LANES IN LINK
C  LLAN= LENGTH OF EACH LANE, CENTER OF INTERSECTION TO CENTER OF
C  INTERSECTION. LEFT AND RIGHT TURN POCKETS ARE MEASURED FROM
C  CENTER OF INTERSECTION BACK TO BEGINNING OF LANE.
C  CURPC= DISTANCE FROM CENTER OF HEAD INTERSECTION BACK TO STOP LINE
C  IPARK= PARKING (YES OR NO, LEFT JUSTIFIED IN FIELD)
C  IOPP= OPPOSING LINKS, BEGINNING WITH RIGHTMOST AND PROCEEDING
C  COUNTERCLOCKWISE.
C  LNKADJ= ADJACENT LINK (PARALLEL,BUT OPPOSITE DIRECTION)
C  NREC= RECEIVING LINK FOR EACH OF 6 TURNS, BEGINNING WITH RIGHTMOST
C  AND PROCEEDING COUNTERCLOCKWISE. STRAIGHT MOVEMENTS, LANE 1
C  AND LANE 2, ARE ALWAYS NUMBERED 3 AND 4, RESPECTIVELY. ONE
C  RIGHT TURN IS NUMBERED 1. TWO RIGHT TURNS ARE NUMBERED 1 AND
C  2. ONE LEFT TURN IS NUMBERED 6. TWO LEFT TURNS ARE NUMBERED
C  5 AND 6.
C
      KNTCARD=KNTCARD+1
      IF (TYPE.NE.1HB) CALL DIAGN(2,KNTCARD)
      IF (L.NE.LINK) CALL DIAGN(3,KNTCARD)
      IF (N.GT.3) CALL DIAGN(4,KNTCARD)
      IF (N.GT.3) N=3
      IF (N.LT.1) CALL DIAGN(5,KNTCARD)
      IF (N.LT.1) N=1
      IF (IPARKR.EQ.NMPRK(2)) GO TO 100
      JPARK(LINK)=1
      GO TO 110
100 JPARK(LINK)=0
110 JP=JPARK(LINK)+1
      FLPAWK(LINK)=FLOAT(JPARK(LINK))

```

TABLE A-3, cont.

```

FLPARKL(LINK)=0.
IF (IPARKL.EQ.NMPRK(1)) FLPARKL(LINK)=1.
NCLANE(LINK)=N
FLNOLN(LINK)=FLOAT(NCLANE(LINK))
WRITE (6,4) L,JTYPE(L),JTYPE(L),NCLANE(L),(LLAN(L,I),I=1,3),(IOPP(
1 L,I),I=1,5),NMPRK(JP),LNKADJ(L)
4 FORMAT (10X,'LINK',I3,7X,I1,A1,10X,I2,6X,3(F4.0,3X),5X,5(I2,3X),11
1X,A3,10X,I2)
IF (JTYPE(LINK).EQ.1HA) JTYPE(LINK)=1
IF (JTYPE(LINK).EQ.1HB) JTYPE(LINK)=2
IF (JTYPE(LINK).EQ.1HC) JTYPE(LINK)=3
IF (JTYPE(LINK).EQ.1HD) JTYPE(LINK)=4
IF (JTYPE(LINK).EQ.1HE) JTYPE(LINK)=5
IF (JTYPE(LINK).EQ.1HF) JTYPE(LINK)=6
IF (JTYPE(LINK).EQ.1HG) JTYPE(LINK)=7
IF (JTYPE(LINK).EQ.1HH) JTYPE(LINK)=8
120 CONTINUE
C
125 READ (5,5) TYPE,LHOLD,(RAD(LHOLD,I),I=1,6)
5 FORMAT (A1,2X,I2,1X,6(F4.1))
C
C      RADIUS CARD
C
C      TYPE= CARD TYPE (C)
C      LHOLD= LINK ON WHICH PRESET RADIUS IS TO BE USED
C      RAD= RADIUS OF EACH OF SIX POSSIBLE TURNS. RAD=0. CAUSES NO PRESET.
C
C      SERIES MUST BE CONCLUDED WITH AN END (COL. 1-3) CARD
C
IF (END,5) 126,127
126 KNTCARD=KNTCARD+1
CALL DIAGN(5,KNTCARD)
GO TO 135
127 KNTCARD=KNTCARD+1
IF (TYPE.EQ.1HE) GO TO 135
130 IF (TYPE.NE.1HC) CALL DIAGN(6,KNTCARD)
GO TO 135
135 DO 140 LINK=ISTART,NOLINK
L=LNKADJ(LINK)
IF (L.EQ.0) GO TO 130
NOL=NCLANE(L)
FN(L)=FLNOLN(L)
IF (FN(L).EQ.1) GO TO 130
IF (LLAN(L,NOL).LT.LLAN(L,NOL-1).OR.LLAN(L,1).LT.LLAN(L,2)) FN(L)=
1 FN(L)-1
138 WADJ=WIDTH*FN(L)+PARKW*FLPARK(L)
GO TO 140
139 WADJ=FLPARKL(LINK)*PARKW
140 W(LINK)=WADJ+WIDTH*FLNOLN(LINK)+PARKW*FLPARK(LINK)
DO 1000 LINK=ISTART,NOLINK
IT=ITYPE(LINK)
JT=JTYPE(LINK)
L=LNKADJ(LINK)
IF (L.EQ.0) GO TO 141
WADJ=WIDTH*FN(L)+PARKW*FLPARK(L)
GO TO 142
141 WADJ=FLPARKL(LINK)*PARKW
142 CONTINUE
GO TO (150,190,250,320,360,400,450,520) IT
150 DO 155 I=1,6
155 IF (RAD(LINK,I).EQ.0.) RAD(LINK,I)=R(I)

```

TABLE A-3, cont.

```

GO TC (170,170,180,170,170) JT
177 N=NREC(LINK,6)
    D90(N,6)=DBOLT90(LINK,5,N)
    D90(N,7)=DBOLT90(LINK,6,N)
    IF (JTYPE(LINK).EQ.2) GO TO 1000
180 N=NREC(LINK,1)
    D90(N,2)=DBORT90(LINK,1,N)
    GO TO 1000
190 IF (JTYPE(LINK).EQ.2.OR.JTYPE(LINK).EQ.5.AND.RAD(LINK,1).EQ.0.)
    1 RAD(LINK,1)=R(2)
    DO 195 I=1,6
195 IF (RAD(LINK,I).EQ.0.) RAD(LINK,I)=R(I)
    GO TC (200,200,220,200,229,240) JT
200 N=NREC(LINK,6)
    D90(N,6)=DBOLT90(LINK,5,N)
    D90(N,7)=DBOLT90(LINK,6,N)
    IF (JTYPE(LINK).EQ.2) GO TO 229
220 N=NREC(LINK,1)
    D90(N,2)=DBORT90(LINK,1,N)
229 N=NREC(LINK,1)
    NT=1
    GO TO 232
230 N=NREC(LINK,2)
    NT=2
232 D90(N,3)=DBORT2(LINK,NT,N)
    GO TO 1000
240 N=NREC(LINK,1)
    IF (RAD(LINK,1).EQ.R(1)) RAD(LINK,1)=27.
    IF (RAD(LINK,2).EQ.R(2)) RAD(LINK,2)=R(1)
    D90(N,1)=DBORT1(LINK,1,N)
    GO TC 230
250 IF (JTYPE(LINK).NE.3.AND.JTYPE(LINK).NE.5) GO TO 251
    IF (RAD(LINK,5).EQ.0.) RAD(LINK,5)=R(3)
    IF (RAD(LINK,6).EQ.0.) RAD(LINK,6)=R(4)
    GO TC 254
251 IF (JTYPE(LINK).NE.6) GO TO 254
    IF (RAD(LINK,5).EQ.0.) RAD(LINK,5)=25.
    IF (RAD(LINK,6).EQ.0.) RAD(LINK,6)=25.
254 DO 255 I=1,6
255 IF (RAD(LINK,I).EQ.0.) RAD(LINK,I)=R(I)
    GO TC (260,260,280,260,290,310) JT
260 N=NREC(LINK,6)
    D90(N,6)=DBOLT90(LINK,5,N)
    D90(N,7)=DBOLT90(LINK,6,N)
    IF (JTYPE(LINK).EQ.2) GO TO 290
280 N=NREC(LINK,1)
    D90(N,2)=DBORT90(LINK,1,N)
290 N=NREC(LINK,5)
    NT=3
    IF (JT.EQ.3.OR.JT.EQ.5) N=NREC(LINK,6)
    IF (JT.EQ.3.OR.JT.EQ.5) NT=5
    D90(N,4)=DBOLT1(LINK,NT,N)
    NT=NT+1
    D90(N,5)=DBOLT1(LINK,NT,N)
    GO TC 1000
310 N=NREC(LINK,6)
    D90(N,8)=DBOLT2(LINK,5,N)
    D90(N,9)=DBOLT2(LINK,6,N)
    GO TO 290
320 IF (RAD(LINK,1).EQ.0.) RAD(LINK,1)=27.
    IF (RAD(LINK,2).EQ.0.) RAD(LINK,2)=R(1)

```

TABLE A-3, cont.

```

DO 325 I=3,6
125 IF (RAD(LINK,I).EQ.0.) RAD(LINK,I)=R(I)
GO TO (110,130,340,330,350) JT
330 N=NRFC(LINK,6)
D90(N,6)=D9OLT90(LINK,5,N)
D90(N,7)=D9OLT90(LINK,6,N)
IF (JTYPE(LINK).EQ.2) GO TO 350
340 N=NRFC(LINK,2)
D90(N,2)=D9OLT90(LINK,2,N)
350 N=NRFC(LINK,1)
D90(N,1)=D9OLT1(LINK,1,N)
GO TO 1000
360 DO 365 I=1,2
165 IF (RAD(LINK,I).EQ.0.) RAD(LINK,I)=R(I)
IF (RAD(LINK,3).EQ.0.) RAD(LINK,3)=R(5)
IF (RAD(LINK,4).EQ.0.) RAD(LINK,4)=R(6)
IF (RAD(LINK,5).EQ.0.) RAD(LINK,5)=25.
IF (RAD(LINK,6).EQ.0.) RAD(LINK,6)=25.
GO TO (170,370,380,370,390) JT
170 N=NRFC(LINK,5)
D90(N,6)=D9OLT90(LINK,3,N)
D90(N,7)=D9OLT90(LINK,4,N)
IF (JTYPE(LINK).EQ.2) GO TO 390
380 N=NRFC(LINK,1)
D90(N,2)=D9OLT90(LINK,1,N)
390 N=NRFC(LINK,6)
D90(N,8)=D9OLT2(LINK,5,N)
D90(N,9)=D9OLT2(LINK,6,N)
GO TO 1000
400 IF (JTYPE(LINK).EQ.3.OR.JTYPE(LINK).EQ.5.AND.RAD(LINK,5).EQ.0.) RA
10(LINK,5)=R(3)
IF (JTYPE(LINK).EQ.3.OR.JTYPE(LINK).EQ.5.AND.RAD(LINK,6).EQ.0.)
1RAD(LINK,6)=R(4)
IF (RAD(LINK,1).EQ.0.) RAD(LINK,1)=27.
IF (RAD(LINK,2).EQ.0.) RAD(LINK,2)=R(1)
DO 405 I=3,6
405 IF (RAD(LINK,I).EQ.0.) RAD(LINK,I)=R(I)
GO TO (410,410,420,410,430) JT
410 N=NRFC(LINK,6)
D90(N,6)=D9OLT90(LINK,5,N)
D90(N,7)=D9OLT90(LINK,6,N)
IF (JTYPE(LINK).EQ.2) GO TO 430
420 N=NRFC(LINK,2)
D90(N,2)=D9OLT90(LINK,2,N)
430 N=NRFC(LINK,1)
D90(N,1)=D9OLT1(LINK,1,N)
N=NRFC(LINK,5)
IF (JTYPE(LINK).EQ.3.OR.JTYPE(LINK).EQ.5) N=NRFC(LINK,6)
D90(N,4)=D9OLT1(LINK,3,N)
D90(N,5)=D9OLT1(LINK,4,N)
GO TO 1000
450 IF (JTYPE(LINK).NE.2.AND.JTYPE(LINK).NE.6) GO TO 454
IF (RAD(LINK,1).EQ.0.) RAD(LINK,1)=R(2)
454 DO 459 I=1,6
IF (I.GT.2) GO TO 455
IF (RAD(LINK,I).EQ.0.) RAD(LINK,I)=R(I)
GO TO 459
455 IF (I.GT.4) GO TO 456
IF (RAD(LINK,I).EQ.0.) RAD(LINK,I)=R(I+2)
GO TO 459
456 IF (RAD(LINK,5).EQ.0.) RAD(LINK,5)=25.

```

TABLE A 3, cont

```

      IF (RAD(LINK,6).EQ.0.) RAD(LINK,6)=75.
459 CONTINUE
      GO TO (460,460,470,460,510) JT
460 N=NRFC(LINK,5)
      DRG(N,6)=DROLT90(LINK,7,N)
      DRG(N,7)=DROLT90(LINK,4,N)
      IF (JTYPE(LINK).EQ.2) GO TO 510
470 N=NRFC(LINK,1)
      DRG(N,2)=DRORT90(LINK,1,N)
480 N=NRFC(LINK,2)
      DRG(N,3)=DRORT2(LINK,2,N)
      GO TO 500
500 N=NRFC(LINK,6)
      DRG(N,8)=DROLT2(LINK,5,N)
      DRG(N,9)=DROLT2(LINK,6,N)
      GO TO 1000
510 N=NRFC(LINK,1)
      DRG(N,3)=DRORT2(LINK,1,N)
      GO TO 500
520 IF (JTYPE(LINK).EQ.4.OR.JTYPE(LINK).EQ.6) GO TO 525
      IF (RAD(LINK,3).EQ.0.) RAD(LINK,3)=R(3)
      IF (RAD(LINK,4).EQ.0.) RAD(LINK,4)=R(4)
      IF (RAD(LINK,5).EQ.0.) RAD(LINK,5)=25.
      IF (RAD(LINK,6).EQ.0.) RAD(LINK,6)=75.
      GO TO 530
525 IF (RAD(LINK,5).EQ.0.) RAD(LINK,5)=R(7)
      IF (RAD(LINK,6).EQ.0.) RAD(LINK,6)=R(4)
530 IF (JTYPE(LINK).EQ.5.OR.JTYPE(LINK).EQ.7) GO TO 535
      IF (RAD(LINK,1).EQ.0.) RAD(LINK,1)=27.
      IF (RAD(LINK,2).EQ.0.) RAD(LINK,2)=R(2)
      GO TO 540
535 IF (RAD(LINK,1).EQ.0.) RAD(LINK,1)=R(2)
540 GO TO (550,560,550,551,560,551,560) JT
550 N=NRFC(LINK,5)
      NT=3
      GO TO 555
551 N=NRFC(LINK,6)
      NT=5
555 DRG(N,4)=DROLT1(LINK,NT,N)
      NT=NT+1
      DRG(N,5)=DROLT1(LINK,NT,N)
      IF (JTYPE(LINK).EQ.4.OR.JTYPE(LINK).EQ.6) GO TO 570
560 N=NRFC(LINK,6)
      DRG(N,8)=DROLT2(LINK,5,N)
      DRG(N,9)=DROLT2(LINK,6,N)
570 GO TO (580,580,580,580,590,580,590) JT
580 N=NRFC(LINK,1)
      DRG(N,1)=DRORT1(LINK,1,N)
      IF (JTYPE(LINK).EQ.3.OR.JTYPE(LINK).EQ.6) GO TO 1000
      N=NRFC(LINK,2)
      GO TO 600
590 N=NRFC(LINK,1)
600 DRG(N,3)=DRORT2(LINK,N,N)
1000 CONTINUE
      DO 1105 L=1,NOLINK
      CR(L)=0.
      DO 1100 IDR=1,9
      IF (DRO(L,IDR).GT.78(L)) CR(L)=DRO(L,IDR)
1100 CONTINUE
1105 CONTINUE
      DO 1210 L=1,START,NOLINK

```


TABLE A-3, cont.

```

NO=NOLANE(L)
GO TO (1110,1120,1150) NO
1110 XLINK(L,1)=LLAN(L,1)-CURPD(L)-DB(L)
      XLINK(L,2)=0.
      XLINK(L,3)=0.
      GO TO 1200
1120 IF (LLAN(L,1)-LLAN(L,2)) 1130,1140,1150
1130 XLINK(L,1)=LLAN(L,1)-CURPD(L)
      XLINK(L,2)=LLAN(L,2)-CURPD(L)-DB(L)
1135 XLINK(L,3)=0.
      GO TO 1200
1140 XLINK(L,1)=LLAN(L,1)-CURPD(L)-DB(L)
      XLINK(L,2)=LLAN(L,2)-CURPD(L)-DB(L)
      GO TO 1135
1150 XLINK(L,1)=LLAN(L,1)-CURPD(L)-DB(L)
      XLINK(L,2)=LLAN(L,2)-CURPD(L)
      GO TO 1135
1160 IF (LLAN(L,1).GE.LLAN(L,2).AND.LLAN(L,1).GE.LLAN(L,3)) GO TO 1170
      IF (LLAN(L,2).GE.LLAN(L,3).AND.LLAN(L,2).GE.LLAN(L,1)) GO TO 1175
      GO TO 1180
1170 XLINK(L,1)=LLAN(L,1)-CURPD(L)-DB(L)
      IF (LLAN(L,2).LT.LLAN(L,1)) GO TO 1173
      XLINK(L,2)=LLAN(L,2)-CURPD(L)-DB(L)
      IF (LLAN(L,3).EQ.LLAN(L,2)) GO TO 1172
1171 XLINK(L,3)=LLAN(L,3)-CURPD(L)
      GO TO 1200
1172 XLINK(L,3)=LLAN(L,3)-CURPD(L)-DB(L)
      GO TO 1200
1173 XLINK(L,2)=LLAN(L,2)-CURPD(L)
      IF (LLAN(L,3).EQ.LLAN(L,1)) GO TO 1172
      GO TO 1171
1175 XLINK(L,2)=LLAN(L,2)-CURPD(L)-DB(L)
      XLINK(L,1)=LLAN(L,1)-CURPD(L)
      IF (LLAN(L,3).EQ.LLAN(L,2)) GO TO 1172
      GO TO 1171
1180 XLINK(L,1)=LLAN(L,1)-CURPD(L)
      XLINK(L,2)=LLAN(L,2)-CURPD(L)
      GO TO 1172
1200 CONTINUE
      STPLINE(L)=AMAX1(XLINK(L,1),XLINK(L,2),XLINK(L,3))
1210 CONTINUE
      DO 1400 L=ISTART,NOLINK
      I=ITYPE(L)
      J=JTYPE(L)
      DO 1400 NT=1,6
      KGO=KSW(NT,J,I)
      IF (KGO.EQ.0) GO TO 1400
      GO TO (1300,1310,1320,1330,1330,1340,1350,1360) KGO
1300 CALL PTOIAG1 (L,NT)
      GO TO 1400
1310 CALL RT90 (L,NT)
      GO TO 1400
1320 CALL RTOIAG2 (L,NT)
      GO TO 1400
1330 CALL STRAITF (L,NT)
      GO TO 1400
1340 CALL LTOTAG1 (L,NT)
      GO TO 1400
1350 CALL LT90 (L,NT)
      GO TO 1400
1360 CALL LTOTAG2 (L,NT)

```


TABLE A 3, cont

```

1400 CONTINUE
    DO 1430 L=1,NOUTPT
        NOL=NOLANE
    DO 1410 LN=1,NOL
1410 XLINK(L,LN)=400.
        STPLINE(L)=400.
    DO 1420 IT=1,6
1420 DBNDRY(L,IT)=400.
1430 CONTINUE
    DO 1439 I=1,NOLINK
    DO 1439 J=1,6
1439 DBND(I,J)=0.
C
1440 READ(5,6) TYPE,L,(DBNEW(L,I),I=1,6)
    6 FORMAT (A1,2X,I2,6F4.0)
C
C     REVISED DISCHARGE BOUNDARY CARD
C
C     TYPE= CARD TYPE (D)
C     L= LINK ON WHICH REVISED DISCHARGE BOUNDARY IS TO BE USED
C     DBNEW=REVISED DISCHARGE BOUNDARY FOR EACH OF SIX POSSIBLE
C     TURNING MOVEMENTS. DBNEW=0 CAUSES NO REVISION.
C
C
    IF (EOF,5) 1441,1442
1441 KNTCARD=KNTCARD+1
    CALL DIAGN(7,KNTCARD)
    GO TO 1449
1442 KNTCARD=KNTCARD+1
    IF (TYPE.EQ.1HE) GO TO 1449
    IF (TYPE.NE.1HD) CALL DIAGN(8,KNTCARD)
    GO TO 1440
1449 CONTINUE
    DO 1451 I=1,NOLINK
    DO 1451 J=1,6
1451 IF (DBNEW(I,J).NE.0.) DBNDRY(I,J)=DBNEW(I,J)
    DO 1454 I=1,NOLINK
    DO 1454 J=1,6
1454 BGNNEW(I,J)=0.
C
1455 READ(5,15) TYPE,L,(BGNNEW(L,I),I=1,6)
    15 FORMAT (A1,2X,I2,6F4.0)
C
C     REVISED BEGIN-TURN POINT CARD
C
C     TYPE= CARD TYPE (F)
C     L= LINK ON WHICH REVISED BEGIN-TURN POINT IS TO BE USED
C     BGNNEW= REVISED BEGIN-TURN POINT FOR EACH OF SIX POSSIBLE TURN
C     TYPES. BGNNEW=0 CAUSES NO REVISION.
C
C
    IF (EOF,5) 1456,1457
1456 KNTCARD=KNTCARD+1
    CALL DIAGN (7,KNTCARD)
    GO TO 1459
1457 KNTCARD=KNTCARD+1
    IF (TYPE.EQ.1HE) GO TO 1458
    IF (TYPE.NE.1HF) CALL DIAGN (8,KNTCARD)
    GO TO 1455
1458 DO 1459 I=1,NOLINK
    DO 1459 J=1,6

```

TABLE A-3, cont.

```

1459 IF (PGNNEW(I,J).NE.0.) BEGINPT(I,J)=PGNNEW(I,J)
      WRITE (6,10)
10  FORMAT (1H1,//////20X,*SUMMARY OF GEOMETRY INPUTS TO SIGNET*,//////2
17X,*STOP*,14X,*LANE LENGTH*,24X,*DISCHARGE BOUNDARY*,/27X,*LINE*,
212X,*1*,6X,*2*,6X,*3*,11X,*1*,6X,*2*,6X,*3*,6X,*4*,6X,*5*,6X,*6*,6
3X//)
      DO 1450 L=1,NOLINK
1450 WRITE (6,11) L,STPLINE(L),(XLINK(L,I),I=1,3),(DBNDRY(L,I),I=1,6)
11  FORMAT (10X,*LINK*,I3,10X,F4.0,10X,3(F4.0,3X),7X,6(F4.0,3X))
      WRITE (6,12)
12  FORMAT (//////35X,*BEGINNING POINT OF TURN*,31X,*RADIUS OF TURN*
1,/29X,*1*,6X,*2*,6X,*3*,6X,*4*,6X,*5*,6X,*6*,6X,7X,*1*,6X,*2*,6X,*
23*,6X,*4*,6X,*5*,6X,*6*//)
      DO 1460 L=ISTART,NOLINK
1460 WRITE (6,13) L,(BEGINPT(L,I),I=1,6),(RAD(L,I),I=1,6)
13  FORMAT (10X,*LINK*,I3,10X,6(F4.0,3X),7X,6(F4.0,3X))
      IF (IPNCH.EQ.0) STOP
      DO 1470 L=1,NOLINK
1470 WRITE (7,14) L,STPLINE(L),(DBNDRY(L,J),J=1,6),(XLINK(L,J),J=1,3),
1(BEGINPT(L,J),J=1,6),(RAD(L,I),I=1,6)
14  FORMAT (I2,3X,16F4.0,/5X,6F3.0)
      STOP
      END

```

```

      SUBROUTINE RTDIAG1 (L,NT)
      COMMON/A,DB(99),CURBD(99),RAD(99,6),DBO(99,9),BEGINPT(99,6),
1  STPLINE(99),DBNDRY(99,6),NREC(99,6)
      N=NREC(L,NT)
      IF (DBO(N,1).LT.DB(N)) GO TO 100
      BEGINPT(L,NT)=STPLINE(L)+CURBD(L)-DB(N)
      DBNDRY(L,NT)=BEGINPT(L,NT)+.75*3.14159*RAD(L,NT)
      RETURN
100  DX=DB(N)-DBO(N,1)
      BEGINPT(L,NT)=STPLINE(L)+CURBD(L)-DBO(N,1)
      DBNDRY(L,NT)=BEGINPT(L,NT)+.75*3.14159*RAD(L,NT)+DX
      RETURN
      END

```

```

      SUBROUTINE RT90 (L,NT)
      COMMON/A,DB(99),CURBD(99),RAD(99,6),DBO(99,9),BEGINPT(99,6),
1  STPLINE(99),DBNDRY(99,6),NREC(99,6)
      COMMON/B/FLPARK(99),PARKW,WIDTH,W(99)
      N=NREC(L,NT)
      DX=DB(N)-DBO(N,2)
      BEGINPT(L,NT)=STPLINE(L)+CURBD(L)-W(N)/2.+FLPARK(N)*PARKW+WIDTH/2.
1-RAD(L,NT)
      DBNDRY(L,NT)=BEGINPT(L,NT)+.5*3.14159*RAD(L,NT)+DX
      RETURN
      END

```

```

      SUBROUTINE RTDIAG2 (L,NT)
      COMMON/A,DB(99),CURBD(99),RAD(99,6),DBO(99,9),BEGINPT(99,6),
1  STPLINE(99),DBNDRY(99,6),NREC(99,6)
      COMMON/B/FLPARK(99),PARKW,WIDTH,W(99)
      N=NREC(L,NT)

```

TABLE A 3, cont

```

DX=DB(N)-DBO(N,3)
BEGINPT(L,NT)=STPLINE(L)+CURBD(L)-RAD(L,NT)*TAN(3.14159/8.)-
1 (W(N)/2.-WIDTH/2.-FLPARK(N)*PARKW)*TAN(3.14159/8.)
DBNDRY(L,NT)=BEGINPT(L,NT)+.25*3.14159*RAD(L,NT)+DX
RETURN
END

```

```

SUBROUTINE STRAITE (L,NT)
COMMON/A/DB(99),CURBD(99),RAD(99,6),DBO(99,3),BEGINPT(99,6),
1 STPLINE(99),DBNDRY(99,6),NREC(99,6)
N=NREC(L,NT)
DBNDRY(L,NT)=STPLINE(L)+CURBD(L)+DB(N)
RETURN
END

```

```

SUBROUTINE LTOIAG1 (L,NT)
COMMON/A/DB(99),CURBD(99),RAD(99,6),DBO(99,9),BEGINPT(99,6),
1 STPLINE(99),DBNDRY(99,6),NREC(99,6)
COMMON/B/FLPARK(99),PARKW,WIDTH,W(99)
N=NRFC(L,NT)
NM=NT
IF (NT.EQ.5) NT=3
IF (NT.EQ.6) NT=5
DXD=DB(N)-DBO(N,4)
DXF=CB(N)-DBO(N,5)
BEGINPT(L,NT)=STPLINE(L)+CURBD(L)-RAD(L,NT)*TAN(3.14159/8.)+
1 1.414*(W(N)/2.-(FLPARK(N)*PARKW+WIDTH/2.))
BEGINPT(L,NT+1)=STPLINE(L)+CURBD(L)-RAD(L,NT+1)*TAN(3.14159/8.)+
1 1.414*(W(N)/2.-(FLPARK(N)*PARKW+WIDTH/2.))
DBNDEL=BEGINPT(L,NT)+.25*3.14159*RAD(L,NT)+DXD
DBNFREE=BEGINPT(L,NT+1)+.25*3.14159*RAD(L,NT+1)+DXF
DBNDRY(L,NM)=AMAX1(DBNDEL,DBNFREE)
NT=NM
RETURN
END

```

```

SUBROUTINE LT90 (L,NT)
COMMON/A/DB(99),CURBD(99),RAD(99,6),DBO(99,9),BEGINPT(99,6),
1 STPLINE(99),DBNDRY(99,6),NREC(99,6)
COMMON/B/FLPARK(99),PARKW,WIDTH,W(99)
N=NREC(L,NT)
NM=NT
IF (NT.EQ.5) NT=3
IF (NT.EQ.6) NT=5
DXD=DB(N)-DBO(N,6)
DXF=CB(N)-DBO(N,7)
BEGINPT(L,NT)=STPLINE(L)+CURBD(L)-RAD(L,NT)+(W(N)/2.-(FLPARK(N)
1 *PARKW+WIDTH/2.))
BEGINPT(L,NT+1)=STPLINE(L)+CURBD(L)-RAD(L,NT+1)+(W(N)/2.-(FLPARK(N)
1 *PARKW+WIDTH/2.))
DBNDEL=BEGINPT(L,NT)+.5*3.14159*RAD(L,NT)+DXD
DBNFREE=BEGINPT(L,NT+1)+.5*3.14159*RAD(L,NT+1)+DXF
DBNDRY(L,NM)=AMAX1(DBNDEL,DBNFREE)
NT=NM
RETURN
END

```

TABLE A-3, cont.

```

SUBROUTINE LTDIAG2 (L,NT)
COMMON/A/DB(99),CURBD(99),RAD(99,6),DBO(99,9),BEGINPT(99,6),
1  STPLINE(99),DBNDRY(99,6),NREC(99,6)
COMMON/B/FLPARK(99),PARKW,WIDTH,W(99)
N=NREC(L,NT)
NM=NT
IF (NT.EQ.5) NT=3
IF (NT.EQ.6) NT=5
DXD=CB(N)-DBO(N,8)
DXF=CB(N)-DBO(N,9)
BEGINPT(L,NT)=STPLINE(L)+CURBD(L)-DBO(N,8)+1.414*(W(N)/2.-
1 (FLPARK(N)*PARKW+WIDTH/2.))
BEGINPT(L,NT+1)=STPLINE(L)+CURBD(L)-DBO(N,9)+1.414*(W(N)/2.-
1 (FLPARK(N)*PARKW+WIDTH/2.))
DBNDEL=BEGINPT(L,NT)+.75*3.14159*RAD(L,NT)+DXD
DBNFREE=BEGINPT(L,NT+1)+.75*3.14159*RAD(L,NT+1)+DXF
DBNDRY(L,NM)=AMAX1(DBNDEL,DBNFREE)
NT=NM
RETURN
END

```

```

REAL FUNCTION DBORT1(L,NT,N)
COMMON/A/DB(99),CURBD(99),RAD(99,6),DBO(99,9),BEGINPT(99,6),
1  STPLINE(99),DBNDRY(99,6),NREC(99,6)
COMMON/B/FLPARK(99),PARKW,WIDTH,W(99)
DBORT1=RAD(L,NT)/TAN(3.14159/8.)+(W(N)/2.-WIDTH/2.-PARKW*FLPARK(N)
1 +1.414*(W(L)/2.-WIDTH/2.-PARKW*FLPARK(L))
RETURN
END

```

```

REAL FUNCTION DBORT90(L,NT,N)
COMMON/A/DB(99),CURBD(99),RAD(99,6),DBO(99,9),BEGINPT(99,6),
1  STPLINE(99),DBNDRY(99,6),NREC(99,6)
COMMON/B/FLPARK(99),PARKW,WIDTH,W(99)
DBORT90=RAD(L,NT)+W(L)/2.-WIDTH/2.-FLPARK(L)*PARKW
RETURN
END

```

```

REAL FUNCTION DBORT2(L,NT,N)
COMMON/A/DB(99),CURBD(99),RAD(99,6),DBO(99,9),BEGINPT(99,6),
1  STPLINE(99),DBNDRY(99,6),NREC(99,6)
COMMON/B/FLPARK(99),PARKW,WIDTH,W(99)
DBORT2=(RAD(L,NT)+W(N)/2.-WIDTH/2.-FLPARK(N)*PARKW)*TAN(3.14159/8.
1)
RETURN
END

```

```

REAL FUNCTION DBOLT1(L,NT,N)
COMMON/A/DB(99),CURBD(99),RAD(99,6),DBO(99,9),BEGINPT(99,6),
1  STPLINE(99),DBNDRY(99,6),NREC(99,6)

```

TABLE A-3, cont.

```

COMMON/9/FLPARK(99), PARKW, WIDTH, W(99)
DBOLT1= (RAD(L, NT) - W(N)/2. + WIDTH/2. + PARKW*FLPARK(N)) * TAN(3.14189/8.
1)
RETURN
END

```

```

REAL FUNCTION DBOLT90(L, NT, N)
COMMON/4/DB(99), CURBD(99), RAD(99,6), DB(99,9), BEGINPT(99,6),
1 STPLINE(99), D3NDRY(99,6), NREC(99,6)
COMMON/9/FLPARK(99), PARKW, WIDTH, W(99)
DBOLT90= RAD(L, NT) - WIDTH/2.
RETURN
END

```

```

REAL FUNCTION DBOLT2(L, NT, N)
COMMON/4/DB(99), CURBD(99), RAD(99,6), DB(99,9), BEGINPT(99,6),
1 STPLINE(99), D3NDRY(99,6), NREC(99,6)
COMMON/9/FLPARK(99), PARKW, WIDTH, W(99)
COMMON/0/FLNOLN(99)
DBOLT2= ((RAD(L, NT) - W(N) + WIDTH + FLPARK(N) * PARKW) * 1.414 + RAD(L, NT) + WID
1 TH * (FLNOLN(L) - .5) - W(L)/2. + PARKW*FLPARK(L)) * 1.414 - RAD(L) + W(N) - WIDTH
2 - PARKW*FLPARK(N)
RETURN
END

```

```

SUBROUTINE DIAGN(N, NCARD)
DIMENSION MESAG1(72), MESAG2(72)
INTEGER FATAL(9)
DATA (MESAG1(I), I=1,8)/80HINCORRECT CARD TYPE. PROGRAM WAS EXPECT
1 ING A TYPE-A CARD. /
DATA (MESAG2(I), I=1,8)/80HERROR IS FATAL. PROGRAM STOPPING. /
1 DATA (MESAG1(I), I=9,16)/80HINCORRECT CARD TYPE. PROGRAM WAS EXPEC
1 TING A TYPE-B CARD. /
DATA (MESAG2(I), I=9,16)/80HERROR IS FATAL. PROGRAM STOPPING. /
1 DATA (MESAG1(I), I=17,24)/80HINCORRECT LINK NUMBER ON CARD. /
1 DATA (MESAG2(I), I=17,24)/80HERROR IS NON-FATAL. PROGRAM CONTINUIN
1 G WITH POSSIBLE ERROR IN LINK DATA. /
DATA (MESAG1(I), I=25,32)/80HINCORRECT NUMBER OF LANES. MAXIMUM NU
1 MBER OF LANES PERMITTED IN SIGNET IS 3. /
DATA (MESAG2(I), I=25,32)/80HERROR IS NON-FATAL. PROGRAM CONTINUIN
1 G WITH NUMBER OF LANES ASSUMED TO BE 3. /
DATA (MESAG1(I), I=33,40)/80HFAILURE TO PLACE END CARD (COL. 1-3) A
1 FTER TURNING RADIUS DATA. /
DATA (MESAG2(I), I=33,40)/80HERROR IS NON-FATAL. PROGRAM CONTINUIN
1 G. /
DATA (MESAG1(I), I=41,48)/80HINCORRECT CARD TYPE. PROGRAM WAS EXPE
1 CTING A TYPE-C CARD. /
DATA (MESAG2(I), I=41,48)/80HERROR IS FATAL. PROGRAM STOPPING. /
1 DATA (MESAG1(I), I=49,56)/80HFAILURE TO PLACE END CARD (COL. 1-3) A
1 FTER DISCHARGE BOUNDARY DATA /
DATA (MESAG2(I), I=49,56)/80HERROR IS NON-FATAL. PROGRAM CONTINUIN

```


TABLE A-3, cont.

```

1G. DATA (MESAG1(I),I=57,64)/80HINCORRECT CARD TYPE. PROGRAM WAS EXPE
1CTING A TYPE-D CARD. /
DATA (MESAG2(I),I=57,64)/80HERROR IS FATAL. PROGRAM STOPPING.
1 /
DATA (MESAG1(I),I=55,72)/80HNUMBER OF LANES SET TO ZERO. SIGNET A
1SSUMES EACH LINK TO HAVE AT LEAST 1 LANE. /
DATA (MESAG2(I),I=55,72)/80HERROR IS NON-FATAL. PROGRAM CONTINUIN
1G WITH NUMBER OF LANES ASSUMED TO BE 1. /
DATA FATAL/1,1,2,2,2,1,2,1,2/
WRITE (6,1) NCARD
1 FORMAT (1H2,/5X,125(1H*),////20X,*ERROR ENCOUNTERED IN DATA CARD N
1UMBER*,I4)
INDEX=(N-1)*8+1
IFNC=INDEX+7
WRITE (6,2) (MESAG1(I),I=INDEX,IEND),(MESAG2(I),I=INDEX,IEND)
2 FORMAT (////30X,*ERROR IS *,8A10,///30X,8A10,////5X,125(1H*)////)
IF (FATAL(N).EQ.2) GO TO 110
RETURN
110 STOP
END

```


Appendix B

SIGNET Program Details

The following is a complete description of the inputs required by SIGNET. Input data for a simulation run are recorded on code sheets, then punched on standard 80-column cards.

The SIGNET input deck consists of the following cards in the specified sequence, as indicated in Figure B-1:

Title Card	(1 card)
General Parameters Card	(1 card)
Program Instruction Cards	(6 cards)
Vehicle Length Card	(1 card)
Link Parameters Cards	(1 card per link, all links)
Link Geometry Cards	(2 cards per link, all links)
Signal Setting Cards	(2 cards per link, input and network links)
Turning Probability Cards	(1 card per link, repeated as desired, network and input links)
Input Volume Cards	(1 card per link, repeated as desired, input links)

FIGURE B-1
INPUT DATA DECK STRUCTURE

The following gives a complete description of the contents of each card type. Specific variable names and formats may be found in Table B-1.

Title Card. This card allows the user to label the output with any alpha-numeric label up to 80 columns in length.

General Parameters Card. This card contains general information pertaining to the simulation run to be performed and the network size. The contents are described below:

<u>Field Name</u>	<u>Field Contents</u>
1. CYCLE	Amount of real-time simulated per simulation scanning cycle, in seconds. CYCLE should be between 1.0 and 5.0 seconds. A value of 2.0 seconds was used in this study.
2. NOLINK	Total number of links in the network. NOLINK must be less than 85. For larger networks program dimension statements must be changed.
3. NOUTPT	Number of output links.
4. NINPUT	Number of input links.
5. Fill Time	Duration of time simulated in order to fill the network with its initial population of vehicles, in minutes. Fill time should be approximately 15 minutes, although this may vary with the network.
6. Duration of Run	Duration of observe time in the simulation run, in minutes.
7. Statistical Summary No. 1	In addition to the automatic statistical summary which occurs at the end of the simulation run, up to 15 intermediate summaries may be requested. Field 7 contains the time,

Field Name	Field Contents
8. Statistical Summary No. 2 . . .	in minutes since the beginning of observe time, at which the first summary is to occur; Field 8, the second; etc. Any of these fields may be left blank, in which case the program assumes there are no further print options.
21. Statistical Summary No. 15	

Program Instruction Cards. This set of 6 cards contains instructions on both the proper times to perform several functions during the simulation run and the links to be considered in two extended bookkeeping operations. The contents are described below:

Field Name	Field Contents
<u>Card No. 1:</u>	
1. Turning Probability Revision Time 1	In addition to the initial turning probabilities read at the beginning of the simulation run, up to 10 revised turning probability sets may be employed. Field 1 contains the time, in minutes since the beginning of observe time, at which the first revision is to occur; Field 2, the second; etc. Any of these fields may be left blank, in which case the program assumes there are no more revisions to be made.
2. Turning Probability Revision Time 2 . . .	
10. Turning Probability Revision Time 10	

Field Name

Card No. 2:

1. Input Volume
Revision Time 1	In a manner similar to the turning probability revisions, up to 10 input volume revisions may be used. A blank field causes no more revisions to be made.
2. Input Volume
Revision Time 2	
.	
.	
10. Input Volume
Revision Time 10	

Card No. 3:

1. Re-
initialization
Time 1	As in the two cases above, a re-initialization of all bookkeeping variables may be requested. This option is used primarily in conjunction with the statistical summary option, causing a re-initialization immediately after each summary. Re-initialization after fill-time is automatic; therefore, it should not be requested here.
2. Re-
initialization
Time 2	
.	
.	
10. Re-
initialization
Time 10	

Card No. 4:

1. Begin Printout	Primarily for debugging purposes, a full printout of all vehicles' positions, velocities, accelerations, acceleration types, and turning movements may be specified. Field 1 contains the time, in seconds since the beginning of the simulation (i.e., beginning of fill time), that the printout is to begin. It is then made once each scan cycle until the time is greater than the value contained in Field 2. If both fields are blank no printout is made.
2. End Printout	

<u>Field Name</u>	<u>Field Contents</u>
3. Generator seed	To obtain a reproducible stream of random numbers it is necessary to provide a seed, or beginning number, for the pseudo-random number generator.

Card No. 5:

1. NTRACE	For more detailed analysis of travel times a route up to 15 links in length may be specified through the network. Field 1 contains the number of links in the route. Field 2 contains the first link in the route, Field 3 the second, etc. NTRACE must be greater than 0.
2. LKTRCE(1)	
3. LKTRCE(2)	
.	
.	
16. LKTRCE(15)	

Card No. 6:

1. NQ	Similarly, a set of up to 10 links may be specified for queue length analysis. Field 1 contains the number of links in this set. Links are ordered within the set by link number. Field 2 contains the first link number, Field 3 the second, etc. NQ must be greater than 0.
2. LQ(1)	
3. LQ(2)	
.	
.	
11. LQ(10)	

Vehicle Length Card. This card contains the parameters defining the effective length distribution for both cars and trucks. The contents are described below:

<u>Field Name</u>	<u>Field Contents</u>
1. ELMN ₁ (Cars)	Mean effective length for cars, in feet. A value of 22.0 feet is reported in the literature for this variable.

Field Name	Field Contents
2. ELSIG ₁ (Cars)	Effective Length standard deviation for cars.
3. ELMN ₂ (Trucks)	Mean effective length for trucks, in feet.
4. ELSIG ₂ (Trucks)	Effective length standard deviation for trucks.

Link Parameters Card. This card contains, for a specific link, lane relationships, link numbering relationships, and the parameters defining the target velocity distributions for cars and trucks on that link. One card of this type must be provided for each link. The contents are described below:

Field Name	Field Contents
1. i	Link number (identification). Links must be numbered in succession.
2. NOLANE _i	Width of link i in number of moving lanes at the intersection approach. NOLANE _i = 1, 2, or 3.
3. LANTN _{i,1}	Lane required for turn. LANTN _{i,k} = lane vehicle must be in to make turn number k.
4. LANTN _{i,2}	Turn numbers are discussed on page 55.
.	
.	
.	
8. LANTN _{i,6}	
9. NREC _{i,1}	Receiving link. NREC _{i,k} = link which receives traffic from turning movement number k.
10. NREC _{i,2}	
.	
.	
.	
14. NREC _{i,6}	

Field Name	Field Contents
15. LREC _{i,1}	Receiving Lane.
.	LREC _{i,k} = lane which receives
.	traffic from turning
.	movement number k.
20. LREC _{i,6}	
21. IOPP _{i,1}	Opposing link.
22. IOPP _{i,2}	Numbering conventions described on
.	page 148.
.	
25. IOPP _{i,5}	
26. TVELMN _{i,cars}	Mean target velocity for cars, in
	miles per hour.
27. TVELSIG _{i,cars}	Target velocity standard deviation
	for cars.
28. TVELMN _{i,trucks}	Mean target velocity for trucks,
	in miles per hour.
29. TVELSIG _{i, trucks}	Target velocity standard deviation
	for trucks.

Link Geometry Cards. This pair of cards contains, for a specified link, all information related to the link geometry. The entire set of cards may be obtained as punched output from PRESIG, the companion geometry determination program, although this is not necessary. One pair of cards is required for each link. The contents are described below:

Field Name	Field Contents
<u>Card No. 1:</u>	
1. i	Link number (identification). Links must be numbered in succession.
2. $STPLINE_i$	Coordinate of the stopline, in feet.
3. $DBNDRY_{i,1}$	$DBNDRY_{i,k}$ = discharge boundary for turning movement number k.
4. $DBNDRY_{i,2}$	
.	
.	
8. $DBNDRY_{i,6}$	
9. $XLINK_{i,1}$	$XLINK_{i,k}$ = length of lane k. (k = 1 = rightmost lane)
10. $XLINK_{i,2}$	
11. $XLINK_{i,3}$	
12. $BEGINPT_{i,1}$	$BEGINPT_{i,k}$ = beginning point of turn number k.
13. $BEGINPT_{i,2}$	
.	
.	
17. $BEGINPT_{i,6}$	
<u>Card No. 2:</u>	
18. $RADIUS_{i,1}$	$RADIUS_{i,k}$ = radius of turn number k.
19. $RADIUS_{i,2}$	
.	
.	
23. $RADIUS_{i,6}$	

Signal Setting Cards. This pair of cards contains, for a specified link, the fixed-time traffic signal settings. One pair must be provided for each input link and each network link. The contents are described below:

Field Name	Field Contents
<u>Card No. 1:</u>	
1. i	Link number (identification). Links must be numbered in succession.
2. $LCYCLE_i$	Cycle length, in seconds.
3. $NOPHAS_{i,1}$	$NOPHAS_{i,k}$ = number of indications to lane k .
4. $NOPHAS_{i,2}$	$NOPHAS_{i,k}$ must be less than 6.
5. $NOPHAS_{i,3}$	
6. $NPHASE_{i,1,1}$	$NPHASE_{i,j,k}$ = number of k^{th} indication to lane j .
7. $NPHASE_{i,1,2}$	Indication numbers are defined in Table 5.
.	
.	
.	
10. $NPHASE_{i,1,5}$	
11. $NPHASE_{i,2,1}$	
12. $NPHASE_{i,2,2}$	
.	
.	
.	
20. $NPHASE_{i,3,5}$	

Field Name	Field Contents
<u>Card No. 2:</u>	
21. $LPHASE_{i,1,1}$	$LPHASE_{i,j,k}$ = duration of k^{th} indication to lane j , in seconds.
22. $LPHASE_{i,1,2}$	
.	
.	
.	
25. $LPHASE_{i,1,5}$	
26. $LPHASE_{i,2,1}$	
.	
.	
.	
35. $LPHASE_{i,3,5}$	
36. $LOFFSET_{i,1}$	$LOFFST_{i,k}$ = offset to beginning of first indication on lane k , in seconds.
37. $LOFFST_{i,2}$	
38. $LOFFST_{i,3}$	

Turning Probability Card. This card contains, for a specified link, the cumulative turning probabilities for the 6 possible turning movements. One card must be provided for each network and input link. The contents are described below:

Field Name	Field Contents
1. i	Link number. Links must be numbered in succession.
2. $PTURN_{i,1}$	$PTURN_{i,1}$ = probability that a vehicle leaving link i will use turning movement number 1.

Field Name	Field Contents
3. $PTURN_{i,2}$	$PTURN_{i,k} - PTURN_{i,k-1}$ = probability that a vehicle leaving link i will use turning movement Number k .
.	
.	
.	
7. $PTURN_{i,6}$	$PTURN$ is a cumulative distribution.

Input Volume Card. This card contains, for a specified link, the hourly input volume and the percent trucks. One card must be provided for each input link. The contents are described below:

Field Name	Field Contents
1. i	Link number. Links must be numbered in succession.
2. VOL_i	Input Volume, vehicles per hour.
3. $PRTRUCK_i$	Percent trucks.

The use of successive sets of Turning Probability Cards and Input Volume Cards provides for modification of certain link characteristics while the simulation is in progress: viz., input volumes, input truck percentages, and turning probabilities. Care should be taken that these sets are in the correct order, as specified by the Program Instruction Cards. If revised turning probabilities and input volumes are required simultaneously, the turning probability cards should appear first. No items except those appearing in these two sets can be changed while a simulation run is in progress.

TABLE B-1
SIGNET Punched Card Input Format

Card Description	Card No.	Column	Symbol	Meaning	Units	FORTAN Format	Notation
Title Card	-	1-80		Any message	---	8A10	----
General Parameters	-	1-2	CYCLE	Scan cycle	sec.	P2.0	X.
		3-5	NOLINK	No. of Links in network	---	I3	XXX
		6-8	NOUTPT	No. of Output Links	---	I3	XXX
		9-11	NINPUT	No. of Input Links	---	I3	XXX
		12-15	MINITIM	Fill Time	min.	I4	XXXX
		16-20	MLRUN	Observe Time	min.	I5	XXXXX
		21-24	SUMTIM(1)	First Summary Time	min.	I4	XXXX
		25-28	SUMTIM(2)	2nd Summary Time	min.	I4	XXXX
		.	.	.	.	.	.
		.	.	.	.	.	.
Program Instructions	1	77-80	SUMTIM(15)	15th Summary	min.	I4	XXXX
		1-4	NEWPR(1)	1st turning prob. revision	min.	I4	XXXX
		5-8	NEWPR(2)	2nd turning prob. revision	min.	I4	XXXX
		.	.	.	.	.	.
		.	.	.	.	.	.

TABLE B-1, Cont.

<u>Card Description</u>	<u>Card No.</u>	<u>Column</u>	<u>Symbol</u>	<u>Meaning</u>	<u>Units</u>	<u>FORTRAN Format</u>	<u>Notation</u>
		37-40	NEWPR(10)	10th turning prob. revision	min.	I4	XXXX
	2	1-4	NEWVOL(1)	1st input vol. revision	min.	I4	XXXX
		5-8	NEWVOL(2)	2nd input vol. revision	min.	I4	XXXX
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.
		37-40	NEWVOL(10)	10th input vol. revision	min.	I4	XXXX
Program	3	1-4	NEWINIT(L)	1st reinit-ialization	min.	I4	XXXX
		5-8	NEWINIT(2)	2nd reinit-ialization	min.	I4	XXXX
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.
		37-40	NEWINIT(10)	10th reinit-ialization	min.	I4	XXXX
	4	1-5	IDEFST	Start printout	sec.	I5	XXXXX
		6-10	IDEFEND	End printout	sec.	I5	XXXXX
		11-30	ISEED	Random number seed	---	020	20 digit (octal)

TABLE B-1, Cont.

<u>Card Description</u>	<u>Card No.</u>	<u>Column</u>	<u>Symbol</u>	<u>Meaning</u>	<u>Units</u>	<u>FORTAN Format</u>	<u>Notation</u>
	5	1-2	NTRACE	No. links in route	---	I2	XX
		3-5	LKTRCE(1)	1st link in route	---	I3	XXX
		6-8	LKTRCE(2)	2nd link in route	---	I3	XXX
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.
		45-47	LKTRCE(15)	15th link in route	---	I3	XXX
	6	1-2	NQ	No. of links to be analyzed for queue length	---	I2	XX
		3-5	LQ(1)	1st link in set	---	I3	XXX
		6-9	LQ(2)	2nd link in set	---	I3	XXX
Vehicle Lengths		.	.	.	.	.	.
		.	.	.	.	.	.
		30-32	LQ(10)	10th link in set	---	I3	XXX
	1-5	ELMN(cars)		Mean EL, cars	ft.	F5.2	XX.XX
	6-10	ELSIG(cars)		EL std. dev., cars	ft.	F5.2	XX.XX
	11-15	ELMN(trucks)		Mean EL, trucks	ft.	F5.2	XX.XX
	16-20	ELSIG		EL std. dev., trucks	ft.	F5.2	XX.XX

TABLE B-1, Cont.

<u>Card Description</u>	<u>Card No.</u>	<u>Column</u>	<u>S-mbol</u>	<u>Meaning</u>	<u>Units</u>	<u>FORTTRAN Format</u>	<u>Notation</u>
Link Parameters	-	6	NOLANE	No. of Lanes	---	I1	X
		7	LANTN(1)	Lane req. for turn 1	---	I1	X
		8	LANTN(2)	Lane req. for turn 2	---	I1	X
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.
		12	LANTN(6)	Lane req. for turn 6	---	I1	X
		13-15	NREC(1)	Receiving link	---	I3	XXX
		16-18	NREC(2)	Receiving link	---	I3	XXX
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.
		28-30	NREC(6)	Receiving link	---	I3	XXX
		31-33	LREC(1)	Receiving link	---	I3	XXX
		.	.	.	.	.	.
		.	.	.	.	.	.
		46-48	LREC(6)	Receiving link	---	I3	XXX
		49-51	IOPP(1)	Opposing link	---	I3	XXX
		.	.	.	.	.	.
		.	.	.	.	.	.
		61-63	IOPP(5)	Opposing link	---	I3	XXX
		64-66	TVELMN(cars)	Mean target vel. (cars)	m.p.h.	F3.0	XX.

TABLE B-1, Cont.

<u>Card</u> <u>Description</u>	<u>Card</u> <u>No.</u>	<u>Column</u>	<u>Symbol</u>	<u>Meaning</u>	<u>Units</u>	<u>FORTAN</u> <u>Format</u>	<u>Notation</u>
Link Geometry	1	70-72	TVELMN (trucks)	Mean target vel. (trucks)	m.p.h.	F3.0	XX.
		73-75	TVELSIG (trucks)	Std. dev. of targ. vel., trucks	m.p.h.	F3.0	XX.
		6-9	STPLINE	Stop-line	ft.	F4.0	XXXX
		10-13	DBNDRY (1)	Discharge Boundary	ft.	F4.0	XXXX
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.
		30-33	DBNDRY (6)	Discharge Boundary	ft.	F4.0	XXXX
		37-40	XLINK (1)	Length of lane 1	ft.	F4.0	XXXX
		38-41	XLINK (2)	Length of lane 2	ft.	F4.0	XXXX
		42-45	XLINK (3)	Length of lane 3	ft.	F4.0	XXXX
		46-49	BEGINPT (1)	Begin-turn point	ft.	Fr.0	XXXX
		.	.	.	.	.	.
		.	.	.	.	.	.
		66-69	BEGINPT (6)	Begin-turn point	ft.	F4.0	XXXX

TABLE B-1, Cont.

<u>Card Description</u>	<u>Card No.</u>	<u>Column</u>	<u>Symbol</u>	<u>Meaning</u>	<u>Units</u>	<u>FORTAN Format</u>	<u>Notation</u>
Link Geometry (cont.)	2	6-8	RADIUS (1)	Radius of turn 1	ft.	F3.0	XXX
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.
		21-23	RADIUS (6)	Radius of turn 6	ft.	F3.0	XXX
Signal Settings	1	6-8	LCYCLE	Cycle length	sec.	I3	XXX
		9	NOPHAS (1)	No. of phases lane 1	---	I1	X
		10	NOPHAS (2)	No. of phases lane 2	---	I1	X
		11	NOPHAS (3)	No. of phases lane 3	---	I1	X
		12-13	NPHASE (1,1)	Phase No. 1, lane 1	---	I2	XX
		14-15	NPHASE (1,2)	Phase No. 2, Lane 1	---	I2	XX
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.
		20-21	NPHASE (1,5)	Phase No. 5, lane 1	---	I2	XX
		22-23	NPHASE (2,1)	Phase No. 1, lane 2	---	I2	XX
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.

TABLE B-1, Cont.

<u>Card Description</u>	<u>Card No.</u>	<u>Column</u>	<u>Symbol</u>	<u>Meaning</u>	<u>Units</u>	<u>FORTAN Format</u>	<u>Notation</u>
		40-41	NPHASE (3,5)	Phase No. 5, lane 3	---	I2	XX
	2	6-8	LPHASE (1,1)	Duration #1, lane 1	sec.	I3	XXX
		.	.	.	.	.	.
		.	.	.	.	.	.
		18-20	LPHASE (1,5)	Duration #5, lane 1	sec.	I3	XXX
		21-23	LPHASE (2,1)	Duration #1, lane 2	sec.	I3	XXX
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.
		48-50	LPHASE (3,5)	Duration #5, lane 3	sec.	I3	XXX
		51-53	LOFFST (1)	Offset, lane 1	sec.	I3	XXX
		54-56	LOFFST (2)	Offset, lane 2	sec.	I3	XXX
		57-59	LOFFST (3)	Offset, lane 3	sec.	I3	XXX
Turning Probability	-	6-9	PTURN (1)	Turning Probability, turn 1	---	F4.2	X.XX
		10-13	PTURN (2)	Turning Probability, turn 2 (cumulative)	---	F4.2	X.XX
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.
		.	.	.	.	.	.

TABLE B-1, Cont.

<u>Card Description</u>	<u>Card No.</u>	<u>Column</u>	<u>Symbol</u>	<u>Meaning</u>	<u>Units</u>	<u>FORTAN Format</u>	<u>Notation</u>
		26-29	PTURN (6)	Turning Probability, turn 6 (cumulative)	---	F4.2	X.XX
Input Volume	-	6-11 12-15	VOL PRTRUCK	Volume Percent trucks	v.p.h. ---	F6.0 F4.2	XXXXXX X.XX

A complete listing of the SIGNET program code is given in Table B-2. The program is written entirely in FORTRAN IV. Approximately 121000_8 central memory words are required to run the program at its current 84-link, 1500 vehicle capacity; the size of the program may be reduced to match the requirements of the application by reducing both the link capacity and the vehicle capacity. The means of doing this is explained below.

Several routines referenced by SIGNET are system routines on the CDC 6500. It is probable that the program could not be run on other installations without provision being made for supplying at least some of these routines or their equivalent. They are as follows:

EXP(X): A real function for finding e^X .

SQRT(X): A real function used for finding X .

ALOG(X): A real function used for finding $\ln(x)$.

RANF(A): A real function used to generate a sequence of pseudo-random numbers which are uniformly distributed in $(0,1)$. In order to generate a sequence of random numbers used RANF with a zero argument:

$Y = \text{RANF}(0).$

In order to restart the sequence at some given point, use RANF with the argument set to some positive seed, written in octal notation:

$X = 17162534251617262534B;$

$Y = \text{RANF}(X);$

then use $\text{RANF}(0).$

The random number is determined from:

$$X_{i+1} = K * X_i \text{ MOD } 2^{48}$$

where

X_{i+1} = the new random number.

X_0 = the seed; and

$K = 200000000000000055364B$.

ISHFTR(X,M): Causes word X to be shifted right M bits.

This COMPASS function is used in unpacking the ICAR word.

ISHFTL(X,M): Causes word X to be shifted left M bits.

This COMPASS function is used in packing the ICAR word.

The CDC 6500 employs a 60-bit word. For installations employing a smaller word revisions would be necessary to the functions IBITM and JBITM (used for unpacking and packing the ICAR word, respectively).

Storage for vehicles is allocated in a way that minimizes the amount of wasted space. In its present state the program will accommodate up to 1500 vehicles at any time. Reducing this capacity for applications with potentially fewer vehicles will result in reduced field length and greater economy. The reduction is easily accomplished by changing the dimension of the variables POSN, VEL, ACCEL, ICAR, NXTCAR, and IPRECAR in common block A1 and the variable IHOLE in common block ZQ. The dimension is equal to the vehicular capacity of the program.

If the program capacity is exceeded at any time a message is printed to that affect and the program continues. As a guide in determining what the program capacity should be, the maximum number of vehicles in the network is printed at the end of each run.

The SIGNET program deck structure is shown in Figure B-2.

TABLE B-2
SIGNET PROGRAM CODE

C	*****	1	10
C	*****	2	70
C		3	80
C		4	40
C	SIGNET-----A PROGRAM FOR THE SIMULATION OF A	5	70
C	TRAFFIC SIGNAL NETWORK	6	70
C		7	40
C	*****	8	40
C	*****	9	170
C		10	170
C		11	170
C		12	170
C	PROGRAM SIGNET (INPUT,OUTPUT,TAPES=INPUT,TAPES=OUTPUT)	13	170
C	COMMON /A1/ IPH,IPHT,IPAX,ISIP,IPSTIC(84,1),TLCAP(84,1),VCL(	14	140
C	11500),POCN(1500),ACCEL(1500),ICAR(1500),KXTCAR(1500),IPRECAR(1500)	15	150
C	COMMON /T1/ INOLE(1500),LASTONE	16	160
C	COMMON /PARAM/ AB,FRFF,REACT,CYCLE	17	170
C	COMMON /LINKS/ LANTN(84,1),XLINK(84,1)	18	180
C	COMMON /TIS/ LCYCLE(P4),NOMAS(84,1),NOMAS(84,1),LPHASE(84,1,5)	19	140
C	1,LOFFST(84,1)	20	700
C	COMMON /RELAT/ LINTAP(84),NOLINK,LANCA(84,1),NOLINK(84)	21	710
C	COMMON /GEOMET/ PATIUS(84,1),RFGINPT(84,1)	22	720
C	COMMON /ORP/ ICOP(16,5)	23	730
C	COMMON /PSK/ MASK(70)	24	740
C	COMMON /GEOM/ STOLINE(84),CONDST(84,1)	25	750
C	COMMON /TURN/ NPEC(84,1),LPEC(84,1),RTURN(84,1),PTD(84,1)	26	760
C	COMMON /R2/ TTINF(84),TDIST(84),TTINF(17,107),TTINF,TTINF,TTINF	27	770
C	117,3,14)	28	780
C	COMMON /RUK/ NHOLD(84,1)	29	790
C	COMMON /OU/ LOUFUP(15,1),LO(10),AO	30	790
C	COMMON /PV/ P(84,1),V(84,1)	31	710
C	COMMON /TM/ TIME	32	720
C	COMMON /TRCF/ KTRCF(15),KTRCF(15),INTTDC,KNTTDC(15)	33	730
C	COMMON /KUNT2/ NCAR(84),NOUTPT,ISTART,IPND,KCCN(84)	34	740
C	COMMON /TINK/ INITIP,LOUN,MINITIM	35	750
C	COMMON /CAPACTY/ LICAP	36	760
C	COMMON /PARAM2/ TVELMN(84,1),TVFLSIG(84,1),ACCM(17),ACCSIG(17),DECM	37	770
C	IN(17),DECSIG(17),FLMN(17),FLSIG(17),PTRUCK(84)	38	780
C	OTMENSION MESSAGE(10)	39	790
C	OTMENSION NFMPP(10), NFMVOL(10)	40	400
C	OTMENSION NFMINT(10)	41	410
C	INTGER SUMTIM(15),SUMKNT	42	420
C	REAL XLINK,LFRFF,LDELAY	43	430
C	DATA LNCAP/70/	44	440
C	DATA ISI7E/1500/	45	450
C	DATA AB,FRFF,REACT,ACCP4,ACCSIG,DECMN,DECSIG/75,10,75,75,4,1,	46	460
C	1,5,4,4,4,4,5/	47	470
C	MASK(1)=10	48	480
C	MASK(2)=30	49	490
C	MASK(3)=70	50	500
C	MASK(4)=170	51	510
C	MASK(5)=370	52	520
C	MASK(6)=770	53	530
C	MASK(7)=1770	54	540
C	MASK(8)=7770	55	550
C	MASK(12)=77770	56	560
C	MASK(13)=177770	57	570
C	TDIST=0.	58	580
C	TTIME=0.	59	590
C		60	600

TABLE B-2, cont.

C C C C C C C C C C	*****	A 610
	READ PROGRAM DATA	A 620
	*****	A 630
	*****	A 640
	*****	A 650
	*****	A 660
	READ (5,320) (MESSAGE(I),I=1,4)	A 670
	READ (5,330) CYCLE,MOLINK,MOUTPT,MINPUT,MINITIM,MLRUN,(SUMTIM(I),I=1,15)	A 680
	INITIM=MINITIM*60	A 690
	LPDA=MLRUN*60	A 700
C C C C C C C C C C	DO 100 I=1,6	A 710
	IF (SUMTIM(I).EQ.0) GO TO 100	A 720
	SUMTIM(I)=SUMTIM(I)+40+INITIM	A 730
	100 CONTINUE	A 740
	READ (5,340) (FMP2(I),I=1,10)	A 750
	READ (5,341) (FMPV(L(I),I=1,10)	A 760
	READ (5,342) (FMPVIT(I),I=1,10)	A 770
	READ (5,350) ICFST,ICFFN,ISCF	A 780
	EN=ENF(ISCFO)	A 790
	READ (5,360) NTPACE,(LKTCF(N),N=1,NTPACE)	A 800
C C C C C C C C C C	READ (5,370) NC,(L7(N),N=1,NC)	A 810
	READ (5,380) (FLMN(I),FLSIG(I),I=1,7)	A 820
	DO 110 L=1,MOLINK	A 830
	READ (5,390) MOLANG(L),(LANH(L,J),J=1,4),(MPC(L,J),J=1,5),(LDEC(L,J),J=1,5),(ICPP(L,J),J=1,5),(TVFLMN(L,J),TVFLSIG(L,J),J=1,7)	A 840
	110 CONTINUE	A 850
	ICPP(I) IS THE OPPOSING LINK WHICH HAS BEEN ESTABLISHED WITH THE LINK UNDER CONSIDERATION. ICPP(I) AND IOPP(I) ARE OPPOSING LINKS TO THE RIGHT. IOPF(L) AND IOPP(L) ARE OPPOSING LINKS TO THE LEFT.	A 860
	DO 120 L=1,MOLINK	A 870
	120 READ (5,400) STPLINE(L),(MPCMDY(L,J),J=1,4),(XLMN(L,J),J=1,7),(ICGTP(L,J),J=1,5),(RADIUS(L,J),J=1,5)	A 880
	NO=ACUTPT+1	A 890
	DO 130 L=NO,MOLINK	A 900
C C C C C C C C C C	130 READ (5,410) LCYCLE(L),(NOPHASE(L,J),J=1,7),(NOPHASE(L,J,V),V=1,5),IJ=1,7),(LOPHASE(L,J,K),K=1,5),J=1,7),(LCFFST(L,I),I=1,7)	A 910
	DO 140 L=1,MOUTPT	A 920
	LCYCLE(L)=60	A 930
	NL=MOLANG(L)	A 940
	DO 140 LN=1,NL	A 950
	NOPHASE(L,LN)=1	A 960
	NOPHASE(L,LN,1)=2	A 970
	LOPHASE(L,LN,1)=60	A 980
	140 LCFFST(L,LN)=0	A 990
	*****	A 1000
C C C C C C C C C C	WRITE PROGRAM DATA	A 1010
	*****	A 1020
	*****	A 1030
	*****	A 1040
	*****	A 1050
	WRITE (5,420) (MESSAGE(I),I=1,4)	A 1060
	WRITE (5,430) CYCLE,MINITIM,MLRUN	A 1070
	DEC1=DECMN(1)	A 1080
	DEC2=DECMN(2)	A 1090
	WRITE (5,440) AD,FREFK,REACT,ACCMN(1),ACCSIG(1),ACCMN(2),ACCSIG(2),DEC1,DEC1SIG(1),DEC2,DEC2SIG(2),FLMN(1),FLSIG(1),FLMN(2),FLSIG(2),I	A 1100

TABLE B-2, cont.

	TVFLMNL(I,1),TVFLSTG(L,1),TVFLMNL(I,2),TVFLSTG(L,2),I=1,NOLINK)	A 1770
	DO 160 L=1,NOLINK	A 1780
	DO 150 I=1,2	A 1790
	TVFLMNL(I,I)=TVFLMNL(I,I)*1.47	A 1800
150	TVFLSTG(L,I)=TVFLSIG(L,I)*1.47	A 1810
	WRITE (5,450)	A 1820
	DO 160 LINK=1,NOLINK	A 1830
	NCL=ACLANE(LINK)	A 1840
	WRITE (5,460) LINK	A 1850
	DO 160 LANE=1,NCL	A 1860
	WRITE (5,470) LANE,LTYPE(LINK),ACPHAS(LINK,LANE),ACPHASE(LINK,LANE,	A 1870
	1,1),LPHASE(LINK,LANE,1),LDEFT(LINK,LANE)	A 1880
	NOP=ACPHAS(LINK,LANE)	A 1890
	IF (NOP,0,1) GO TO 160	A 1900
	WRITE (5,480) (ACPHASE(LINK,LANE,K),LPHASE(LINK,LANE,K),K=2,NOP)	A 1910
150	CONTINUE	A 1920
	NOTEC=1	A 1930
	ISTART=NOLINK+1	A 1940
	IFND=NOLINK	A 1950
C	*****	A 1960
C		A 1970
C	*****	A 1980
C	DEAN AND WRITE TRAFFIC DATA	A 1990
C	*****	A 2000
C		A 2010
C	*****	A 2020
	WRITE (5,490)	A 2030
	DO 170 L=1,NOLINK	A 2040
	WRITE (5,500) L,INRT(L,1),LDEFT(L,1),J=1,1, (INPH(L,1),I=1,1)	A 2050
170	CONTINUE	A 2060
	WRITE (5,510)	A 2070
	DO 180 L=1,NOLINK	A 2080
180	WRITE (5,520) L,STPLINK(I), (CROSS-Y(L,I),T=1,1, (FLINK(L,1),I=1,1),	A 2090
	1,RADIUS(L,I),I=1,6)	A 2100
	DO 210 I=1,10	A 2110
	IF (NEWMO(I),0,0) GO TO 190	A 2120
	NEWPR(I)=NEWMO(I)*50+INITIM	A 2130
190	IF (NEWVOL(I),0,0) GO TO 200	A 2140
	NEWVCL(I)=NEWVOL(I)*60+INITIM	A 2150
200	IF (NEWINT(I),0,1) GO TO 210	A 2160
	NEWJIT(I)=NEWINT(I)*50+INITIM	A 2170
210	CONTINUE	A 2180
	CALL INITRN	A 2190
	WRITE (5,530)	A 2200
	CALL INIVOL	A 2210
C	*****	A 2220
C		A 2230
C	*****	A 2240
C	INITIALIZE	A 2250
C	*****	A 2260
C		A 2270
C	*****	A 2280
	CALL INIT	A 2290
	ITIME=0	A 2300
	INIMT=0	A 2310
	DO 220 L=1,NCL	A 2320
	DO 220 LN=1,3	A 2330
220	L7UFUF(L,LN)=0	A 2340
	DO 230 L=1,NOLINK	A 2350
	LINCAR(L)=0	A 2360
	DO 230 K=1,3	A 2370

TABLE B-2, cont.

	IF FIRSTC(L,K) = 0	A 1830
	LASTC(L,K) = 0	A 1840
275	LASTC(L,K) = 0	A 1850
	IT = INITIALRUN + 1	A 1860
	INTT = INT(CYCLE)	A 1870
	CUMKNT = 1	A 1880
	KNTVOL = 1	A 1890
	KNTTAN = 1	A 1900
	KNTINIT = 1	A 1910
	LASTCNP = 0	A 1920
	INFTN = 1	A 1930
	IFP = 15170	A 1940
	IN = 0	A 1950
C		A 1960
C	*****	A 1970
C		A 1980
C	RESTN TIME LOOP	A 1990
C		A 2000
C	*****	A 2010
C		A 2020
	CUM = TIMEFLP(0)	A 2030
	ON 310 KNTIN = 1, ITT, INTT	A 2040
	ITIME = KNTIN - 1	A 2050
	IF (ITIME, LT, INITIME) GO TO 240	A 2060
	IF (CUMKNT, GT, 10) GO TO 240	A 2070
	IF (CUMTAN(CUMKNT), EQ, 0) GO TO 240	A 2080
	IF (ITIME, LT, SUMTIME(CUMKNT)) GO TO 240	A 2090
	CUM = TIMEFLP(1)	A 2100
	CALL SUBARY (VOL, ITIME, SUMKNT)	A 2110
	SUMKNT = SUMKNT + 1	A 2120
	CUM = TIMEFLP(1)	A 2130
240	CONTINUE	A 2140
	IF (ITIME, LT, INITIME) GO TO 240	A 2150
	IF (KNTTAN, GT, 10) GO TO 250	A 2160
	IF (NEWVOL(KNTTAN), EQ, 0) GO TO 250	A 2170
	IF (ITIME, LT, NEWVOL(KNTTAN)) GO TO 250	A 2180
	KNTTAN = 1 + KNTTAN	A 2190
	KTIM = ITIME/40 - MINITIM	A 2200
	WRITE (6, 540) KTIM	A 2210
	CALL INITRN	A 2220
250	IF (ITIME, LT, INITIME) GO TO 240	A 2230
	IF (KNTVOL, GT, 10) GO TO 260	A 2240
	IF (NEWVOL(KNTVOL), EQ, 0) GO TO 260	A 2250
	IF (ITIME, LT, NEWVOL(KNTVOL)) GO TO 260	A 2260
	KNTVOL = 1 + KNTVOL	A 2270
	KTIM = ITIME/60 - MINITIM	A 2280
	WRITE (6, 550) KTIM	A 2290
	CALL INIVOL	A 2300
260	CONTINUE	A 2310
	IF (ITIME, LT, INITIME) GO TO 270	A 2320
	IF (KNTINIT, GT, 10) GO TO 270	A 2330
	IF (NEWINIT(KNTINIT), EQ, 0) GO TO 270	A 2340
	IF (ITIME, LT, NEWINIT(KNTINIT)) GO TO 270	A 2350
	KNTINIT = KNTINIT + 1	A 2360
	CALL INIT	A 2370
270	CONTINUE	A 2380
	IF (ITIME, GT, INITIME + 1, AND, INIKNT, EQ, 0) GO TO 240	A 2390
	GO TO 240	A 2400
C		A 2410
C	*****	A 2420
C		A 2430

TABLE B-2, cont.

C	RF=INITIALIZE FOR RUN-TIME	A 2440
C		A 2450
C	*****	A 2460
C		A 2470
240	CONTINUE	A 2480
	INITIAL=1	A 2490
	CALL INIT	A 2500
240	CONTINUE	A 2510
C		A 2520
C	*****	A 2530
C		A 2540
C	SCAN AND UPDATE	A 2550
C		A 2560
C	*****	A 2570
C		A 2580
	CALL INPLIN	A 2590
	IF (IDFFST.F0.0.0.0).IDFFEND.F0.0) GO TO 300	A 2600
	IF (ITIME.LT.IDFFST) GO TO 100	A 2610
	IF (ITIME.GT.IDFFEND) GO TO 100	A 2620
	CALL OFFINE	A 2630
240	CONTINUE	A 2640
	CALL UPDATE	A 2650
240	CONTINUE	A 2660
C		A 2670
C	*****	A 2680
C		A 2690
C	SUMMARY FOR RUN	A 2700
C		A 2710
C	*****	A 2720
C		A 2730
	CUMTIME(LP(1))	A 2740
	CALL SUMMARY (VOL,ITIME,SLKMT)	A 2750
	WRITE (4,540) IMAX	A 2760
	STOP	A 2770
C		A 2780
320	FORMAT (A410)	A 2790
330	FORMAT (F7.0,1I1,I1,I5,1F14)	A 2800
340	FORMAT (10I4)	A 2810
350	FORMAT (2I5,0P0)	A 2820
360	FORMAT (12,15I1)	A 2830
370	FORMAT (12,10I1)	A 2840
380	FORMAT (4F5,2)	A 2850
390	FORMAT (5X,7I1,17I1,4F3.0)	A 2860
400	FORMAT (5X,16F4.0,5X,4F3.0)	A 2870
410	FORMAT (5X,1I,1I,15I2,5X,1F12,1I2)	A 2880
420	FORMAT (1H1,4X,125(1H*),5X,125(1H*),5X,125(1H*),///62X, 11MS T	A 2890
	16 M E T,///51X, 13H1 PROGRAM FOR THE SIMULATION OF A,167X, 22HTRAFF	A 2900
	2IC SIGNAL NETWORK,///22X,8A10,///5X,125(1H*),5X,125(1H*),5X,125	A 2910
	1(1H*)	A 2920
430	FORMAT (///10X, 13HSCAN CYCLE = ,F1.1, 8H SECONDS,///10X, 22MIN	A 2930
	INITIALIZATION TIME = ,I4, 8H MINUTES,20X, 11HQUV TIME = ,I5, 8H M	A 2940
	INUTES)	A 2950
440	FORMAT (///10X, 14HPROGRAM PARAMETERS,///10X, 23HCHARACTERISTIC S	A 2960
	PEED = ,F5.2, 14H FEET PER SECOND,///10X, 10HPROPORTIONALITY COEFFI	A 2970
	PCFNT = ,F4.2,///10X, 16HREACTION TIME = ,F4.2, 8H SECONDS,///10X,	A 2980
	12HMAXIMUM DESIRED ACCELERATION,20X, 4HCARS,10X, 7HMEAN = ,F5.2	A 2990
	4, 21H FEET PER SECOND SQD.,10X, 12HSTD. DEV. = ,F4.2, 21H FEET PER	A 3000
	5 SECOND SQD.,20X, 6HTRUCKS,8X, 7HMEAN = ,F5.2, 21H FEET PER SEC	A 3010
	60ND SQD.,10X, 12HSTD. DEV. = ,F5.2, 21H FEET PER SECOND SQD.,///10X	A 3020
	7, 22HMINIMUM DESIRED DECELERATION,20X, 4HCARS,10X, 7HMEAN = ,F5	A 3030
	4.2, 16H FEET PER SECOND,10X, 12HSTD. DEV. = ,F4.2, 21H FEET PER SE	A 3040

TABLE B-2, cont.

DO 110 L=1,N	A	280
CALL STIX (HT0,10,0,0,10,0,0,1,7,CUTHE,LENGTH,17,LINK)	A	300
DO 110 I=1,14	A	300
110 WRITE(L,LA,I)HT0(I)	A	310
RETURN	A	320
END	A	330
FLORCUTINE INITN	C	10
COMMON /PFLAT/ LINT0(14),NOLINK,LANTRF(14,1),NOLANE(14)	C	20
COMMON /TURN/ NPT(14,6),LPCT(14,6),PTURN(14,6),PTON(14,2)	C	30
COMMON /RWNT7/ NCS(14),AQTTOT,ISTART,IFAN,NCT(14)	C	40
COMMON /LINKS/ LANTN(14,6),LL(14,1)	C	50
REAL LL(1)	C	60
K=NCUTPT+1	C	70
DO 100 L=N,NOLINK	C	80
100 IF (L,100) (PTURN(L,J),J=1,6)	C	90
WRITE (4,770)	C	100
DO 170 I=1,NOUTPT	C	110
NLN=NOLANE(I)	C	120
DO 170 (110,170,170), NLAT	C	130
110 PTURN(I,1)=1.	C	140
PTURN(I,2)=0.	C	150
PTURN(I,3)=1.	C	160
PTURN(I,4)=1.	C	170
PTURN(I,5)=1.	C	180
PTURN(I,6)=1.	C	190
DO 170 170	C	200
170 PTURN(I,1)=0.	C	210
PTURN(I,2)=0.	C	220
PTURN(I,3)=0.5	C	230
PTURN(I,4)=1.	C	240
PTURN(I,5)=1.	C	250
PTURN(I,6)=1.	C	260
130 CONTINUE	C	270
DO 140 L=N,NOLINK	C	280
WRITE (4,290) L,(PTURN(L,J),J=1,6),(LANTN(L,J),J=1,6)	C	290
140 CONTINUE	C	300
NOUT=NOUTPT+1	C	310
DO 200 L=N,NOUT,NOLINK	C	320
IF (NOLANE(LINK),0,1) GO TO 200	C	330
NCL=NCLANE(LINK)	C	340
N=NCT(LINK,1)	C	350
IF (N,0,0) GO TO 100	C	360
DO 200 L=N,NCL	C	370
IF (LANC,0,1) GO TO 200	C	380
IF (PTURN(LINK,4),IF,PTURN(LINK,2)) GO TO 150	C	390
150 PST=1.	C	400
PROIFF=0.	C	410
GO TO 170	C	420
160 CONTINUE	C	430
PST=(PTURN(LINK,LANC+1)-PTURN(LINK,LANC+1))/(PTURN(LINK,4)-PTURN(LINK,2))	C	440
PST=PST	C	450
IF (PST,0,0) GO TO 150	C	460
PST=PST-PTURN(N,2)*PST-(1.00-PTURN(N,4))*PST	C	470
PROIFF=PTURN(N,LANC+1)-PTURN(N,LANC+1)-PST	C	480
IF (PROIFF,0,0) PROIFF=0.	C	490
170 GO TO (170,190), LANC	C	500

TABLE B-2, cont.

188	PTON(4,LANF)=PTURN(4,4)*PDIFF/PSW	C	570
	GO TO 258	C	570
189	PTON(4,LANF)=PTURN(4,7)+PDIFF/PSW	C	570
190	CONTINUE	C	570
	OTCLAN	C	570
		C	570
191	FORMAT (5X,6F4.2)	C	570
192	FORMAT (////10X, 17HCUMULATIVE TURNING COORDINATES, 5X, 13HLANF	C	570
	1 RECF. FOR TURN, //7X, 74PT1, 5X, 74PT2, 5X, 74PT3, 5X, 74PT4, 5X,	C	570
	7 74LT1, 5X, 74LT2, 5X, 74LT3, 5X, 74LT4, 5X, 74LT5, 5X, 74LT6, 5X,	C	570
	7 74LT7, 5X, 74LT8, //7)	C	570
193	FORMAT (10X, 4HLIN, 13, 5X, 6F4.2, 5(10X, 7), 10X, 1(11X, 1X))	C	570
		C	570
	END	C	570
	EXECUTIVE INVOL	C	10
	COMMON /LINKS/ LANF(4,7), LLANF(4,7)	C	70
	COMMON /RELAT/ LINDAP(4), NOLIP(4), LINDAP(4,7), NOLANF(4)	C	70
	COMMON /GEOM/ CURR(4), CONDPR(4,4)	C	70
	COMMON /TIME/ FLTIME(8,3)	C	70
	COMMON /KMITZ/ NCAT(4), NOUTPT, ISTART, IEND, KGEN(4)	C	70
	COMMON /TURN/ NOFC(8,4), LDFC(4,7), PTURN(4,4), PTON(4,7)	C	70
	COMMON /VLEF/ VOL(4), VOLUM(4,7)	C	70
	COMMON /TM/ TIME	C	70
	COMMON /VOLN/ TVELN(4,7), TVELSIG(4,7), ACCN(7), ACCSIG(7), DECM	C	70
	IN(7), DECSIG(7), FLN(7), FLSIG(7), PTRUCK(4,7)	C	70
	REAL LLAN	C	70
	GO TO 258 L=ISTART, IEND	C	70
	REAL (5, 760) VOL(1), PTRUCK(1)	C	70
	WRITE (5, 770) L, VOL(1), PTRUCK(1)	C	70
	NOLLL=VOLANF(1)	C	70
	ILTR1	C	70
	P1=PTURN(1,7)	C	70
	GO TO 258 L=1, NOLLL	C	70
	IF (ITIME, F2, 3) FLTIME(1, LAN)=0.	C	70
	IF (FLTIME(1, LAN), 70, 10, 770, 1) FLTIME(1, LAN)=FLOST(ITIME+7)	C	70
	IF (NOLLL, 4F, 1) GO TO 110	C	70
100	VOLUM(1, 1)=VOL(1)/7600.	C	70
	GO TO 240	C	70
110	IF (LLAN(1, 1), 4F, CURR(1)) GO TO 150	C	70
	IF (LLAN(1, NOLLL), 4F, CURR(1)) GO TO 140	C	70
	GO TO (170, 170, 270), LAN	C	70
120	VOLUM(1, 1)=0.1*VOL(1)/7600.	C	70
	GO TO 240	C	70
130	VOLUM(1, 7)=0.1*-0.1*VOL(1)/7600.	C	70
	GO TO 240	C	70
140	IF (LAN, F2, NOLLL) GO TO 250	C	70
	IF (LANF(1, F2, 7) GO TO 100	C	70
	GO TO (170, 170), LAN	C	70
150	GO TO (140, 140, 170), NOLLL	C	70
160	IF (LAN, F2, 1) GO TO 250	C	70
	VOLUM(1, LAN)=VOL(1)/7600.	C	70
	GO TO 240	C	70
170	GO TO (270, 140, 140), LAN	C	70
180	IF (CURR(1, F2, LLAN(1, 1)) GO TO 200	C	70
	GO TO (250, 190, 250), LAN	C	70
190	VOLUM(1, LAN)=VOL(1)/7600.	C	70
	GO TO 240	C	70
200	GO TO (270, 210, 270), LAN	C	70

TABLE B-2, cont.

710	VOLUPIL,LANI=P1*VOLIL/7600.	7	410
720	TC P40	8	420
730	VOLLPIL,LANI=(1.-P1)*VOLIL/7600.	9	430
740	TC P40	10	440
750	VOLUPIL,LANI=0.	11	450
760	TC IVCLIPIL,LANI,C3,0,1 FLTI=FIL,LANI=10.***.	12	460
770	CONTINUE	13	470
780	RETURN	14	480
790	FORMAT (5X,FE,0,FW,3)	15	490
800	FORMAT (10X, 4MLINK,14,1X, 10M VOLUME = ,FW,0,11X, 12HPERCENT TR)	16	500
810	10KS X ,FW,2)	17	510
820	END	18	520
830	END	19	530
840	END	20	540
850	END	21	550
860	END	22	560
870	END	23	570
880	END	24	580
890	END	25	590
900	END	26	600
910	END	27	610
920	END	28	620
930	END	29	630
940	END	30	640
950	END	31	650
960	END	32	660
970	END	33	670
980	END	34	680
990	END	35	690
1000	END	36	700
1010	END	37	710
1020	END	38	720
1030	END	39	730
1040	END	40	740
1050	END	41	750
1060	END	42	760
1070	END	43	770
1080	END	44	780
1090	END	45	790
1100	END	46	800
1110	END	47	810
1120	END	48	820
1130	END	49	830
1140	END	50	840
1150	END	51	850
1160	END	52	860
1170	END	53	870
1180	END	54	880
1190	END	55	890
1200	END	56	900
1210	END	57	910
1220	END	58	920
1230	END	59	930
1240	END	60	940
1250	END	61	950
1260	END	62	960
1270	END	63	970
1280	END	64	980
1290	END	65	990
1300	END	66	1000
1310	END	67	1010
1320	END	68	1020
1330	END	69	1030
1340	END	70	1040
1350	END	71	1050
1360	END	72	1060
1370	END	73	1070
1380	END	74	1080
1390	END	75	1090
1400	END	76	1100
1410	END	77	1110
1420	END	78	1120
1430	END	79	1130
1440	END	80	1140
1450	END	81	1150
1460	END	82	1160
1470	END	83	1170
1480	END	84	1180
1490	END	85	1190
1500	END	86	1200
1510	END	87	1210
1520	END	88	1220
1530	END	89	1230
1540	END	90	1240
1550	END	91	1250
1560	END	92	1260
1570	END	93	1270
1580	END	94	1280
1590	END	95	1290
1600	END	96	1300
1610	END	97	1310
1620	END	98	1320
1630	END	99	1330
1640	END	100	1340
1650	END	101	1350
1660	END	102	1360
1670	END	103	1370
1680	END	104	1380
1690	END	105	1390
1700	END	106	1400
1710	END	107	1410
1720	END	108	1420
1730	END	109	1430
1740	END	110	1440
1750	END	111	1450
1760	END	112	1460
1770	END	113	1470
1780	END	114	1480
1790	END	115	1490
1800	END	116	1500
1810	END	117	1510
1820	END	118	1520
1830	END	119	1530
1840	END	120	1540
1850	END	121	1550
1860	END	122	1560
1870	END	123	1570
1880	END	124	1580
1890	END	125	1590
1900	END	126	1600
1910	END	127	1610
1920	END	128	1620
1930	END	129	1630
1940	END	130	1640
1950	END	131	1650
1960	END	132	1660
1970	END	133	1670
1980	END	134	1680
1990	END	135	1690
2000	END	136	1700
2010	END	137	1710
2020	END	138	1720
2030	END	139	1730
2040	END	140	1740
2050	END	141	1750
2060	END	142	1760
2070	END	143	1770
2080	END	144	1780
2090	END	145	1790
2100	END	146	1800
2110	END	147	1810
2120	END	148	1820
2130	END	149	1830
2140	END	150	1840
2150	END	151	1850
2160	END	152	1860
2170	END	153	1870
2180	END	154	1880
2190	END	155	1890
2200	END	156	1900
2210	END	157	1910
2220	END	158	1920
2230	END	159	1930
2240	END	160	1940
2250	END	161	1950
2260	END	162	1960
2270	END	163	1970
2280	END	164	1980
2290	END	165	1990
2300	END	166	2000
2310	END	167	2010
2320	END	168	2020
2330	END	169	2030
2340	END	170	2040
2350	END	171	2050
2360	END	172	2060
2370	END	173	2070
2380	END	174	2080
2390	END	175	2090
2400	END	176	2100
2410	END	177	2110
2420	END	178	2120
2430	END	179	2130
2440	END	180	2140
2450	END	181	2150
2460	END	182	2160
2470	END	183	2170
2480	END	184	2180
2490	END	185	2190
2500	END	186	2200
2510	END	187	2210
2520	END	188	2220
2530	END	189	2230
2540	END	190	2240
2550	END	191	2250
2560	END	192	2260
2570	END	193	2270
2580	END	194	2280
2590	END	195	2290
2600	END	196	2300
2610	END	197	2310
2620	END	198	2320
2630	END	199	2330
2640	END	200	2340
2650	END	201	2350
2660	END	202	2360
2670	END	203	2370
2680	END	204	2380
2690	END	205	2390
2700	END	206	2400
2710	END	207	2410
2720	END	208	2420
2730	END	209	2430
2740	END	210	2440
2750	END	211	2450
2760	END	212	2460
2770	END	213	2470
2780	END	214	2480
2790	END	215	2490
2800	END	216	2500
2810	END	217	2510
2820	END	218	2520
2830	END	219	2530
2840	END	220	2540
2850	END	221	2550
2860	END	222	2560
2870	END	223	2570
2880	END	224	2580
2890	END	225	2590
2900	END	226	2600
2910	END	227	2610
2920	END	228	2620
2930	END	229	2630
2940	END	230	2640
2950	END	231	2650
2960	END	232	2660
2970	END	233	2670
2980	END	234	2680
2990	END	235	2690
3000	END	236	2700
3010	END	237	2710
3020	END	238	2720
3030	END	239	2730
3040	END	240	2740
3050	END	241	2750
3060	END	242	2760
3070	END	243	2770
3080	END	244	2780
3090	END	245	2790
3100	END	246	2800
3110	END	247	2810
3120	END	248	2820
3130	END	249	2830
3140	END	250	2840
3150	END	251	2850
3160	END	252	2860
3170	END	253	2870
3180	END	254	2880
3190	END	255	2890
3200	END	256	2900
3210	END	257	2910
3220	END	258	2920
3230	END	259	2930
3240	END	260	2940
3250	END	261	2950
3260	END	262	2960
3270	END	263	2970
3280	END	264	2980
3290	END	265	2990
3300	END	266	3000
3310	END	267	3010
3320	END	268	3020
3330	END	269	3030
3340	END	270	3040
3350	END	271	3050
3360	END	272	3060
3370	END	273	3070
3380	END	274	3080
3390	END	275	3090
3400	END	276	3100
3410	END	277	3110
3420	END	278	3120
3430	END	279	3130
3440	END	280	3140
3450	END	281	3150
3460	END	282	3160
3470	END	283	3170
3480	END	284	3180
3490	END	285	3190
3500	END	286	3200
3510	END	287	3210
3520	END	288	3220
3530	END	289	3230
3540	END	290	3240
3550	END	291	3250
3560	END	292	3260
3570	END	293	3270
3580	END	294	3280
3590	END	295	3290
3600	END	296	3300
3610	END	297	3310
3620	END	298	3320
3630	END	299	3330
3640	END	300	3340
3650	END	301	3350
3660	END	302	3360
3670	END	303	3370
3680	END	304	3380
3690	END	305	3390
3700	END	306	3400
3710	END	307	3410
3720	END	308	3420
3730	END	309	3430
3740	END	310	3440
3750	END	311	3450
3760	END	312	3460
3770	END	313	3470
3780	END	314	3480
3790	END	315	3490
3800	END	316	3500
3810	END	317	3510
3820	END	318	3520
3830	END	319	3530
3840	END	320	3540
3850	END	321	3550
3860	END	322	3560
3870	END	323	3570
3880	END	324	3580
3890	END	325	3590
3900	END	326	3600
3910	END	327	3610
3920	END	328	3620
3930	END	329	3630
3940	END	330	3640
3950	END	331	36

TABLE B 2, cont.

CALL ROUTINE INPLIN	F	10
COMMON /A1/ IREF, IINXT, ITHX, ISIZE, IPRSTC(IK, 1), ILSTCAR(IK, 1), VEL(	F	20
15000), PTFN(1500), ATCFL(1500), ICAR(1500), NYTCAR(1500), IPRFCAR(1500)	F	30
COMMON /A2/ IHOLF(1500), LASTONE	F	40
COMMON /A3ELST/ LINCAR(IK), NOLTHK, LANCAR(IK, 1), NPLANE(84)	F	50
COMMON /A4TURN/ NREF(IK, 6), LREF(IK, 6), PTURN(IK, 1), PTP(IK, 2)	F	60
COMMON /A5TIME/ FLTIME(IK, 1)	F	70
COMMON /A6TRCF/ KTRCF(15), KOTPC(15), INTTRC, KNTRAC(15)	F	80
COMMON /A7MIXT/ NCAR(IK), ACLOT, ISTART, IFAC, KGEN(IK)	F	90
COMMON /A8/ ITIME	F	100
COMMON /A9VME/ VOL(IK), VOLUM(IK, 1)	F	110
COMMON /A10THK/ LMTN(IK, 1), LLAN(IK, 1)	F	120
COMMON /A11GE/ CURT(IK), DENDRY(IK, 6)	F	130
COMMON /A12CAPACTY/ LNCAP	F	140
COMMON /A13PARAM/ DUMAY(1), CYCLE	F	150
COMMON /A14PARAM2/ TVLTMN(IK, 2), TVFLSIG(IK, 2), ACCM(1), ACCSTG(1), OFCM	F	160
IN(1), OFCSIG(1), FLMY(1), FLSIG(1), FOTRUCK(IK)	F	170
COMMON /A15MASK/ MASK(10)	F	180
REAL LLAN	F	190
REAL NORMAL, NREFXP	F	200
1000 ITH(IK, 1, N) = ISIZE / (IK, 1), AND, MASK(1)	F	210
GO TO ISTART, IFNO	F	220
NLANF = NLANF(1)	F	230
ILT = 1	F	240
IF (LLAN(I, NLANF), IF, CURT(I)) ILT = 2	F	250
IF ILT = 1	F	260
IF (LLAN(I, 1), IF, CURT(I)) ILT = 2	F	270
GO TO K = 1, NLANF	F	280
FLT = FLOAT(ITIME)	F	290
IF (FLT - FLT(I, K), IF, FLT - CYCLE) GO TO 130	F	300
110 K = LANCAR(I, K) + 1	F	310
IF (IFEM, LE, 0, AND, LASTONE, EQ, 0) GO TO 140	F	320
IF (K, EQ, 1) GO TO 110	F	330
IL = ILSTCAR(I, K)	F	340
IF (FCSN(IL), LT, -170, 1) GO TO 130	F	350
110 CONTINUE	F	360
IF (LASTONE, EQ, 0) GO TO 120	F	370
INDEX = IHOLF(LASTONE)	F	380
LASTONE = LASTONE - 1	F	390
GO TO 130	F	400
120 INDEX = INEXT	F	410
INEXT = INEXT + 1	F	420
IREF = IREF - 1	F	430
130 CONTINUE	F	440
IPRCAR(INDEX) = ILSTCAR(I, K)	F	450
IF (LANCAR(I, K), EQ, 0) IPRFCAR(INDEX) = 0	F	460
NYTCAR(INDEX) = 0	F	470
LINDEX = IPRFCAR(INDEX)	F	480
IF (IPRCAR(INDEX), NE, 0) NYTCAR(LINDEX) = INDEX	F	490
ILSTCAR(I, K) = ININDEX	F	500
IF = ISIZE - IREF - LASTONE	F	510
ITAX = MAX0(ITS, IMAX)	F	520
IF (LANCAR(I, K), EQ, 0) IPRSTC(I, K) = ININDEX	F	530
LANCAR(I, K) = LANCAR(I, K) + 1	F	540
LINCAR(I) = LINCAR(I) + 1	F	550
KGEN(I) = KGEN(I) + 1	F	560
QK = PANK(1)	F	570
IVLTY = 1	F	580
IF (CN, LE, DOTRUCK(1)) IVLTY = 2	F	590
ITAPVF = NORMAL(TVELM(I, IVLTY), TVFLSIG(I, IVLTY))	F	600
IFL = NORMAL(FLMY(IVLTY), FLSIG(IVLTY))	F	610

TABLE B-2, cont.

IXEAC=NORMAL(ACCH(IVLTYP),ACCSIG(IVLTYP))	F 620
IDDFEC=NORMAL(DFCH*(IVLTYP),DECSIG(IVLTYP))	F 630
IACCTY=7	F 640
RN=RN*(10)	F 650
IF (IRT.EQ.1) GO TO 150	F 660
IF (ILT.EQ.1) GO TO 160	F 670
GO TO 180	F 680
140 GO TO (200,700,720), K	F 690
150 CONTINUE	F 700
IF (ILT.EQ.1) GO TO 160	F 710
GO TO 170	F 720
160 IF (INLANF.LT.2) GO TO 180	F 730
IF (K.EQ.1) GO TO 700	F 740
GO TO 720	F 750
170 IF (NLANF.LT.3) GO TO 180	F 760
IF (K.EQ.1) GO TO 700	F 770
GO TO 720	F 780
180 GO 190 L=1,6	F 790
JTURN=L	F 800
IF (RN.LF.PTURN(I,L)) GO TO 240	F 810
190 CONTINUE	F 820
GO TO 240	F 830
200 RN=RN*PTURN(I,1)	F 840
DO 210 L=1,3	F 850
JTURN=L	F 860
IF (RN.LF.PTURN(I,L)) GO TO 240	F 870
210 CONTINUE	F 880
GO TO 240	F 890
220 RN=RN*(1.-PTURN(I,3))+PTURN(I,3)	F 900
DO 230 L=4,6	F 910
JTURN=L	F 920
IF (RN.LF.PTURN(I,L)) GO TO 240	F 930
230 CONTINUE	F 940
GO TO 240	F 950
240 IF (I.EQ.LKTRCE(1)) GO TO 250	F 960
CALL JBITH (ICAR(I*JFX),F6,1,0)	F 970
GO TO 250	F 980
250 IF (LKTRCE(2).EQ.N*RC(I,JTURN)) GO TO 250	F 990
CALL JBITH (ICAR(I*JFX),F6,1,0)	F 1000
GO TO 250	F 1010
260 IF (INTTRC.EQ.NOTRC(1)) GO TO 270	F 1020
NOTRC(1)=NOTRC(1)+1	F 1030
CALL JBITH (ICAR(I*JFX),F6,1,0)	F 1040
GO TO 250	F 1050
270 NOTRC(1)=0	F 1060
CALL JBITH (ICAR(I*JFX),F6,1,1)	F 1070
280 CONTINUE	F 1080
CALL JBITH (ICAR(INDEX),F6,7,ITARVE)	F 1090
CALL JBITH (ICAR(INDEX),50,4,MAXDAC)	F 1100
CALL JBITH (ICAR(INDEX),40,4,INDEAC)	F 1110
CALL JBITH (ICAR(INDEX),40,6,IFL)	F 1120
CALL JBITH (ICAR(INDEX),0,13,ITIME)	F 1130
CALL JBITH (ICAR(INDEX),54,7,IVLTYP)	F 1140
CALL JBITH (ICAR(INDEX),36,4,IACCTY)	F 1150
CALL JBITH (ICAR(INDEX),13,1,JTURN)	F 1160
IF (K.EQ.1) GO TO 290	F 1170
JFL=JBITH(ICAR(LINDEX),40,6)	F 1180
IF (POSN(LINDEX)-FLOAT(JFL)-70.,GT.0.) GO TO 290	F 1190
POSN(LINDEX)=POSN(LINDEX)-FLOAT(JFL)-70.	F 1200
GO TO 300	F 1210
290 POSN(INDEX)=-(FLTINF(I,K)-FLOAT(ITIME))*FLOAT(ITARVE)	F 1220
300 CONTINUE	

TABLE B-2, cont

	ISTXINT(ARS(PDSN(IJDFX)))	F 1730
	CALL JMTM (ICAP(IJDFX),14,9,IST)	F 1740
	ACCEL(IJDFX)=0.	F 1750
	IF (KK,F3,1) GO TO 310	F 1760
	OTLS=PCSH(L(IJDFX)-PDSN(IJDFX)-FLCAT(FL))	F 1770
	IF (OFLS.GT.100.) GO TO 310	F 1780
	IF (VFL(LINDEX).GT.FLOAT(ITARVF)) GO TO 310	F 1790
	VFL(IJDFX)=FLOAT(ITARVF)-(100.-OFLS)/100.*FLOAT(ITARVF)+VFL(LINDEX)	F 1800
	GO TO 320	F 1810
110	VFL(IJDFX)=FLOAT(ITARVF)	F 1820
120	CONTINUE	F 1830
	IF (VOLUME(I,K),F3,1) GO TO 330	F 1840
	FLTIME(I,K)=FLTIME(I,K)+ACCEL(VOLUME(I,K))	F 1850
	IF (FLTIME(I,K).LT.FLT) GO TO 100	F 1860
130	CONTINUE	F 1870
140	CONTINUE	F 1880
	RETURN	F 1890
145	WRITE (6,140)	F 1900
	RETURN	F 1910
F		F 1920
	140 FORMAT (1/10X,14HTABLE OVERFLOW)	F 1930
C		F 1940
	END	F 1950
	SUBROUTINE UPDATE	G 1960
	COMMON /A1/ IPEN,INEXT,IMAX,ISTRT,IPRSTC(14,1),PLSTRA(14,1),VFL(	G 1970
	11500),PDSN(11500),ACCEL(11500),ICAP(11500),MYECAP(11500),IPRCC(11500)	G 1980
	COMMON /A2/ ITIME	G 1990
	COMMON /PARAM/ AG,REFCT,REFCT,CYCLE	G 2000
	COMMON /LINKS/ LANTN(84,1),LLAN(84,1)	G 2010
	COMMON /RUK/ NHOLD(84,1)	G 2020
	COMMON /XMIT/ MCA(84),NOUTPT,ISTART,ICAP,MSHIFT,1	G 2030
	COMMON /PV/ P(84,1),V(84,1)	G 2040
	COMMON /GEOM/ CURR(84),CONDOP(84,6)	G 2050
	COMMON /RELAT/ LINCAP(84),NOLINK,LANCAP(84,1),NOLANE(84)	G 2060
	COMMON /CAP/ IACC,TVFL,ITURN,FL,IACCL	G 2070
	COMMON /STARTS/ T2,MGO	G 2080
	COMMON /MSK/ MASK(70)	G 2090
	INTEGER ACCTYP	G 2100
	REAL LLAN	G 2110
C		G 2120
C	1=STOPPED	G 2130
C	2=STOPPING	G 2140
C	3=STOPPED FOR TURN	G 2150
C	4=STOPPING FOR TURN	G 2160
C	5=STOPPING FOR LANE CHANGE	G 2170
C	6=STOPPED FOR LANE CHANGE	G 2180
C		G 2190
	7=FREE BEHAVIOR	G 2200
	8=CAR FOLLOWING	G 2210
	9=IN DELAYED LEFT TURN	G 2220
	10=IN NORMAL TURN	G 2230
	11=SLOWING FOR TURN	G 2240
	12=STOPPING FOR RED SIGNAL	G 2250
		G 2260
	IRITH(I,M,N)=ISHFT(I,M).AND.MASK(N)	G 2270
	DO 100 I=1,NOLINK	G 2280
	NLAN=NOLANE(I)	G 2290
	DO 100 J=1,NLAN	G 2300
	NHOLD(I,J)=0	G 2310
100	CONTINUE	G 2320
	THOK=REFCT	G 2330
C		G 2340
C	*****	G 2350
C		G 2360

TABLE B-2, cont.

C	CHANGE LANES	G	350
C		G	360
C		G	370
C	CALL CHANGE	G	380
C		G	390
C	*****	G	400
C		G	410
C	UPDATE VEHICLES	G	420
C		G	430
C		G	440
C	DO 430 LINK=1,NOLINK	G	450
	IF (LTNCAR(LINK),F0,0) GO TO 430	G	460
	NOLL=NOLANF(LINK)	G	470
	DO 430 LANE=1,NOLL	G	480
	IF (LANCAP(LINK,LANE),F0,0) GO TO 430	G	490
	CALL SIGNAL (ISTG,LINK,LANE,ITIME)	G	500
	LANNA=LANCAP(LINK,LANE)	G	510
	NUM=IFIRSTCP(LINK,LANE)	G	520
110	MSD=0	G	530
	IACCL=IACC	G	540
	IACC=IPRITH(ICAR(NUM),36,4)	G	550
	TVFL=FLOAT(IPRITH(ICAP(NUM),2F,7))	G	560
	ITURN=IPRITH(ICAR(NUM),33,1)	G	570
	IF (LINK,GT,MOUPT) GO TO 120	G	580
	IF (NUM,NF,IFIRSTCP(LINK,LANE)) GO TO 120	G	590
	ACCFI(NUM)=PDEFK*(TVFL-VFL(NUM))	G	600
	ACCTYP=7	G	610
	GO TO 330	G	620
120	CONTINUE	G	630
	ACCPB=ACCFI(NUM)	G	640
	ICHAGH=0	G	650
	IF (IACC,F0,13) GO TO 330	G	660
	IF (IACC,NF,6,AND,IACC,NF,6,OP,LANE,F0,LANTH(LINK,ITURN)) GO TO 13	G	670
130	GO TO (140,160,170,180,190,200,210,220,230,240,250,260,270,150), I	G	680
	ISTG	G	690
140	CALL PFOAPP (LINK,LANE,NLM,ACCTYP)	G	700
	GO TO 330	G	710
150	CALL AMBER (IGO,LINK,LANE,NUM,ACCTYP)	G	720
	IF (IGO,F0,0) GO TO 140	G	730
160	GO TO (280,280,300,300,290,290), ITURN	G	740
170	IF (ITURN,NF,3,AND,ITURN,NF,4) GO TO 140	G	750
	GO TO 300	G	760
180	IF (ITURN,NF,6) GO TO 140	G	770
	GO TO 290	G	780
190	IF (ITURN,NF,1) GO TO 140	G	790
	GO TO 280	G	800
200	IF (ITURN,NF,5) GO TO 140	G	810
	GO TO 290	G	820
210	IF (ITURN,NF,2) GO TO 140	G	830
	GO TO 280	G	840
220	IF (ITURN,NF,3,AND,ITURN,NF,4,AND,ITURN,NF,6) GO TO 140	G	850
	IF (ITURN,F0,6) GO TO 290	G	860
	GO TO 300	G	870
230	IF (ITURN,NF,3,AND,ITURN,NF,4,AND,ITURN,NF,1) GO TO 140	G	880
	IF (ITURN,F0,1) GO TO 280	G	890
	GO TO 300	G	900
		G	910
		G	920
		G	930
		G	940
		G	950

TABLE B-2, cont.

245	IF (ITURN.NF.3.AND.ITURN.NF.4.AND.ITURN.NF.5) GO TO 140	C 960
	IF (ITURN.FQ.5) GO TO 290	C 970
	GO TO 300	C 980
255	IF (ITURN.NF.3.AND.ITURN.NF.4.AND.ITURN.NF.7) GO TO 140	C 990
	IF (ITURN.FQ.7) GO TO 280	C 1000
	GO TO 300	C 1010
260	IF (ITURN.LF.4) GO TO 140	C 1020
	GO TO 230	C 1030
270	IF (ITURN.CE.3) GO TO 140	C 1040
280	CONTINUE	C 1050
	CALL RIGHT (LINK,LANF,NUM,ACCTYP)	C 1060
	GO TO 310	C 1070
290	CONTINUE	C 1080
	CALL LEFT (LINK,LANF,NUM,ITIME,ACCTYP)	C 1090
	GO TO 310	C 1100
300	CONTINUE	C 1110
	IF (NUM.NE.IFRCTCR(LINK,LANF)) GO TO 310	C 1120
	IF (POSN(NUM).LT.CIRN(LINK)-100.) GO TO 310	C 1130
	CALL CONST (LINK,LANF,NUM,ICON,0)	C 1140
	IF (ICON.EQ.1) GO TO 370	C 1150
310	CALL GENPR (LINK,LANF,NUM,ACCTYP)	C 1160
	GO TO 330	C 1170
320	IF (VFL(NUM).LT.0.) VFL(NUM)=0.	C 1180
	IF (POSN(NUM).GE.CIRN(LINK)) GO TO 310	C 1190
	XIX=CURR(LINK)-POSI(NUM)	C 1200
	IF (XIX.EQ.0.) XIX=.5	C 1210
	ACCF(NUM)=-(VEL(NUM)*VEL(NUM))/XIX	C 1220
	ACCTYP=2	C 1230
330	CONTINUE	C 1240
	IF (ICHNGH.NF.1) GO TO 340	C 1250
	IF (ACCF(NUM).LT.ACPR) GO TO 340	C 1260
	ACCEL(NUM)=ACPR	C 1270
	ACCTYP=IACC	C 1280
340	CONTINUE	C 1290
	IF (NUM.EQ.IFRCTCR(LINK,LANF).AND.IACC.FQ.1.AND.ACCTYP.NF.1) REACT	C 1300
	1=.35	C 1310
	P(LINK,LANF)=POSN(NUM)	C 1320
	V(LINK,LANF)=VFL(NUM)	C 1330
	IF (MGO.EQ.0) GO TO 350	C 1340
	VEL(NUM)=ACCF(NUM)*(CYCLE-T2)	C 1350
	IF (VEL(NUM).LT.0.) VEL(NUM)=0.	C 1360
	TT3=Cycle-T2	C 1370
	POSN(NUM)=POSN(NUM)+.5*ACCF(NUM)*TT3*TT3	C 1380
	MGO=0	C 1390
	GO TO 400	C 1400
350	V2=VFL(NUM)+ACPR*REACT	C 1410
	VEL(NUM)=V2+ACCF(NUM)*(CYCLE-REACT)	C 1420
	IF (VEL(NUM).GE.0.) GO TO 380	C 1430
	IF (V2.GE.0.) GO TO 360	C 1440
	T=-V(LINK,LANF)/ACPR	C 1450
	ARST=ABS(T)	C 1460
	POSN(NUM)=POSN(NUM)+V(LINK,LANF)*T+.5*ACPR*ARST*ARST	C 1470
	GO TO 370	C 1480
360	T=-V2/ACCEL(NUM)	C 1490
	ARST=ABS(T)	C 1500
	POSN(NUM)=POSN(NUM)+V2*T+.5*ACPR*ARST*ARST	C 1510
370	ACCEL(NUM)=0.	C 1520
	VEL(NUM)=0.	C 1530
	ACCTYP=1	C 1540
	IF (ICHNGH.FQ.1) ACCTYP=6	C 1550
	GO TO 400	C 1560

TABLE B-2, cont.

140	POSN(NUM)=P(LINK,LANF)+V(LINK,LANF)*REACT+.5*ACCDP*REACT*REACT+V2*	C	1570		
	1(CYCLE-REACT)+.5*ACFL(NUM)*(CYCLE-REACT)*(CYCLE-REACT)	C	1580		
	IF (POSN(NUM).LT.P(LINK,LANF)) POSN(NUM)=P(LINK,LANF)	C	1590		
	IF (NUM.FQ.IFSTC(P(LINK,LANF)) GO TO 300	C	1600		
	N=IPREFCAR(NUM)	C	1610		
	IF (POSN(NUM).GT.P(SN(N)) POSN(NUM)=POSN(N)-70.	C	1620		
145	CONTINUE	C	1630		
	IF (VFL(NUM).LT.0.1) VFL(NUM)=0.	C	1640		
148	CONTINUE	C	1650		
	IF (ACCTYP.FQ.0) GO TO 410	C	1660		
	CALL JRTM (ICAR(NUM),TA,4,ACCTYP)	C	1670		
150	CONTINUE	C	1680		
	IF (ACCTYP.FQ.0) ACCTYP=IACC	C	1690		
	REACT=THOR	C	1700		
	CALL ROK (LINK,LANF,NUM,ITIME)	C	1710		
	IF (NUM.FQ.ILSTCAR(LINK,LANF)) GO TO 420	C	1720		
	VIM=NXTCAR(NUM)	C	1730		
	IF (NUM.NF.0) GO TO 110	C	1740		
170	CONTINUE	C	1750		
	CALL QUEUE (LINK,LANF)	C	1760		
171	CONTINUE	C	1770		
	CALL CUTPT (ITIME)	C	1780		
	RETURN	C	1790		
C		C	1800		
	END	C	1810		
	SUBROUTINE SIGNAL (ITIG,LINK,LANF,ITIME)	H	10		
	COMMON /SIG/ LCYCLE(14),NOPHAS(14,3),NPHASE(14,3),LPHASE(14,3,5)	H	20		
	1,LOFFST(14,3)	H	30		
	INTEGER OFFPT	H	40		
C		H	50		
C	1= RED	5= GAL	11= GAS+GADR	H	60
C	2= GREEN	7= GADR	12= GAL+GADR	H	70
C	3= GR	8= GAS+GAL	13= GAD+GADR	H	80
C	4= GAL	9= GAS+GAR	14= ABED	H	90
C	5= GAR	10= GAS+GADR		H	100
C				H	110
	IF (ITIME.LT.LOFFST(LINK,LANF)) GO TO 100	H	120		
	NCYC=(ITIME-LOFFST(LINK,LANF))/LCYCLE(LINK)	H	130		
	OFFPT=NCYC*LCYCLE(LINK)+LOFFST(LINK,LANF)	H	140		
	GO TO 110	H	150		
100	NCYC=0	H	160		
	OFFPT=LCYCLE(LINK)+LOFFST(LINK,LANF)	H	170		
110	CONTINUE	H	180		
	NOPP=NOPHAS(LINK,LANF)	H	190		
	DO 170 I=1,NOPP	H	200		
	IT=OFFPT+LPHASE(LINK,LANF,I)	H	210		
	IHOLD=I	H	220		
	IF (ITIME.LE.IT) GO TO 130	H	230		
	OFFPT=IT	H	240		
170	CONTINUE	H	250		
130	CONTINUE	H	260		
	ISIG=NPHASE(LINK,LANF,IHOLD)	H	270		
	RETURN	H	280		
C		H	290		
	END	H	300		

TABLE B-2, cont.

	SUBROUTINE AMREP (IG0, LINK, LANF, NUM, ACCTYP)	I	10
	COMMON /A1/ IREF, ITEXT, IMAX, ISIZE, IFRSTCAR(84, 1), TLSTCAR(84, 1), VFL(	I	20
	11500), POSN(1500), ACCEL(1500), ICAR(1500), NXTCAR(1100), IPRFCAR(1500)	I	30
	COMMON /GEOM/ CURR(84), OPNDRY(84, 6)	I	40
	COMMON /MSK/ MASK(70)	I	50
	COMMON /CAR/ IACC, TVFL, ITURN, FL, IACCL	I	60
	COMMON /PRSTOP/ PRSTOP(17)	I	70
	REAL NORMAL	I	80
	INTFCR ACCTYP	I	90
	DATA (PRSTOP(I), I=1, 10) / .908, .927, .954, .977, .997, .999, .999, .999, .999, .999,	I	100
	1.999/	I	110
	IF (IACC.NF.2.AND.IACC.NF.17) GO TO 100	I	120
	IF (IACCL(NUM).CF.1.1) GO TO 100	I	130
	XFINAL=-VFL(NUM)*VEL(NUM)/(2.*ACCEL(NUM)+ACCEL(NUM))	I	140
	IF (XFINAL.LF.CURR(LINK)) GO TO 140	I	150
100	CONTINUE	I	160
	IF (IACC.NF.1) GO TO 110	I	170
	IF (POSN(NUM).GT.CURR(LINK)) GO TO 170	I	180
	GO TO 140	I	190
110	CONTINUE	I	200
	IF (NUM.GE.IFRSTCAR(LINK, LANF)) GO TO 120	I	210
	N=IPRFCAR(NUM)	I	220
	IF (ACCTYP.LF.6.70.ACCTYP.FN.12) GO TO 140	I	230
120	CONTINUE	I	240
	XFINAL=CURR(LINK)	I	250
	V2=VFL(NUM)+ACCEL(NUM)*REACT	I	260
	IF (V2.LT.0.) V2=0.	I	270
	P2=POSN(NUM)+VFL(NUM)*REACT+.5*ACCEL(NUM)*REACT*REACT	I	280
	XX=XFINAL-P2	I	290
	IF (XX.LF.0.) GO TO 130	I	300
	ACC=-(V2*V2)/(2.*XX)	I	310
	FN=RNANF(0)	I	320
	KACC=ABS(ACC)/2.	I	330
	IF (KACC.LT.1) GO TO 140	I	340
	IF (KACC.GT.10) GO TO 170	I	350
	IF (RN.GT.PRSTOP(KACC)) GO TO 140	I	360
130	CONTINUE	I	370
	ISO=1	I	380
C		I	390
C	ISO = 1 MEANS GO	I	400
C		I	410
	RETURN	I	420
140	CONTINUE	I	430
	ISO=0	I	440
C		I	450
C	ISO = 0 MEANS STOP	I	460
C		I	470
	RETURN	I	480
C		I	490
	END	I	500
	SUBROUTINE CHANGE	J	10
	COMMON /A1/ IREF, ITEXT, IMAX, ISIZE, IFRSTCAR(84, 1), TLSTCAR(84, 1), VFL(	J	20
	11500), POSN(1500), ACCEL(1500), ICAR(1500), NXTCAR(1100), IPRFCAR(1500)	J	30
	COMMON /GEOM/ CURR(84), OPNDRY(84, 6)	J	40
	COMMON /LINKS/ LANTN(84, 6), LLAN(84, 1)	J	50
	COMMON /RELAT/ LINCAP(84), NOLINK, LANCAP(84, 1), NOLANE(84)	J	60
	COMMON /PARAM/ AD, FREEF, REACT, CYCLE	J	70
	COMMON /CAPACTY/ LUCAP	J	80
	COMMON /TM/ ITIME	J	90

TABLE B-2, cont.

	IF (KACC.NF.6) GO TO 240	J 710
	ACCFI(INDEX)=0.	J 720
	GO TO (210,220,230), LANE	J 730
210	LADJ=2	J 740
	GO TO 230	J 750
220	LADJ=1	J 760
	GO TO 230	J 770
230	IF (LANE.LT.LADJ) GO TO 270	J 780
	INDADJ=IFRSTCR(LINK,LADJ)	J 790
240	IF (INDADJ.EQ.0) GO TO 270	J 800
	IF (POSN(INDEX).LT.POSN(INDADJ)-FLOAT(1BITM(ICAR(INDADJ),40,6))-10	J 810
	1.) GO TO 250	J 820
	GO TO 260	J 830
250	CONTINUE	J 840
	INDADJ=NXTCAR(INDADJ)	J 850
	GO TO 240	J 860
260	IF (1BITM(ICAR(INDADJ),36,4).NF.6) GO TO 270	J 870
	IF (KACC.NE.13) CALL JBITH (ICAR(INDEX),36,4,13)	J 880
C	IACC = 1 ³ IS TO BYPASS THE NORMAL EFFECT OF A	J 890
C	RED SIGNAL IN ADJUSTMENT FOR LANE CHANGE	J 900
	GO TO 590	J 910
270	IF (KACC.NF.6) CALL JBITH (ICAR(INDEX),36,4,6)	J 920
	GO TO 590	J 930
280	CONTINUE	J 940
	IF (KACC.EQ.5) GO TO 290	J 950
	ACCFI(INDEX)=-FLOAT(1BITM(ICAR(INDEX),46,4))*2.0	J 960
290	PF=FCSN(INDEX)-VEL(INDEX)*VEL(INDEX)/(2.*ACCFI(INDEX))	J 970
	GO TO (300,310,320), LANE	J 980
300	LADJ=2	J 990
	GO TO 330	J 1000
310	LADJ=1	J 1010
	GO TO 330	J 1020
320	LADJ=2	J 1030
330	INDADJ=IFRSTCR(LINK,LADJ)	J 1040
	IF (INDADJ.EQ.0) GO TO 400	J 1050
340	CONTINUE	J 1060
	IF (PF.GT.POSN(INDADJ)+EL+10.) GO TO 350	J 1070
	IF (PF.LT.POSN(INDADJ)-40.) GO TO 360	J 1080
	ELADJ=FLOAT(1BITM(ICAR(INDADJ),40,6))	J 1090
	IF (PF.LT.POSN(INDADJ)-ELADJ-10.) GO TO 360	J 1100
	IF (1BITM(ICAR(INDADJ),36,4).NF.6) GO TO 400	J 1110
	X=POSN(INDADJ)-ELADJ-10.-POSN(INDEX)	J 1120
	IF (X.LE.0.) GO TO 370	J 1130
	ACCFI(INDEX)=-VFL(INDEX)*VFL(INDEX)/(2.*X)	J 1140
	PF=FCSN(INDADJ)-ELADJ-10.	J 1150
350	IF (LANE.LT.LADJ) GO TO 400	J 1160
	IF (ACCEL(INDADJ).GE.0.) GO TO 360	J 1170
	PFADJ=POSN(INDADJ)-VFL(INDADJ)*VEL(INDADJ)/(2.*ACCEL(INDADJ))	J 1180
	IF (PF-FL-10..GT.PFADJ) GO TO 400	J 1190
	PFLZN=PFADJ-10.-FLOAT(1BITM(ICAR(INDADJ),40,6))	J 1200
	IF (PF.LT.PFLZN) GO TO 360	J 1210
	IF (POSN(INDEX).GE.PFLZN) GO TO 400	J 1220
	ACCEL(INDEX)=-VEL(INDEX)*VFL(INDEX)/(2.*(PFLZN-POSN(INDEX)))	J 1230
360	CONTINUE	J 1240
	INDADJ=NXTCAR(INDADJ)	J 1250
	IF (INDADJ.NE.0) GO TO 340	J 1260
370	IF (INDADJ.EQ.IFRSTCR(LINK,LADJ)) GO TO 390	J 1270
	LINDADJ=IPRECAR(INDADJ)	J 1280
	IF (POSN(LINDADJ)-POSN(INDADJ).LT.EL+10.+25.) GO TO 400	J 1290
380	IF (KACC.NF.7) CALL JBITH (ICAR(INDEX),36,4,7)	J 1300
	GO TO 590	J 1310

TABLE B-2, cont.

390	IF (CUR9(LINK)-POS4(INDADJ),GE,FL+10.) GO TO 390	J 1320
400	CONTINUE	J 1330
	KACC=5	J 1340
	IF (IACC.NE,KACC) CALL JBITH (ICAR(INDEX),36,4,KACC)	J 1350
	GO TO 590	J 1360
C	CHANGE LANFS	J 1370
410	IF (N.EQ,IFRSTCR(LINK,LANNEW)) GO TO 440	J 1380
	IF (LANCAR(LINK,LANNEW).EQ,0) GO TO 440	J 1390
	NL=IPRECAR(INDADJ)	J 1400
	IF (N.EQ,0) NL=ILSTCAR(LINK,LANNEW)	J 1410
	IF (ACCEL(NL).LT,0.) GO TO 420	J 1420
	T=CYCLE+REACT	J 1430
	P2L=FOSN(NL)+VEL(NL)*T+.5*ACCEL(NL)*T*T	J 1440
	P2N=FOSN(INDEX)+VEL(INDEX)*REACT+.5*ACCEL(INDEX)*REACT*REACT+(VEL(INDEX)+ACCEL(INDEX)*REACT)*CYCLE+.5*(-7.)*CYCLE*CYCLE	J 1450
	GO TO 430	J 1460
420	P2L=FOSN(NL)-VEL(NL)*VEL(NL)/(2.*ACCEL(NL))	J 1470
	P2N=FOSN(INDEX)*VEL(INDEX)*VEL(INDEX)	J 1480
430	CONTINUE	J 1490
	IF (F2N.GT,P2L-FLOAT(JBITH(ICAR(NL),40,6))) GO TO 190	J 1500
440	IOL=IPRECAR(INDEX)	J 1510
	IOF=NXTCAR(INDEX)	J 1520
	LANCAR(LINK,LANNEW)=LANCAR(LINK,LANNEW)+1	J 1530
	LANCAR(LINK,LANF)=LANCAR(LINK,LANF)-1	J 1540
	LA1=LANCAR(LINK,LANNEW)	J 1550
	LA=LA1	J 1560
	IF (N.NE,0) GO TO 460	J 1570
	IL=ILSTCAR(LINK,LANNEW)	J 1580
	IF (IL.EQ,0) GO TO 450	J 1590
	NXTCAR(IL)=INDEX	J 1600
450	IPRECAR(INDEX)=IL	J 1610
	NXTCAR(INDEX)=0	J 1620
	ILSTCAR(LINK,LANNEW)=INDEX	J 1630
	IF (LANCAR(LINK,LANNEW).EQ,1) IFRSTCR(LINK,LANNEW)=INDEX	J 1640
	N=LA1	J 1650
	GO TO 490	J 1660
460	CONTINUE	J 1670
	IL=IPRECAR(N)	J 1680
	IF=N	J 1690
	IF (IL.EQ,0) GO TO 470	J 1700
	NXTCAR(IL)=INDEX	J 1710
470	NXTCAR(INDEX)=IF	J 1720
	IPRECAR(INDEX)=IL	J 1730
	IPRECAR(IF)=INDEX	J 1740
	IF (IPRECAR(INDEX).EQ,0) IFRSTCR(LINK,LANNEW)=INDEX	J 1750
480	CONTINUE	J 1760
	IF (IFRSTCR(LINK,LANF).EQ,INDEX) IFRSTCR(LINK,LANF)=IOF	J 1770
	IF (IFRSTCR(LINK,LANF).EQ,IOF) IPRECAR(IOF)=0	J 1780
	IF (ILSTCAR(LINK,LANF).EQ,INDEX) ILSTCAR(LINK,LANF)=IOL	J 1790
	IF (ILSTCAR(LINK,LANF).EQ,IOL) NXTCAR(IOL)=0	J 1800
	IF (JBITH(ICAR(INDEX),36,4).EQ,6) GO TO 490	J 1810
	CALL JBITH (ICAR(INDEX),36,4,8)	J 1820
	GO TO 500	J 1830
490	CALL JBITH (ICAR(INDEX),36,4,13)	J 1840
500	CONTINUE	J 1850
	IF (IOL.EQ,0) GO TO 510	J 1860
	NXTCAR(IOL)=IOF	J 1870
510	IF (IOF.EQ,0) GO TO 520	J 1880
	IPRECAR(IOF)=IOL	J 1890
520	CONTINUE	J 1900
530	IF (IL.EQ,0) GO TO 540	J 1910
		J 1920

TABLE B-2, cont.

	IF=NXTCAR(IL)	J 1978
	IF (IF.EQ.0) GO TO 540	J 1940
	PX=PCSN(IL)-FLCAT(IBITH(ICAR(IL),40,6))-5.	J 1950
	IF (PX-POSN(IF).LE.0.) POSN(IF)=PX	J 1960
	IL=NXTCAR(IL)	J 1970
	GO TO 530	J 1980
540	INDEX=IOF	J 1990
	IF (IOF.NE.0) GO TO 100	J 2000
	GO TO 600	J 2010
550	CONTINUE	J 2020
	DT=ENDLAN-POSN(INDEX)	J 2030
	IF (DT.LT.100.) GO TO 560	J 2040
	CALL JBITH (ICAR(INDEX),36,4,13)	J 2050
	GO TO 590	J 2060
560	CONTINUE	J 2070
	IACC=IBITH(ICAR(INDEX),36,4)	J 2080
	IF (CURB(LINK)-POSN(INDEX).GT.300..AND..LLAN(LINK,LANNEW).F2..LLAN(LINK,LANE)) GO TO 590	J 2090
	LA=LANCAR(LINK,LANNEW)	J 2100
	IF (LA.EQ.0) GO TO 570	J 2110
	ILST=ILSTCAR(LINK,LANNEW)	J 2120
	DT=PCSN(ILST)-POSN(INDEX)	J 2130
	IF (DT.GE.100.) GO TO 570	J 2140
	IF (DT.LE.0.) DT=1.	J 2150
	IF ((VEL(ILST)*VEL(ILST)-VEL(INDEX)*VEL(INDEX))/2.*DT).GE.-12.) G	J 2160
	GO TO 570	J 2170
	ACCEL(INDEX)=(VEL(ILST)-VEL(INDEX))/DT*AD	J 2180
	IF (IACC.EQ.5.OR.IACC.EQ.6) GO TO 580	J 2190
	CALL JBITH (ICAR(INDEX),36,4,5)	J 2200
	GO TO 590	J 2210
570	CALL JBITH (ICAR(INDEX),36,4,13)	J 2220
	GO TO 590	J 2230
580	IF (IACC.EQ.6.AND.ACCEL(INDEX).GT.0.) CALL JBITH (ICAR(INDEX),36,4	J 2240
	1,5)	J 2250
590	CONTINUE	J 2260
	INDEX=NXTCAR(INDEX)	J 2270
	IF (INDEX.GT.0) GO TO 100	J 2280
600	CONTINUE	J 2290
610	CONTINUE	J 2300
	RETURN	J 2310
C	END	J 2320
		J 2330
		J 2340

SUBROUTINE GENBHR (LINK,LANE,NUM,IFLAG)	K 10
COMMON /A1/ IREM,INEXT,IPAX,ISIZE,IFRSTCR(84,3),ILSTCAR(84,3),VEL(	K 20
11500),POSN(1500),ACCEL(1500),ICAR(1500),NXTCAR(1500),IPRECAR(1500)	K 30
COMMON /PARAM/ AD,FRECK,REACT,CYCLE	K 40
COMMON /RELAT/ LINCAR(84),NOLINK,LANCAR(84,3),NOLANE(84)	K 50
COMMON /PV/ P(84,3),V(84,3)	K 60
COMMON /GEOM/ CURB(84),DBNDRY(84,6)	K 70
COMMON /TURN/ NREC(84,6),LREC(84,6),PTRN(84,6),PTR(84,2)	K 80
COMMON /CAR/ IACTYP,TVEL,ITURN,EL,JACTYP	K 90
COMMON /KUNTZ/ NCAR(84),NOUTPT,ISTART,IEND,KGEN(84)	K 100
COMMON /STARTS/ T2,HGO	K 110
COMMON /MSK/ MASK(70)	K 120
REAL NORMAL	K 130
IRITH(IX,M,N)=ISHFT(IX,M).AND.MASK(N)	K 140
LNUM=IPRECAR(NUM)	K 150
MAXDAC=IRITH(ICAR(LNUM),50,4)	K 160

TABLE B 2, cont.

LACTYP=IFLAG	K	778
IFLAG=0	K	780
MSD=0	K	780
ACCL=ACCELNUM	K	780
IF (NUM.FQ.IFSTORLINK,LANF) GO TO 100	K	780
DTST=RELINK,LANF)-POSNUM	K	770
IF (DTST.LE.0.) DIST=.001	K	770
IF (DIST.GT.700.) GO TO 100	K	767
IF (JACTYP.FQ.1,AN).LACTYP,NF.1) GO TO 100	K	768
GO TO 110	K	768
100 IF (LACTYP.FQ.0) GO TO 110	K	770
IF (ACCEL(LNUM).FQ.0.) GO TO 110	K	768
IF (IACTYP,NF.1,AN).IACTYP,NF.1) GO TO 110	K	760
T=Cycle-VEL(LNUM)/ACCEL(LNUM)+1.	K	760
IF (T.GT.Cycle) GO TO 110	K	760
V=VELINK,LANF)+ACCEL(LNUM)	K	760
P=RELINK,LANF)+.5*ACCEL(LNUM)	K	770
ACCEL=40*(V2-VEL(NUM))/((T-POSNUM))	K	760
MSD=1	K	760
GO TO 170	K	760
110 CONTINUE	K	770
ACCEL=(VELINK,LANF)-VEL(NUM))/40/PICT	K	760
IF (ACCEL.LE.0.) GO TO 140	K	760
120 CONTINUE	K	760
ACCF=REFX*(TVEL-VEL(NUM))	K	760
IF (ACCF.LE.ACCEL) GO TO 100	K	760
ACCEL(NUM)=ACCF	K	760
IF (ACCEL.LE.0.,.40,VEL(NUM).LE.0.) GO TO 140	K	760
IF (ACCEL.GT.FLOAT(MAXACC)) ACCEL(NUM)=FLOAT(MAXACC)	K	760
130 IF (IACTYP.FQ.0) RETURN	K	760
IFLAG=0	K	770
GO TO 210	K	760
140 CONTINUE	K	760
IF (JACTYP.GT.0,AN).IACTYP,LT.12) GO TO 170	K	760
IF (IACTYP.FQ.1) GO TO 140	K	760
IF (LACTYP.GT.0,AN).LACTYP,LT.12) GO TO 170	K	770
IF (ACCEL(LNUM).FQ.0.) GO TO 140	K	760
IF (VEL(LNUM).LT.0.) VEL(LNUM)=0.	K	760
SIF=-(VEL(LNUM)*VEL(LNUM))/12.*ACCEL(LNUM)+POS(LNUM)	K	760
GO TO 160	K	760
150 SIF=VEL(LNUM)*CYCLE+POS(LNUM)	K	770
160 FL=FLCAT(IRITH(ICAP(LNUM),40,0))	K	760
SIF=SIF-FL	K	760
V2=VEL(LNUM)+ACCL*REACT	K	760
IF (V2.LE.0.) V2=.001	K	760
S2=FCN(NUM)+VEL(LNUM)*REACT+.5*ACCEL(LNUM)*REACT*REACT	K	760
XT=S2F-S2	K	760
IF (XI.LE.0.) XI=1.	K	760
ACCEL(NUM)=-(V2*V2)/(12.*XI)	K	760
IF (ACCEL(NUM).LT.-FLOAT(IRITH(ICAP(NUM),40,41)) GO TO 170	K	760
S2=S2+CYCLE*V2	K	760
XT=S2F-S2	K	760
IF (XI.LT.0.) GO TO 170	K	760
IF (XI.FQ.0.) XI=1	K	760
ACC=-(V2*V2)/(12.*XI)	K	760
IF (ACC.LT.-10.) GO TO 170	K	760
ACCEL(NUM)=0.	K	760
IF (IACTYP.FQ.0) RETURN	K	760
IFLAG=0	K	760
GO TO 210	K	760
170 IF (IACTYP.FQ.2) RETURN	K	770

TABLE B-2, cont.

	IFLAG=2	K	780
	GO TO 210	K	790
180	IF (IACCTYP.NE.1) IFLAG=1	K	800
	ACCFCL(NUM)=0.	K	810
	GO TO 210	K	820
190	CONTINUE	K	830
	ACCFRE=FREEK*(TVEL-VEL(NUM))	K	840
200	ACCEL(NUM)=FLOAT(MAXDAC)	K	850
	IF (ACCFRE.LE.ACCEL(NUM)) ACCEL(NUM)=ACCFRE	K	860
	IF (IACCTYP.NE.7) IFLAG=7	K	870
	IF (NUM.EQ.IFRTCP(LINK)) GO TO 210	K	880
	IF (LINK.LE.NOUTPT) GO TO 210	K	890
	POSNEX=ACCEL(NUM)*CYCLE*CYCLE/2.+VEL(NUM)*CYCLE+POSN(NUM)	K	900
	IF (POSNEX.LE.DPNDRY(LINK,ITURN)) GO TO 210	K	910
	IT=NPFC(LINK,ITURN)	K	920
	JJ=LPEC(LINK,ITURN)	K	930
	IF (II.LE.0) GO TO 210	K	940
	KK=ILSTCAR(II,JJ)	K	950
	IF (KK.EQ.0) GO TO 210	K	960
	IF (FCSN(KK)+DPNDRY(LINK,ITURN)-POSN(NUM).GT.700.) GO TO 210	K	970
	IF (II.GT.LINK) GO TO 220	K	980
	XI=DPNDRY(LINK,ITURN)-POSN(NUM)+P(II,JJ)	K	990
	IF (XI.LE.0.) XI=1.	K	1000
	ACCFOL=A0*(V(II,JJ)-VEL(NUM))/XI	K	1010
210	IF (ACCEL(NUM).LE.ACCFOL) GO TO 210	K	1020
	ACCEL(NUM)=ACCFOL	K	1030
	GO TO 110	K	1040
220	XI=DPNDRY(LINK,ITURN)-POSN(NUM)+FCSN(KK)	K	1050
	IF (XI.LE.0.) XI=1.	K	1060
	ACCFOL=A0*(VEL(KK)-VEL(NUM))/XI	K	1070
	GO TO 210	K	1080
230	CONTINUE	K	1090
	RETURN	K	1100
C		K	1110
	END	K	1120
	SUBROUTINE REDAPP (LINK,LANE,NUM,ACCTYP)	L	10
	COMMON /A1/ IREN,INEXT,IPAX,ISI7E,IFRTCP(84,3),TLSTCAR(84,3),VEL(	L	20
	11500),POSN(1500),ACCEL(1500),ICAR(1500),NXTCAR(1500),IPRECAR(1500)	L	30
	COMMON /PARAM/ A0,FREEK,PEACT,CYCLE	L	40
	COMMON /GEOM/ CURB(84),DPNDRY(84,6)	L	50
	COMMON /PV/ P(84,3),V(84,3)	L	60
	COMMON /TM/ ITIME	L	70
	COMMON /CAR/ IACC,TVEL,ITURN,EL,IACCL	L	80
	COMMON /MSK/ MASK(20)	L	90
	INTEGER ACCTYP	L	100
	REAL NORMAL	L	110
	I9ITP(IX,M,N)=ISHFTR(IX,P).AND.MASK(N)	L	120
	LNUP=IPRECAR(NUM)	L	130
	N=LNUP	L	140
	IFLAG=0	L	150
	LACC=ACCTYP	L	160
	ACCTYP=0	L	170
	ACC1=ACCEL(NUM)	L	180
	IF (POSN(NUM).GE.CURB(LINK)-400.) GO TO 140	L	190
100	CALL GENRHR (LINK,LANE,NUM,LACC)	L	200
	ACCTYP=LACC	L	210
	GO TO 480	L	220
110	GO TO (170,120,100,100,130,130), ITURN	L	230

TABLE B-2, cont

170	CALL RIGHT (LINK,LANE,NUM,LACC)	L	248
	ACCTYP=LACC	L	250
	RETURN	L	251
171	CALL LEFT (LINK,LANE,NUM,ITIME,LACC)	L	270
	ACCTYP=LACC	L	280
	RETURN	L	290
140	CONTINUE	L	300
	IF (AUM.NF.IFPOSTRELINK,LANE) GO TO 248	L	310
	IF (IACC.NF.1) GO TO 190	L	320
	K=CURRELINK+1	L	330
	IF (POS(NUM).LT.K) GO TO 140	L	340
	ACCEL(NUM)=0.	L	350
	VFL(NUM)=0.	L	360
	IF (IACC.FQ.1) RETURN	L	370
	ACCTYP=1	L	380
	RETURN	L	390
150	CONTINUE	L	400
	ACCF5=FLOAT(I8ITH(ICAR(NUM),50,4))	L	410
	ACCEL(NUM)=FFFF*(VFL-VFL(NUM))	L	420
160	CONTINUE	L	430
	ACC=ACCF5/2.	L	440
	IF (ACCEL(NUM).LE.ACC) GO TO 170	L	450
	ACCEL(NUM)=ACC	L	460
170	CONTINUE	L	470
	IF (IACC.EQ.7) GO TO 140	L	480
	ACCTYP=7	L	490
140	CONTINUE	L	500
	RETURN	L	510
190	CONTINUE	L	520
	IF (IACC.NF.12) GO TO 200	L	530
	IF (VFL(NUM).GT.0.) GO TO 220	L	540
	VFL(NUM)=0.	L	550
	IF (IACC.FQ.1) RETURN	L	560
	ACCTYP=1	L	570
	RETURN	L	580
200	CONTINUE	L	590
	IF (PCSN(NUM).GT.CURRELINK+1.) GO TO 110	L	600
	IF (CURRELINK-POS(NUM).LT.100.) GO TO 210	L	610
	CALL GENRHP (LINK,LANE,NUM,ACCTYP)	L	620
	RETURN	L	630
210	CONTINUE	L	640
	V2=VFL(NUM)+ACCI*REACT	L	650
	SF=CURRELINK+1.	L	660
	S2=PCSN(NUM)+VFL(NUM)*REACT+.5*ACCI*REACT*REACT	L	670
	DELS=SF-S2	L	680
	IF (DELS.LE.0.) DELS=1.	L	690
	IF (V2.LE.0.) V2=.101	L	700
	ACCEL(NUM)=-(V2*V2)/(2.*DELS)	L	710
	IF (ACCEL(NUM).GT.-FLOAT(I8ITH(ICAR(NUM),46,4))) GO TO 240	L	720
220	CONTINUE	L	730
	IF (IFLAG.FQ.5) GO TO 270	L	740
270	IF (IACC.FQ.12) RETURN	L	750
	ACCTYP=12	L	760
	RETURN	L	770
240	CONTINUE	L	780
	DELS=SF-S2-V2*CYCLE	L	790
	IF (DELS.LE.0.) GO TO 220	L	800
	A1=-(V2*V2)/(2.*DELS)	L	810
	IF (A1.LT.-10.) GO TO 220	L	820
	ACCEL(NUM)=0.	L	830
	IF (IFLAG.FQ.5) GO TO 250	L	840

TABLE B-2, cont.

IF (IACC.EQ.7) RETURN	L 850
ACCTYP=7	L 850
RETURN	L 870
250 DFLS=P(LINK,LANE)-POSN(NUM)	L 880
IF (DELS.LE.0.) DFLS=1.	L 890
ACC=AD*(V(LINK,LANE)-VFL(NUM))/DFLS	L 900
IF (IACC.GE.ACCEL(NUM)) GO TO 260	L 910
ACCEL(NUM)=ACC	L 920
IF (IACC.NE.8) ACCTYP=8	L 930
RETURN	L 940
260 IF (IACC.NE.7) ACCTYP=7	L 950
RETURN	L 960
270 DFLS=P(LINK,LANE)-POSN(NUM)	L 970
IF (DELS.LE.0.) DFLS=1.	L 980
ACC=AD*(V(LINK,LANE)-VFL(NUM))/DFLS	L 990
IF (IACC.GE.ACCEL(NUM)) GO TO 280	L 1000
ACCEL(NUM)=ACC	L 1010
IF (IACC.NE.8) ACCTYP=8	L 1020
RETURN	L 1030
280 CONTINUE	L 1040
N=IFREQAR(NUM)	L 1050
X=P(LINK,LANE)-POSN(NUM)	L 1060
IF (X.GT.300.) GO TO 150	L 1070
IF (IACC.EQ.12) GO TO 410	L 1080
IF (IACC.EQ.11) GO TO 430	L 1090
290 IF (POSN(NUM).GT.CIRRLINK) GO TO 110	L 1100
IF (POSN(NUM).LE.CURRLINK) GO TO 300	L 1110
IFLAG=5	L 1120
GO TO 210	L 1130
300 CONTINUE	L 1140
IF (IACC.LT.7.OR.IACC.EQ.17.AND.X.LT.200.) GO TO 150	L 1150
IF (IPITH(ICAR(N),13,7).NE.ITURN) GO TO 210	L 1160
DFLS=P(LINK,LANE)-POSN(NUM)	L 1170
IF (DELS.LE.0.) DFLS=1.	L 1180
ACCEL(NUM)=AD*(V(LINK,LANE)-VFL(NUM))/DFLS	L 1190
IF (ACCEL(NUM).LE.3.) GO TO 150	L 1200
310 CONTINUE	L 1210
ACC=FLOAT(IPITH(ICAR(NUM),50,4)/7)	L 1220
IF (ACCEL(NUM).LT.ACC) GO TO 320	L 1230
ACCEFS=FLOAT(IPITH(ICAR(NUM),50,4)/7)	L 1240
GO TO 160	L 1250
320 CONTINUE	L 1260
ACC=FLOAT(IPITH(ICAR(NUM),50,4)/7)	L 1270
IF (ACCEL(NUM).LE.ACC) GO TO 330	L 1280
ACCEL(NUM)=ACC	L 1290
ACCEL(NUM)=0.	L 1300
GO TO 310	L 1310
IF (ACCEL(N).EQ.0.) GO TO 370	L 1320
330 CONTINUE	L 1330
IF (IACC.EQ.8) GO TO 340	L 1340
ACCTYP=8	L 1350
340 CONTINUE	L 1360
RETURN	L 1370
350 CONTINUE	L 1380
IF (IACCL.EQ.17.OR.IACCL.EQ.21) GO TO 360	L 1390
IF (IACCL.NE.11) GO TO 310	L 1400
360 CONTINUE	L 1410
IF (ACCEL(N).EQ.0.) GO TO 370	L 1420
SIF=- (VFL(N)*VEL(N))/ (2.*ACCEL(N))+POSN(N)	L 1430
GO TO 350	L 1440
370 SIF=VFL(N)*CYCLE+POSN(N)	L 1450

TABLE B-2, cont.

380	FL=FLOAT (IBITH(ICAR(N),40,6))	L 1460
	S2F=S1F-FL	L 1470
	V22=VEL(NUM)+ACC1*REACT	L 1480
	IF (V22.LE.0.) V22=.001	L 1490
	IF (ACC1.EQ.0.) GO TO 390	L 1500
	S22=POSN(NUM)+VFL(NUM)*REACT+.5*ACC1*REACT*REACT	L 1510
	GO TO 400	L 1520
390	CONTINUE	L 1530
	S22=POSN(NUM)+VFL(NUM)*REACT	L 1540
400	XIX=S2F-S22	L 1550
	IF (XIX.LE.0.) XIX=1.	L 1560
	ACCEL(NUM)=- (V22*V22)/(2.*XIX)	L 1570
	ACC=-FLOAT (IBITH(ICAR(NUM),46,4))	L 1580
	IF (ACCEL(NUM).LT.ACC) GO TO 420	L 1590
	DELS=S2F-S22-V22*CYCLE	L 1600
	IF (DELS.LT.0.) GO TO 420	L 1610
	IF (DELS.EQ.0.) DELS=1.	L 1620
	A=-V22*V22/(2.*DELS)	L 1630
	IF (A.LT.-10.) GO TO 420	L 1640
	ACCEL(NUM)=0.	L 1650
	IF (IACC.EQ.8) RETURN	L 1660
	ACCTYP=8	L 1670
	RETURN	L 1680
410	CONTINUE	L 1690
	IF (IACCL.NE.12) GO TO 290	L 1700
	ACCEL(NUM)=ACC1	L 1710
420	CONTINUE	L 1720
	IF (VFL(NUM).LE.0.) GO TO 430	L 1730
	IF (IACC.EQ.12) RETURN	L 1740
	ACCTYP=12	L 1750
	RETURN	L 1760
430	ELL=FLOAT (IBITH(ICAR(N),40,6))	L 1770
	IF (POSN(NUM).GT.POSN(N)-ELL-10.) GO TO 470	L 1780
	ACC=FREEK*(TVEL-VFL(NUM))	L 1790
	P22=POSN(NUM)+.5*ACC*CYCLE*CYCLE	L 1800
	V22=VFL(NUM)+ACC*CYCLE	L 1810
	P12=POSN(N)+VEL(N)*REACT+.5*ACCEL(N)*REACT*REACT	L 1820
	V12=VEL(N)+ACCEL(N)*REACT	L 1830
	IF (P22.GE.P12-FLL) GO TO 470	L 1840
	IF (ACCEL(N).LT.0.) GO TO 450	L 1850
	P1F=P12+V12*CYCLE+.5*ACCEL(N)*CYCLE*CYCLE	L 1860
440	P2F=P1F-FLL	L 1870
	IF (P2F.LE.P22) GO TO 470	L 1880
	A=-V22*V22/(2.*(P2F-P22))	L 1890
	GO TO 460	L 1900
450	P1F=P12-V12*V12/(2.*ACCEL(N))	L 1910
	GO TO 440	L 1920
460	IF (A.LT.-12.) GO TO 470	L 1930
	ACCEL(NUM)=ACC	L 1940
	IF (IACC.NE.7) ACCTYP=7	L 1950
	RETURN	L 1960
470	CONTINUE	L 1970
	ACCEL(NUM)=0.	L 1980
	VFL(NUM)=0.	L 1990
	IF (IACC.EQ.1) RETURN	L 2000
	ACCTYP=1	L 2010
480	CONTINUE	L 2020
	RETURN	L 2030
C		L 2040
	END	L 2050

TABLE B-2, cont.

SUBROUTINE RIGHT (LINK,LANE,NUM,IFCN)	M	10
COMMON /A1/ IREM,INEXT,IPAX,ISITF,IFRSTCR(R4,3),ILSTCAR(R4,3),VFL(	M	20
1500),POSN(1500),ACCEL(1500),ICAR(1500),NXTCAR(1500),IPRCAR(1500)	M	30
COMMON /PARAM/ A0,PFEX,REACT,CYCLE	M	40
COMMON /GEOMET/ RADIUS(R4,6),REGINPT(R4,6)	M	50
COMMON /PV/ P(R4,3),V(R4,3)	M	60
COMMON /TURN/ KPEC(R4,6),LREC(R4,6),OTURN(R4,6),OTR(R4,2)	M	70
COMMON /RELAT/ LINCAP(R4),NOLINK,LANCAP(R4,3),NOLANE(R4)	M	80
COMMON /GEOM/ CURR(R4),OPNDRY(R4,6)	M	90
COMMON /CAR/ IACC,TVEL,ITURN,EL,IACCL	M	100
COMMON /MSK/ MASK(20)	M	110
EQUIVALENCE (VTURN,ARSVT)	M	120
IPITH(IX,M,N)=ISHFT(IX,M).AND.MASK(N)	M	130
IFLAG=IFCN	M	140
IFOK=0	M	150
NT=1	M	160
IF (ITURN.F0.2) NT=2	M	170
ACCR=ACCEL(NUM)	M	180
IF (FOSN(NUM).GE.CURR(LINK)-300.) GO TO 110	M	190
100 CALL GENPHR (LINK,LANE,NUM,IFLAG)	M	200
IFOK=IFLAG	M	210
RETURN	M	220
110 CONTINUE	M	230
IF (NUM.F0.IFRSTCR(LINK,LANE)) GO TO 120	M	240
IF (IACCL.E0.1.AND.IFLAG.NF.1) GO TO 100	M	250
IF (IACC.E0.1.AND.IACCL.NF.1) GO TO 100	M	260
120 CONTINUE	M	270
VTUPA=SQRT(9.6E*RADIUS(LINK,NT))	M	280
DECHTN=-FLOAT(IPITH(ICAR(NUM),46,4))	M	290
X=BEGINPT(LINK,NT)-(POSN(NUM)+VFL(NUM)*REACT+.5*ACCEL(NUM)*REACT*2	M	300
REACT)	M	310
IF (X.LE.0.) GO TO 150	M	320
CALL GENPHO (LINK,LANE,NUM,IFLAG)	M	330
IFOK=IFLAG	M	340
ARSVX=VEL(NUM)+ACCR*REACT	M	350
IF (ARSVX.LT.0.) ARSVX=0.	M	360
ACC=(ARSVT*ARSVT-ARSVX*ARSVX)/(2.*X)	M	370
IF (ACC.GT.ACCEL(NUM)) GO TO 140	M	380
IF (ACC.GT.DECHTN.AND.ACC.LT.0.) GO TO 130	M	390
GO TO 140	M	400
130 XX=CURR(LINK)-(POSN(NUM)+VFL(NUM)*REACT+.5*ACCR*REACT*REACT+(VFL(	M	410
1NUM)+ACCR*REACT)*CYCLE+.5*ACCEL(NUM)*CYCLE*CYCLE)	M	420
ARSV=VEL(NUM)+ACCR*REACT+ACCEL(NUM)*CYCLE	M	430
IF (ARSV.LT.0.) ARSV=0.	M	440
A=(ARSVT*ARSVT-ARSV*ARSV)/(2.*XX)	M	450
IF (A.GT.-10.) GO TO 140	M	460
ACCEL(NUM)=ACC	M	470
IFCN=11	M	480
140 CONTINUE	M	490
IF (NUM.F0.IFRSTCR(LINK,LANE)) GO TO 160	M	500
N=IPRCAR(NUM)	M	510
IF (IRITH(ICAR(N),33,3).NE.ITURN) GO TO 160	M	520
GO TO 190	M	530
150 ACCEL(NUM)=(VTURN-(VFL(NUM)+ACCR*REACT))/CYCLE	M	540
IF (IACC.NF.10) IE3N=10	M	550
GO TO 220	M	560
160 CONTINUE	M	570
X=PCSN(NUM)+VEL(NUM)*REACT+.5*ACCR*REACT*REACT	M	580
IF (X.LT.REGINPT(LINK,NT)) GO TO 190	M	590

TABLE B-2, cont.

	N=NREC(LINK,ITURN)	M	600
	L=LREC(LINK,ITURN)	M	610
	NQ=LANCAR(N,L)	M	620
	IF (NO.EQ.0) GO TO 190	M	630
	NQ=ILSTCAR(N,L)	M	640
	IF (N.LT.LINK) GO TO 170	M	650
	X=DPNDRY(LINK,ITURN)-PCSN(NUM)+PCSN(NQ)	M	660
	IF (X.LE.0.) X=1.	M	670
	ACC=A0*(VEL(NQ)-VEL(NUM))/X	M	680
	GO TO 190	M	690
170	CONTINUE	M	700
	X=DPNDRY(LINK,ITURN)-PCSN(NUM)+P(N,L)	M	710
	IF (X.LE.0.) X=1.	M	720
	ACC=A0*(V(N,L)-VEL(NUM))/X	M	730
180	CONTINUE	M	740
	IF (ACCEL(NUM).LE.ACC) GO TO 190	M	750
	IF (ACC.GT.DECMIN) GO TO 190	M	760
	ACCFL(NUM)=ACC	M	770
	IF (IACC.NE.1) IEQN=1	M	780
	GO TO 220	M	790
190	CONTINUE	M	800
	CALL CONST (LINK,LANE,NUM,ICON,1)	M	810
	IF (ICON.EQ.0) GO TO 220	M	820
	X=CURB(LINK)-PCSN(NUM)	M	830
	IF (X.LE.0.) GO TO 200	M	840
	IF (X.GT.50.) GO TO 220	M	850
	IF (X.FI.0.) X=.001	M	860
	V2=VEL(NUM)+ACCP*REACT	M	870
	IF (V2.LE.0.) V2=0.	M	880
	ACC=- (V2*V2)/X	M	890
	GO TO 210	M	900
200	ACC=-2.*FLOAT(IRITH(ICAR(NUM),46,4))	M	910
210	CONTINUE	M	920
	IF (ACC.LT.ACCFL(NUM)) ACCEL(NUM)=ACC	M	930
	IF (IACC.NE.2) IEQN=2	M	940
220	CONTINUE	M	950
	ACCPAX=FLOAT(IRITH(ICAR(NUM),50,4))	M	960
	IF (ACCEL(NUM).GT.ACCHMAX) ACCFL(NUM)=ACCHMAX	M	970
	RETURN	M	980
C.	END	M	990
		M	1000
	SUBROUTINE LCFT (LINK,LANE,NUM,ITIME,IEQN)	N	10
	COMMON /A1/ IREM,INEXT,IPAX,ISIZE,IFRSTCR(84,3),ILSTCAR(84,3),VEL(	N	20
	11500),PCSN(1500),ACCFL(1500),ICAR(1500),NXTCAR(1500),IPRECAR(1500)	N	30
	COMMON /PARAM/ A0,FRFEK,PEACT,CYCLE	N	40
	COMMON /OPP/ IOPP(84,5)	N	50
	COMMON /RELAT/ LINCAR(84),NOLINK,LANCAR(84,3),NOLANE(84)	N	60
	COMMON /GEOMET/ RADIUS(84,6),BEGINPT(84,6)	N	70
	COMMON /GEOM/ CURB(84),OBNDRY(84,6)	N	80
	COMMON /TURN/ NREC(84,6),LREC(84,6),PTURN(84,6),PTR(84,2)	N	90
	COMMON /PV/ P(84,3),V(84,3)	N	100
	COMMON /CAR/ IACC,TVFL,ITURN,EL,IACCL	N	110
	COMMON /MSK/ MASK(20)	N	120
	IRITH(IX,M,N)=ISHFT(IX,P).AND.MASK(N)	N	130
	IFGG=IEQN	N	140
	IEQN=0	N	150
	NTD=5	N	160
	NTF=6	N	170

TABLE B-2, cont.

IF (ITURN.EQ.5) GO TO 100	N 180
NTD=3	N 190
NTF=4	N 200
100 CONTINUE	N 210
ACCP=ACCEL(NUM)	N 220
IF (BEGINPT(LINK,NTF)-POSN(NUM).LT.100.) GO TO 110	N 230
CALL GENRMP (LINK,LANE,NUM,IFGO)	N 240
IFGO=IFGO	N 250
GO TO 540	N 260
110 CONTINUE	N 270
JOPP=IOPP(LINK,1)	N 280
IF (IOPP.NE.0) GO TO 120	N 290
IA=3	N 300
GO TO 190	N 310
120 CONTINUE	N 320
NOLLL=NOLANE(JOPP)	N 330
JA=0	N 340
DO 180 LANOP=1,NOLLL	N 350
CALL SIGNAL (JSIG,IOPP,LANOP,ITIME)	N 360
IF (JSIG.EQ.1.OR.JSIG.EQ.2.OR.JSIG.EQ.4.OR.JSIG.EQ.6.OR.JSIG.EQ.12	N 370
1.OR.JSIG.EQ.14) GO TO 130	N 380
IA=2	N 390
GO TO 170	N 400
130 CONTINUE	N 410
IF (JSIG.EQ.14.OR.JSIG.EQ.2) GO TO 140	N 420
IA=3	N 430
GO TO 170	N 440
140 CONTINUE	N 450
CALL CONST (LINK,LANE,NUM,ICON,0)	N 460
IF (ICON.EQ.1) GO TO 150	N 470
CALL CONLT (LINK,LANE,NUM,ICON)	N 480
150 CONTINUE	N 490
IF (ICON.EQ.0) GO TO 160	N 500
IA=1	N 510
GO TO 170	N 520
160 CONTINUE	N 530
IA=2	N 540
170 CONTINUE	N 550
IF (IA.EQ.1) GO TO 190	N 560
IF (IA.EQ.2) JA=2	N 570
180 CONTINUE	N 580
IF (IA.EQ.3.AND.JA.EQ.2) IA=3	N 590
190 CONTINUE	N 600
GO TO (370,440,230,270,470,470,470,470,200,220,120,440), IACC	N 610
200 CONTINUE	N 620
IF (IACC.NE.9) IEQY=9	N 630
V2=VEL(NUM)+ACCP*REACT	N 640
IF (V2.LT.0.) V2=0.	N 650
ACCEL(NUM)=(SQRT(9.56*RADIUS(LINK,NTD))-V2)/CYCLE	N 660
210 IF (NUM.EQ.IFSTOR(LINK,LANE)) GO TO 500	N 670
N=IPRECAR(NUM)	N 680
IF (IBITH(ICAR(N),11,3).NE.ITURN) GO TO 500	N 690
XX=P(LINK,LANE)-POSN(NUM)	N 700
IF (XX.LT.0.) XX=.701	N 710
ACC=A0*(V(LINK,LANE)-VEL(NUM))/XX	N 720
IF (ACC.LT.ACCEL(NUM)) ACCEL(NUM)=ACC	N 730
GO TO 540	N 740
220 CONTINUE	N 750
IF (IACC.NE.10) IE3N=10	N 760
V2=VEL(NUM)+ACCP*REACT	N 770
IF (V2.LT.0.) V2=0.	N 780

TABLE B-2, cont.

ACCEL(NUM)=(SQRT(9.66*RACTUS(LINK,NTF))-V2)/CYCLE	N 790
GO TO 210	N 800
210 CONTINUE	N 810
GO TO (740,750,760), IA	N 820
240 CONTINUE	N 830
ACCEL(NUM)=0.	N 840
IF (IACC.NF.3) IFQ4=1	N 850
GO TO 490	N 860
250 CONTINUE	N 870
CALL GAP (ICON,LINK,LANE,ITIME)	N 880
GO TO (760,740), ICON	N 890
760 CALL GENRMP (LINK,LANE,NUM,IFQ0)	N 900
IFQ4=9	N 910
GO TO 510	N 920
270 CONTINUE	N 930
GO TO (280,300,310), IA	N 940
280 CONTINUE	N 950
IF (VFL(NUM).LF.0.) GO TO 290	N 960
IF (IACC.NF.4) IFQ4=4	N 970
GO TO 490	N 980
290 CONTINUE	N 990
ACCEL(NUM)=0.	N 1000
VFL(NUM)=0.	N 1010
IF (IACC.NF.3) IFQ4=3	N 1020
GO TO 490	N 1030
300 CONTINUE	N 1040
CALL GAP (ICON,LINK,LANE,ITIME)	N 1050
GO TO (310,280), ICON	N 1060
310 IF (PCSN(NUM)+VFL(NUM)*REACT+.5*ACCP*REACT*REACT).GT.BEGINPT(LINK,NTF)) GO TO 200	N 1070
GO TO 220	N 1080
320 CONTINUE	N 1090
IF (PCSN(NUM)+VEL(NUM)*REACT+.5*ACCP*REACT*REACT+30.*CYCLE).GT.BEGINPT(LINK,NTF)) GO TO 310	N 1100
GO TO 530	N 1110
330 GO TO (340,350,360), IA	N 1120
340 X=BEGINPT(LINK,NTF)+(DRNDY(LINK,ITURN)-BEGINPT(LINK,NTF))/4.-(POS1N(NUM)+VEL(NUM)*REACT+.5*ACCP*REACT*REACT)	N 1130
IF (X.LE.0.) X=.001	N 1140
V2=VFL(NUM)+ACCP*REACT	N 1150
IF (V2.LE.0.) V2=0.	N 1160
ACCEL(NUM)=-(V2*V2)/(2.*X)	N 1170
IFQ4=4	N 1180
GO TO 540	N 1190
350 CALL GAP (ICON,LINK,LANE,ITIME)	N 1200
GO TO (360,340), ICON	N 1210
360 IF (PCSN(NUM)+VFL(NUM)*REACT+.5*ACCP*REACT*REACT).GT.BEGINPT(LINK,NTF)) GO TO 220	N 1220
GO TO 490	N 1230
370 IF (PCSN(NUM)+VFL(NUM)*REACT+.5*ACCP*REACT*REACT+30.*CYCLE).GT.BEGINPT(LINK,NTF)) GO TO 390	N 1240
380 CALL GENRMP (LINK,LANE,NUM,IFQ0)	N 1250
IFQ4=IFQ0	N 1260
GO TO 540	N 1270
390 GO TO (400,410,420), IA	N 1280
400 IF (NUM.NF.IFSTCP(LINK,LANE)) GO TO 380	N 1290
IFQ4=3	N 1300
GO TO 540	N 1310
410 IF (PCSN(NUM).LT.BEGINPT(LINK,NTF)-100.) GO TO 430	N 1320
CALL GAP (ICON,LINK,LANE,ITIME)	N 1330
GO TO (420,400), ICON	N 1340

TABLE B-2, cont.

420	CALL CONST (LINK,LANE,NUM,ICON,0)	N 1400
	IF (ICON.EQ.1) GO TO 400	N 1410
430	CALL GENRHP (LINK,LANE,NLM,IFGO)	N 1420
	IFON=IFGO	N 1430
	GO TO 540	N 1440
440	IF (POSN(NUM)+VEL(NUM)*REACT+.5*ACCP*REACT*REACT+.10.*CYCLE*.LT.REF	N 1450
	1 INPT(LINK,NTF)) GO TO 450	N 1460
	IFON=11	N 1470
	XX=REGINPT(LINK,NTF)-(POSN(NUM)+VEL(NUM)*REACT+.5*ACCP*REACT*REACT	N 1480
	1T)	N 1490
	IF (XX.LE.0.) GO TO 330	N 1500
	V2=VEL(NUM)+ACCP*REACT	N 1510
	IF (V2.LT.0.) V2=0.	N 1520
	ACCEL(NUM)=(9.86*RADIUS(LINK,NTF)-V2*V2)/(2.*XX)	N 1530
	GO TO 330	N 1540
450	CALL GENRHR (LINK,LANE,NUM,IFGO)	N 1550
	IFON=IFGO	N 1560
	XX=REGINPT(LINK,NTF)-(POSN(NUM)+VEL(NUM)*REACT+.5*ACCP*REACT*REACT	N 1570
	1T)	N 1580
	IF (XX.LE.0.) XX=1.	N 1590
	V2=VEL(NUM)+ACCP*REACT	N 1600
	IF (V2.LT.0.) V2=0.	N 1610
	ACC=(9.86*RADIUS(LINK,NTF)-V2*V2)/(2.*XX)	N 1620
	IF (ACC.GE.ACCEL(NUM)) GO TO 540	N 1630
	IF (ACC.GE.-FLCAT(IRITH(ICAR(NUM),46,41).AND,ACC.LT.0.) GO TO 460	N 1640
	ACCEL(NUM)=ACC	N 1650
	IFON=11	N 1660
	GO TO 540	N 1670
460	XX=REGINPT(LINK,NTF)-(POSN(NUM)+VEL(NUM)*REACT+.10.*CYCLE*REACT)	N 1680
	IF (XX.LT.0.) GO TO 540	N 1690
	IF (XX.LE.0.) XX=.101	N 1700
	A=(9.86*RADIUS(LINK,NTF)-VEL(NUM)*VEL(NUM))/(2.*XX)	N 1710
	IF (A.LE.-10.) GO TO 540	N 1720
	ACCEL(NUM)=ACC	N 1730
	IFON=11	N 1740
	GO TO 540	N 1750
470	XX=REGINPT(LINK,NTF)-(POSN(NUM)+VEL(NUM)*REACT+.1*ACCP*REACT*REACT	N 1760
	1T)	N 1770
	IF (XX.GT.10.*CYCLE) GO TO 450	N 1780
	IF (XX.LE.0.) GO TO 320	N 1790
	GO TO (340,480,220), IA	N 1800
480	CALL GAP (ICON,LINK,LANE,ITIME)	N 1810
	GO TO (220,340), ICON	N 1820
490	CONTINUE	N 1830
	IF (NUM.EQ.IFSTCR(LINK,LANE)) GO TO 500	N 1840
	N=IPRECAR(NUM)	N 1850
	IF (IBITH(ICAR(N),33,31).NE.ITURN.AND,POSN(NUM).GE.REGINPT(LINK,NTF	N 1860
	1) GO TO 500	N 1870
	GO TO 530	N 1880
500	CONTINUE	N 1890
	X=POSN(NUM)+VEL(NUM)*REACT+.5*ACCEL(NUM)*REACT*REACT	N 1900
	IF (X.LT.DRNDRY(LINK,ITURN)) GO TO 530	N 1910
	N=NFFC(LINK,ITURN)	N 1920
	L=LFFC(LINK,ITURN)	N 1930
	NO=LANCAR(N,L)	N 1940
	IF (NO.EQ.0) GO TO 530	N 1950
	NO=ILSTCAR(N,L)	N 1960
	IF (N.LT.LINK) GO TO 510	N 1970
	XX=DRNDRY(LINK,ITURN)-POSN(NUM)+POSN(NO)	N 1980
	IF (XX.LE.0.) XX=1.	N 1990
	ACC=40*(VEL(NO)-VEL(NUM))/XX	N 2000

TABLE B-2, cont.

	GO TO 520	N 2010
510	CONTINUE	N 2020
	XX=DPNDRY(LINK,ITURN)-POSN(NUM)+P(N,L)	N 2030
	IF (XX.EQ.0.) XX=1.	N 2040
	ACC=80*(V(N,L)-VFL(NUM))/XX	N 2050
520	CONTINUE	N 2060
	IF (ACCEL(NUM).LE.ACC) GO TO 530	N 2070
	ACCEL(NUM)=ACC	N 2080
	IF (IACC.NE.8) IEQV=P	N 2090
530	CONTINUE	N 2100
	ACCMAX=FLOAT(IRITH(ICAR(NUM),50,4))	N 2110
	IF (ACCEL(NUM).GT.ACCMAX) ACCEL(NUM)=ACCMAX	N 2120
540	CONTINUE	N 2130
	RETURN	N 2140
C		N 2150
	END	N 2160

	SUBROUTINE CONLT (LINK,LANE,NUM,ICON)	O 10
	COMMON /A1/ IREM,INEXT,IMAX,ISIZE,IFRSTCR(84,3),ILSTCAR(84,3),VEL(	O 20
	11500),POSN(1500),ACCEL(1500),ICAR(1500),NXTCAR(1500),IPRECAR(1500)	O 30
	COMMON /GEOM/ CURR(84),DPNDRY(84,6)	O 40
	COMMON /OPP/ IOPP(84,5)	O 50
	COMMON /RELAT/ LINCAR(84),NOLINK,LANCAR(84,3),NOLANE(84)	O 60
	I=ICFP(LINK,3)	O 70
	IF (I.EQ.0) GO TO 130	O 80
	NOLLL=NOLANE(I)	O 90
	DO 120 LAN=1,NOLLL	O 100
	IF (LANCAR(I,LAN).EQ.0) GO TO 110	O 110
	NO=IFRSTCR(I,LAN)	O 120
100	IF (NO.EQ.0) GO TO 120	O 130
	IF (POSN(NO).LE.CURR(I)) GO TO 120	O 140
	IF (POSN(NO).LE.CURR(I)+75.) GO TO 140	O 150
	NO=NXTCAR(NO)	O 160
	GO TO 100	O 170
110	CONTINUE	O 180
120	CONTINUE	O 190
130	CONTINUE	O 200
	ICON=0	O 210
C		O 220
C	ICON=0 MEANS NO CONFLICT EXISTS	O 230
C		O 240
	RETURN	O 250
140	CONTINUE	O 260
	ICON=1	O 270
C		O 280
C	ICON=1 MEANS CONFLICT EXISTS	O 290
C		O 300
	RETURN	O 310
C		O 320
	END	O 330

	SUBROUTINE CONST (LINK,LANE,NUM,ICON,IFL)	P 10
	COMMON /A1/ IREM,INEXT,IMAX,ISIZE,IFRSTCR(84,3),ILSTCAR(84,3),VEL(	P 20
	11500),POSN(1500),ACCEL(1500),ICAR(1500),NXTCAR(1500),IPRECAR(1500)	P 30
	COMMON /RELAT/ LINCAR(84),NOLINK,LANCAR(84,3),NOLANE(84)	P 40
	COMMON /OPP/ IOPP(84,5)	P 50
	COMMON /GEOM/ CURR(84),DPNDRY(84,6)	P 60

TABLE B-2, cont.

	COMMON /CAP/ IACC,TVFL,ITURN,FL,IACCL	P	70
	COMMON /MSK/ MASK(70)	P	80
	IRITH(I,M,N)=ISHTP(I,M,ANG,MASK(N)	P	90
	IF (IFL.FQ.1) GO TO 150	P	100
	I=ICPP(LINK,1)	P	110
100	IF (I.EQ.0) GO TO 170	P	120
	NOLLL=NOLANE(I)	P	130
	DO 140 LAN=1,NOLLL	P	140
	IF (LANCAR(I,LAN).EQ.0) GO TO 140	P	150
	NO=IFSTCR(I,LAN)	P	160
110	IF (NO.FQ.0) GO TO 140	P	170
	IF (POSN(NO).LT.CURR(I)) GO TO 140	P	180
	IF (VFL(NO).LF.0.) GO TO 140	P	190
	IF (POSN(NO).GT.CURR(I)+75.) GO TO 170	P	200
	ICCN=1	P	210
	RETURN	P	220
170	CONTINUE	P	230
	NO=NXTCAR(NO)	P	240
	GO TO 110	P	250
170	CONTINUE	P	260
140	CONTINUE	P	270
	IF (I.EQ.ICPP(LINK,2)) GO TO 150	P	280
	I=ICPP(LINK,2)	P	290
	GO TO 100	P	300
150	I=ICPP(LINK,4)	P	310
160	IF (I.FQ.0) GO TO 190	P	320
	NOLLL=NOLANE(I)	P	330
	DO 200 LAN=1,NOLLL	P	340
	IF (LANCAR(I,LAN).FQ.0) GO TO 200	P	350
	NO=IFSTCR(I,LAN)	P	360
170	IF (NO.EQ.0) GO TO 200	P	370
	IF (POSN(NO).LT.CURR(I)) GO TO 200	P	380
	IF (VFL(NO).LF.0.) GO TO 200	P	390
	IF (POSN(NO).LT.CURR(I)+55.) GO TO 140	P	400
	NO=NXTCAR(NO)	P	410
	GO TO 170	P	420
140	CONTINUE	P	430
	IF (IRITH(ICAR(NO),11,1).GF.3.AND.ITURN.LT.5) GO TO 220	P	440
190	CONTINUE	P	450
200	CONTINUE	P	460
	IF (I.FQ.ICPP(LINK,5)) GO TO 210	P	470
	I=ICPP(LINK,5)	P	480
	GO TO 160	P	490
210	CONTINUE	P	500
	ICCN=0	P	510
C		P	520
C	ICCN=0 MEANS NO CONFLICT EXISTS	P	530
C		P	540
	RETURN	P	550
220	CONTINUE	P	560
	ICCN=1	P	570
C		P	580
C	ICCN=1 MEANS CONFLICT EXISTS	P	590
C		P	600
	RETURN	P	610
C		P	620
	END	P	630
	SUBROUTINE GAP (ICCN,LINK,LANE,ITIME)	P	10

TABLE B-2, cont.

COMMON /A1/ IREM, ITEXT, IPAX, ISIZF, IFRSTCR(84,3), ILSTCAR(84,3), VEL(	Q	20
11500), POSN(1500), ACCFL(1500), ICAR(1500), NXTCAR(1500), IPRECAR(1500)	Q	30
COMMON /GDM/ CURB(84), CPNDRY(84,6)	Q	40
COMMON /OPP/ IOPP(84,5)	Q	50
COMMON /SIG/ LCYCLE(84), NOPHAS(84,3), NPHASE(84,3,5), LPHASE(84,3,5)	Q	60
1, LOFFST(84,3)	Q	70
COMMON /RELAT/ LINCAR(84), NOLINK, LANGAP(84,3), NOLANE(84)	Q	80
COMMON /MSK/ MASK(20)	Q	90
IRITH(IX,M,N)=ISHFTR(IX,M).AND.MASK(N)	Q	100
DIMENSION TABLE1(15), TABLE2(15)	Q	110
INTEGER OFFPT	Q	120
DATA (TABLE1(I), I=1,15)/.10,.23,.42,.60,.76,.86,.92,.96,.98,.99,.9	Q	130
199,.9975,.999,.99945,.9997/	Q	140
DATA (TABLE2(I), I=1,15)/.01,.06,.20,.40,.62,.80,.90,.96,.982,.993,	Q	150
1.9975,.9991,.9997,.9998,.9999/	Q	160
RN=RANF(0)	Q	170
I=ICFP(LINK,3)	Q	180
IF (I.EQ.0) GO TO 170	Q	190
NOLLL=NOLANE(I)	Q	200
DO 170 LAN=1,NOLLL	Q	210
IF (LANCAR(I,LAN).EQ.0) GO TO 170	Q	220
NO=IFRSTCR(I,LAN)	Q	230
100 IF (NO.EQ.0) GO TO 170	Q	240
IF (POSN(NO).GT.CURB(I)+35.) GO TO 160	Q	250
IQ=IBITH(ICAR(NO),13,3)	Q	260
IF (IQ.GT.4) GO TO 160	Q	270
IF (POSN(NO).GT.CURB(I)+20.) GO TO 160	Q	280
X=CURB(I)-POSN(NO)+20.	Q	290
IF (ACCEL(NO).LE.0.) GO TO 130	Q	300
110 T1=VEL(NO)*VEL(NO)+2.*ACCEL(NO)*X	Q	310
T=ARS(VEL(NO)-SQRT(T1))/ACCEL(NO)	Q	320
GO TO 140	Q	330
120 XF=POSN(NO)-VEL(NO)*VEL(NO)/(2.*ACCEL(NO))	Q	340
IF (XF.LE.CURB(I)+20.) GO TO 170	Q	350
GO TO 110	Q	360
130 IF (VEL(NO).EQ.0.) GO TO 170	Q	370
IF (ACCEL(NO).LT.0.) GO TO 120	Q	380
T=X/VEL(NO)	Q	390
140 CONTINUE	Q	400
IF (T.LT.3.) GO TO 180	Q	410
IF (T.GT.10.) GO TO 170	Q	420
IT=2.*T-5.	Q	430
NCYC=(ITIME-LOFFST(LINK,LANE))/(LCYCLE(LINK))	Q	440
OFFPT=NCYC*LCYCLE(LINK)+LOFFST(LINK,LANE)	Q	450
IF (ITIME-OFFPT.LE.2) GO TO 150	Q	460
IF (TABLE1(IT).LE.RN) GO TO 160	Q	470
GO TO 170	Q	480
150 IF (TABLE2(IT).LE.RN) GO TO 160	Q	490
GO TO 170	Q	500
160 CONTINUE	Q	510
NO=NXTCAR(NO)	Q	520
GO TO 100	Q	530
170 CONTINUE	Q	540
ICON=1	Q	550
C	Q	560
C	Q	570
C	Q	580
RETURN	Q	590
180 CONTINUE	Q	600
ICON=7	Q	610
C	Q	620

TABLE B-2, cont.

C	IFOK=2 MEANS UNACCEPTABLE GAP	Q	630
C	RETURN	Q	640
C	END	Q	650
		Q	660
		Q	670
	SUBROUTINE ROKK (LINK,LANE,NUM,ITIME)	Q	10
	COMMON /A1/ IPFH,INEXT,IMAX,IST7F,IFRSTCR(84,3),ILSTCAR(84,3),VFL(	Q	20
	11500),POSN(1500),ATCFL(1500),ICAR(1500),NXTCAR(1500),IPRECAR(1500)	Q	30
	COMMON /CAR/ IACC,TVFL,ITURN,EL,IACCL	Q	40
	COMMON /PARAM/ 80,FREK,RFAC,CYCLE	Q	50
	COMMON /GEOM/ CURR(84),DRNDRY(84,8)	Q	60
	COMMON /RUK/ NHOLD(84,3)	Q	70
	COMMON /R2/ TTIME(84),TDIST(84),IRTIME(15,105),TTIMEF,TTDIST,WSTQ(	Q	80
	115,3,14)	Q	90
	COMMON /R3/ DELAY(84),DELOFV(84),NCALL(84)	Q	100
	COMMON /TRCE/ NTRAC,LKTRCE(15),NTRC(15),INTTRC,KNTRAC(15)	Q	110
	COMMON /KNMTZ/ NCAR(84),NOITPT,ISTART,IFNO,KSEN(84)	Q	120
	COMMON /MSK/ MASK(70)	Q	130
	DIMENSION HQ(105)	Q	140
	IRITH(IX,M,N)=TSHTF(IX,M).AND.MASK(N)	Q	150
	REAL LNKTIM	Q	160
	IF (FOSN(NUM).LT.DRNDRY(LINK,ITURN)) GO TO 140	Q	170
	NHOLD(LINK,LANE)=NHOLD(LINK,LANE)+1	Q	180
	NCAR(LINK)=NCAR(LINK)+1	Q	190
	LNKTIM=FLOAT(ITIME-IRITH(ICAR(NUM),8,1))	Q	200
	TTIMEF=TTIME+LNKTIM	Q	210
	START=FLOAT(IRITH(ICAR(NUM),14,9))	Q	220
	IF (LINK.GF.ISTART) START=-START	Q	230
	TTDIST=TTDIST+POSN(NUM)-START	Q	240
	TTIME(LINK)=TTIME(LINK)+LNKTIM	Q	250
	TDIST(LINK)=TDIST(LINK)+FOSN(NUM)-START	Q	260
	UNDFL=(POSN(NUM)-START)/TVFL	Q	270
	DFL=LNKTIM-UNDFL	Q	280
	CALL STIX5 (DELAY(LINK),DELOFV(LINK),DEL,NCALL(LINK),VMEAN,STOFV)	Q	290
	IPOS=INT(POSN(NUM)-DRNDRY(LINK,ITURN))	Q	300
	CALL JBITH (ICAR(NUM),14,9,IPOS)	Q	310
	DO 100 I=1,NTRAC	Q	320
	IF (LINK.EQ.LKTRCE(I)) GO TO 110	Q	330
100	CONTINUE	Q	340
	GO TO 140	Q	350
110	IF (IRITH(ICAR(NUM),56,1).EQ.0) GO TO 140	Q	360
	KNTRAC(I)=KNTRAC(I)+1	Q	370
	DO 120 J=1,105	Q	380
120	HQ(J)=IRTIME(I,J)	Q	390
	CALL STIX2 (HQ,700,0,100,LNKTIM,11HTPAVEL TIME,11,LINK)	Q	400
	DO 130 J=1,105	Q	410
130	TTIME(I,J)=HQ(J)	Q	420
140	CONTINUE	Q	430
	RETURN	Q	440
C	END	Q	450
		Q	460
	SUBROUTINE OUTPT (ITIME)	S	10
	COMMON /A1/ IPFH,INEXT,IMAX,IST7F,IFRSTCR(84,3),ILSTCAR(84,3),VEL(	S	20
	11500),POSN(1500),ATCFL(1500),ICAR(1500),NXTCAR(1500),IPRECAR(1500)	S	30
	COMMON /70/ IHOLE(1500),LASTONE	S	40

TABLE B-2, cont.

COMMON /GEOM/ CURR(84),DRNDRY(84,6)	S	60
COMMON /RELAT/ LINCAR(84),NOLINK,LANCAP(84,3),NOLANE(84)	S	80
COMMON /TURN/ NREC(84,6),LFEC(84,6),PTURN(84,6),PTRN(84,2)	S	70
COMMON /RUK/ NHOLD(84,3)	S	80
COMMON /JOU/ LQLEUF(15,3),LQ(10),NQ	S	90
COMMON /R2/ TTIME(34),TOIST(84),TPTIME(15,105),TTTIME,TTDIST,MSTQI	S	100
115,3,14)	S	110
COMMON /TRACE/ NTRACE,LKTRCF(15),NCTR(15),INTTRC,KNTFAC(15)	S	120
COMMON /KWNT7/ NCAR(84),NOUTPT,ISTART,IFNO,KSCN(84)	S	130
COMMON /CAPACTY/ LNCAP	S	140
COMMON /PARAM2/ TVELHN(84,2),TVELSIG(84,2),ACOMN(2),ACCSIG(2),DECH	S	150
IN(2),DECSIG(2),FLMN(2),ELSIG(2),PTRTRUCK(84)	S	160
COMMON /MSK/ MASK(20)	S	170
DIMENSION HQ(45)	S	180
REAL NORMAL	S	190
IPITH(IX,M,N)=ISHFT(IX,M).AND.MASK(N)	S	200
DIMENSION PTR(6)	S	210
DO 270 LINK=1,NOLINK	S	220
NOLNN=NOLANE(LINK)	S	230
DO 270 LANE=1,NOLNN	S	240
IF (NHOLD(LINK,LANE).EQ.0) GO TO 270	S	250
NMMH=NHOLD(LINK,LANE)	S	260
DO 260 NUM=1,NMMH	S	270
INDEX=IFRSTCR(LINK,LANE)	S	280
IF=NXTCAR(INDEX)	S	290
IF (LINK.LE.NOUTPT) GO TO 200	S	300
ITURN=IBITH(ICAP(INDEX),33,3)	S	310
POSA(INDEX)=POSN(INDEX)-DBNDRY(LINK,ITURN)	S	320
N=NREC(LINK,ITURN)	S	330
L=LREC(LINK,ITURN)	S	340
LINCAR(N)=LINCAR(N)+1	S	350
LANCAR(N,L)=LANCAR(N,L)+1	S	360
LNCAP=LANCAR(N,L)	S	370
CALL JBITH (ICAP(INDEX),0,13,ITIME)	S	380
IL=ILSTCAR(N,L)	S	390
IF (IL.EQ.0) GO TO 100	S	400
NXTCAR(IL)=INDEX	S	410
100 ILSTCAR(N,L)=INDEX	S	420
IPRECAP(INDEX)=IL	S	430
NXTCAR(INDEX)=0	S	440
IF (LANCAR(N,L).EQ.1) IFRSTCR(N,L)=INDEX	S	450
RN=RANF(0)	S	460
IF (NOLANE(LINK).GT.1) GO TO 120	S	470
DO 110 I=1,6	S	480
JTURN=I	S	490
IF (RN.LE.PTURN(LINK,I)) GO TO 140	S	500
110 CONTINUE	S	510
GO TO 140	S	520
120 PTR(1)=PTURN(N,1)	S	530
PTP(2)=PTURN(N,2)	S	540
PTR(3)=PTRN(N,L)	S	550
PTR(4)=PTURN(N,4)	S	560
PTR(5)=PTURN(N,5)	S	570
PTR(6)=PTURN(N,6)	S	580
IF (ITURN.NE.3.AND.ITURN.NE.4) PTR(3)=PTURN(N,3)	S	590
DO 130 I=1,6	S	600
JTURN=I	S	610
IF (RN.LE.PTR(I)) GO TO 140	S	620
130 CONTINUE	S	630
140 CONTINUE	S	640
DO 150 I=1,NTRACE	S	650

TABLE B-2, cont.

	IF (N.FQ.LKTRCF(I).AND.LINK.FQ.LKTRCF(I-1)) GO TO 160	S 460
150	CONTINUE	S 470
	CALL JRITH (ICAR(INDEX),56,1,0)	S 480
	GO TO 190	S 490
160	IF (I+1.GT.NTRACE) GO TO 170	S 700
	IF (NPRC(IN,JTURN).FQ.LKTRCF(I+1)) GO TO 170	S 710
	CALL JRITH (ICAR(INDEX),56,1,0)	S 720
	GO TO 190	S 730
170	IF (INTTRC.EQ.NOTRC(I)) GO TO 180	S 740
	NOTRC(I)=NOTRC(I)+1	S 750
	CALL JRITH (ICAR(INDEX),56,1,0)	S 760
	GO TO 190	S 770
180	NOTRC(I)=0	S 780
	CALL JRITH (ICAR(INDEX),56,1,1)	S 790
190	CALL JBITH (ICAR(INDEX),31,3,JTURN)	S 800
	IVENTYP=JBITH(ICAR(INDEX),54,2)	S 810
	ITARVEL=NORMAL(TVELMN(N,IVENTYP),TVELSIG(N,IVENTYP))	S 820
	CALL JRITH (ICAR(INDEX),26,7,ITARVEL)	S 830
	CALL JBITH (ICAR(INDEX),36,4,7)	S 840
200	CONTINUE	S 850
	LINKAR(LINK)=LINKAR(LINK)-1	S 860
	LANCAR(LINK,LANE)=LANCAR(LINK,LANE)-1	S 870
	IF (LANCAR(LINK,LANE).FQ.0) ILSTCAR(LINK,LANE)=0	S 880
	IF (LANCAR(LINK,LANE).FQ.0) IFRSTCR(LINK,LANE)=0	S 890
	DO 210 I=1,NQ	S 900
	IF (LINK.EQ.LQ(I)) GO TO 270	S 910
210	CONTINUE	S 920
	GO TO 270	S 930
220	IF (LQUEUF(I,LANE).LF.0) GO TO 270	S 940
	LQUEUF(I,LANE)=LQUEUF(I,LANE)-1	S 950
270	CONTINUE	S 960
	IF (LANCAR(LINK,LANE).FQ.0) GO TO 240	S 970
	IFRSTCR(LINK,LANE)=IF	S 980
	IPRECAR(IF)=0	S 990
240	IF (LANCAR(LINK,LANE).EQ.0) ILSTCAR(LINK,LANE)=0	S 1000
	IF (LINK.GT.NOUTPT) GO TO 260	S 1010
	LASTONF=LASTONE+1	S 1020
	IF (LASTONE.GT.ISITE) GO TO 250	S 1030
	INOLE(LASTONE)=INDEX	S 1040
250	CONTINUE	S 1050
260	CONTINUE	S 1060
270	CONTINUE	S 1070
	DO 290 I=1,NQ	S 1080
	L=LC(I)	S 1090
	N=NOLANE(L)	S 1100
	DO 290 LN=1,N	S 1110
	DO 280 J=1,14	S 1120
280	HQ(J)=HSTQ(I,LN,J)	S 1130
	QLEN=FLOAT(LQUEUE(I,LN))	S 1140
	CALL STIX2 (HQ,10.,0.,10,QLEN,12HQUEUE LFNGTH,12,LINK)	S 1150
	DO 290 J=1,14	S 1160
290	HSTQ(I,LN,J)=HQ(J)	S 1170
	RETURN	S 1180
C	END	S 1190
		S 1200

SUBROUTINE QUEUE (LINK,LANE)	T 10
COMMON /A1/ IRFM,INEXT,IPAX,ISIZE,IFRSTCR(84,3),ILSTCAR(84,3),VEL(	T 20
15000),POSN(1500),ACCF(1500),ICAR(1500),NXTCAR(1500),IPRECAR(1500)	T 30

TABLE B-2, cont.

	COMMON /OU/ LQFUE(15,3),LQ(10),NLO	T	40
	COMMON /RELAT/ LINCAP(84),NOLINK,LANCAP(84,3),NOLANE(84)	T	50
	COMMON /MSK/ MASK(70)	T	60
	IRITH(IY,M,N)=ICHFT(IY,P).AND,MASK(N)	T	70
	DO 100 I=1,NLO	T	80
	IF (LINK.EQ.LQ(I)) GO TO 110	T	90
100	CONTINUE	T	100
	GO TO 160	T	110
110	CONTINUE	T	120
	IF (LOUEUE(I,LANF).EQ.0) GO TO 150	T	130
	LLO=LQFUE(I,LANE)	T	140
	IF=IFSTCR(LINK,LANE)	T	150
	IF (LLO.GT.1) GO TO 120	T	160
	NQ=NXTCAP(IF)	T	170
	GO TO 140	T	180
120	DO 130 INTE=2,LLO	T	190
130	IF=NXTCAP(IF)	T	200
	IF (IF.EQ.ILSTCAP(LINK,LANF)) GO TO 160	T	210
	NQ=NXTCAP(IF)	T	220
140	CONTINUE	T	230
	IF (NQ.EQ.0) GO TO 160	T	240
	IQ=IFITH(ICAR(NQ),36,4)	T	250
	IF (IQ.NF.1.AND.IQ.NF.1.AND.IQ.NF.6) GO TO 150	T	260
	LQFUE(I,LANF)=LQFUE(I,LANF)+1	T	270
	IF (NQ.EQ.ILSTCAP(LINK,LANE)) GO TO 160	T	280
	NQ=NXTCAP(NQ)	T	290
	GO TO 140	T	300
150	CONTINUE	T	310
	IF (LANCAP(LINK,LANF).EQ.0) GO TO 160	T	320
	NQ=IFSTCR(LINK,LANF)	T	330
	GO TO 140	T	340
160	CONTINUE	T	350
	RETURN	T	360
C		T	370
	END	T	380
	SUBROUTINE SUMAPT (VOL,ITIME,SUMKNT)	U	10
	COMMON /RELAT/ LINCAP(84),NOLINK,LANCAP(84,3),NOLANE(84)	U	20
	COMMON /R2/ TTIME(84),TOIST(84),TRTIME(15,105),TTIME,TTOIST,WSTO(	U	30
	115,3,14)	U	40
	COMMON /R3/ DELAY(84),OFLDEV(84),NCALL(84)	U	50
	COMMON /TRCF/ NTRACE,LKTRCF(15),NOTRC(15),INITRC,KNTRAC(15)	U	60
	COMMON /RWNTZ/ NCAR(84),NOUTPT,ISTART,IFAD,KGEN(74)	U	70
	COMMON /OU/ LQFUE(15,3),LQ(10),NO	U	80
	COMMON /OPLTO/ OUN(5),IPLOT	U	90
	COMMON /TIMKK/ INITIM,LRUN,MINITIM	U	100
	INTGFR SUMKNT	U	110
	DIMENSION HQ(105)	U	120
	DIMENSION VOL(84)	U	130
	IPLOT=0	U	140
	KTIP=(ITIME-INITIM)/60	U	150
	WRITE (6,340) SUMKNT,KTIP	U	160
	WRITE (6,350)	U	170
	DO 140 LK=1,NOLINK	U	180
	WRITE (6,360) LINK,NCAR(LINK)	U	190
	IF (LINK.LT.ISTART) GO TO 180	U	200
	WRITE (6,370) KCONELINK	U	210
100	DO 110 LW=1,NTRACE	U	220
	IF (LINK.EQ.LKTRCF(LW)) GO TO 120	U	230

TABLE B-2, cont.

110	CONTINUE	U	740
	GO TO 130	U	750
120	WRITE (6,390) KNTRAC(LK)	U	760
130	WRITE (6,390)	U	770
140	CONTINUE	U	780
	IF (TTTIME.EQ.0.) TTTIME=1.	U	790
	SPEED=TTDIST/TTTIME/1.47	U	800
	WRITE (6,400)	U	810
	NV=0	U	820
	DO 150 I=1,NOLINK	U	830
150	NV=NV+NCAR(I)	U	840
	TVH=TTDIST/5280.	U	850
	T0=0.	U	860
	DO 160 I=1,NOLINK	U	870
160	T0=DFLAY(I)+T0	U	880
	ATD=TO/FLOAT(NV)	U	890
	ATDP=TO/TVH	U	900
	TTR=TTTIME/TVH	U	910
	SUMSQ=0.	U	920
	DO 170 I=1,NOLINK	U	930
170	SUMSQ=SUMSQ+DELOEV(I)	U	940
	STDEV=SQRT(ABS(SUMSQ/FLOAT(NV)-ATD*ATD))	U	950
	ATT=TTTIME/FLOAT(NV)	U	960
	WRITE (6,410) NV,TVH,TO,ATD,STDEV,ATDP,TTTIME,ATT,TTR,SPEED	U	970
	WRITE (6,420)	U	980
	DO 260 L=1,NOLINK	U	990
	CALL STIX6 (DELAY(L),DELOEV(L),DUM1,NCALL(L),VMEAN,STDEV)	U	1000
	IF (TTIME(L).EQ.0.) GO TO 180	U	1010
	SPEED=TDIST(L)/TTIME(L)/1.47	U	1020
	GO TO 190	U	1030
180	SPEED=0.	U	1040
190	IF (NCAR(L).EQ.0) GO TO 200	U	1050
	TIM=TTIME(L)/FLOAT(NCAR(L))	U	1060
	GO TO 210	U	1070
200	TIM=0.	U	1080
210	TVH=TDIST(L)/5280.	U	1090
	IF (TDIST(L).EQ.0.) GO TO 220	U	1100
	ATDP=DELAY(L)/TVH	U	1110
	TTR=TTIME(L)/TVH	U	1120
	GO TO 230	U	1130
220	ATDP=0.	U	1140
	TTR=0.	U	1150
230	IF (NCAR(L).EQ.0) GO TO 240	U	1160
	ATT=TTIME(L)/FLOAT(NCAR(L))	U	1170
	GO TO 250	U	1180
240	ATT=0.	U	1190
250	WRITE (6,410) NCAR(L),TVH,DELAY(L),VMEAN,STDEV,ATDP,TTIME(L),ATT,TTR,SPEED	U	1200
260	WRITE (6,430) L	U	1210
	DO 290 LK=1,NTRACE	U	1220
	LNKE=LKTRCE(LK)	U	1230
	IF (KNTRAC(LK).EQ.0) GO TO 280	U	1240
	DO 270 I=1,105	U	1250
270	HQ(I)=TRTIME(LK,I)	U	1260
	CALL STIX3 (HQ,200.,0.,100,LNKTIM,11HTRAVFL TIME,11,LNKE,0)	U	1270
	GO TO 290	U	1280
280	WRITE (6,440) LNKE	U	1290
290	CONTINUE	U	1300
	DO 330 L=1,N0	U	1310
	LK=LC(L)	U	1320
	N=NOLANE(LK)	U	1330

TABLE B-2, cont.

07 110 LN=1,N	U	850
07 300 I=1,12	U	860
110 T=T+HSTQ(L,N,I)	U	870
IF (T.EQ.0.) GO TO 170	U	880
07 110 J=1,14	U	890
110 H7(J)=HSTQ(L,N,J)	U	900
CALL STIY (HQ,10.,0.,10,DUM,12HCUFUE LENGTH,17,LK,LN)	U	910
GO TO 110	U	920
170 WRITE (6,450) LK	U	930
110 CONTINUE	U	940
RETURN	U	950
C	U	960
140 FORMAT (1H1,4X,125(1H*),/5X,17(1H*),///60X,11HCUHMAPY NO.,13,/,5	U	970
10X, 8HTIME,13, 4H MINUTES,///5X,125(1H*),/5X,175(1H*))	U	980
150 FORMAT (////////27X,14HNUM. GENERATED,14X,11HNET VOLUME,10X,17H	U	990
1SPACE VOLUME,///)	U	1000
160 FORMAT (1H+,9X, 44LINK,13,16X,15)	U	1010
170 FORMAT (1H+,26X,15)	U	1020
180 FORMAT (1H+,74X,15)	U	1030
190 FORMAT (1H)	U	1040
400 FORMAT (1H0,////////53X, 7HAVRAGE,4X, 1HTOTAL,6X, 7HAVRAGE,4X,	U	1050
16HTRAVFL,11X, 5HTOTAL,6X, 5HTOTAL,5X, 1HTOTAL,6X, 7HAVRAGE,6	U	1060
7X, 5HDELAY,6X, 5HDELAY,5X, 6HTRAVFL,6X, 6HTRAVFL,6X, 4HTIME,5	U	1070
7X, 7HAVRAGE,11X, 8HVEHICLES,4X, 7HVEHICLE,4X, 7HDELAY,7X, 5	U	1080
4HDELAY,5X, 9HSTD. DEV.,4X, 5HSEC./,6X, 4HTIME,7X, 4HTIME,6X,	U	1090
5HSEC./,4X, 5HSPEC0,11X, 9HPROCESS0,4X, 5HFILES,4X, 8HVEH.SF	U	1100
6C.,7X, 9HSEC./VEH.,7X, 9HSEC./VEH.,3X, 7HVEH.MI.,7X, 8HVEH.SEC	U	1110
7.,3X, 9HSEC./VEH.,7X, 7HVEH.MI.,5X, 3HMPH,11X)	U	1120
410 FORMAT (11X,15,4X,F8.3,2Y,F7.0,5X,F8.1,5X,F7.1,7X,F8.1,3X,F7.0,4X,	U	1130
1F7.2,4X,F7.2,5X,F5.2)	U	1140
420 FORMAT (1H+,7HOVERALL,11)	U	1150
430 FORMAT (1H+,4HLINK,13)	U	1160
440 FORMAT (1H+,5X,5(1H*), 19H TOTAL CARS, LINK,14, 5H = 0,5X,5(1H	U	1170
1*))	U	1180
450 FORMAT (1H+,4X,5(1H*), 22H CARS IN QUEUF, LINK,14, 5H = 0,5X,5	U	1190
1(1H*))	U	1200
C	U	1210
END	U	1220
SUBROUTINE JBITH (X,M,N,Y)	V	10
COMMON /MSK/ MASK(20)	V	20
INTEGER X1,X,Y,Y1	V	30
X1=ISHFT(X,M).AND.MASK(1)	V	40
X1=ISHFTL(X1,M)	V	50
Y1=Y.AND.MASK(1)	V	60
X=ISHFTL(Y1,M)+X-X1	V	70
RETURN	V	80
C	V	90
END	V	100
SUBROUTINE STIX (TABLE,VMAX,VMIN,NINK,VAR,ITITLE,LNHTL,LINK)	W	10
COMMON /PLOT/ PLOT,TCATF,ISW,IOVER,IUNDER,PLOTSM	W	20
DIMENSION TABLE(111)	W	30
INDEX=NINK+4	W	40
TCATF=DATE(0)	W	50
DO 100 K=1,INDEX	W	60
100 TABLE(K)=0.0	W	70

TABLE B-2, cont.

C	RETURN		W	80
			W	90
	END		W	100
C	SUBROUTINE STIX2 (TABLE,VMAX,VMIN,NINK,VAR,ITITLE,LNTHTL,LINK)		X	10
	DIMENSION ITITLE(LNTHTL), TABLE(NINK)		X	20
	TABLE(NINK+3)=TABLE(NINK+3)+VAR		X	30
	TABLE(NINK+4)=TABLE(NINK+4)+VAR*VAR		X	40
	VINK=(VMAX-VMIN)/NINK		X	50
	IF (VAR-VMIN) 100,110,110		X	60
	100 CONTINUE		X	70
	IV=0		X	80
	GO TO 120		X	90
	110 CONTINUE		X	100
	IV=(VAR-VMIN)/VINK		X	110
	IF (IV.GT.NINK) IV=NINK+1		X	120
	120 CONTINUE		X	130
	TABLE(IV+1)=TABLE(IV+1)+1		X	140
	RETURN		X	150
			X	160
	END		X	170
C	SUBROUTINE STIX1 (TABLE,VMAX,VMIN,NINK,VAR,ITITLE,LNTHTL,LINK,LANC		Y	10
	1)		Y	20
	COMMON /Y/ EPSILON,ALLOFIT		Y	30
	COMMON /QPLOTR/ QPLOT,TDATE,ISW,IOVER,IUNDFR,PLOTSW		X	40
	DIMENSION XTABLE(310), TABL(310)		Y	50
	DIMENSION ITITLE(LNTHTL), TABLE(NINK)		Y	60
	INTFCFR OUTPUT,PUNCH,CORFSZ,WORDS7,PLOTSW,MS7		Y	70
	REAL IOVER,IUNDFR		Y	80
	EQUIVALENCE (MS7,WORDS7)		Y	90
	DATA INPUT,OUTPUT,PUNCH/6,7/		Y	100
	DATA MS7/10/		Y	110
	DATA EPSILON,ALLOFIT/.0005,99.9995/		Y	120
	DO 100 I=1,310		Y	130
	XTABLE(I)=0.0		Y	140
	TABL(I)=XTABLE(I)		Y	150
	100 CONTINUE		Y	160
	SUMS=0.0		Y	170
	T=0.0		Y	180
	VINK=(VMAX-VMIN)/NINK		Y	190
	INDEX=NINK+2		Y	200
	DO 110 I=1,INDEX		Y	210
	110 T=T+TABLE(I)		Y	220
	IF (T) 120,130,120		Y	230
	120 CONTINUE		Y	240
	IUNDFR=100.0*TABLE(1)/T		Y	250
	IOVER=100.0*TABLE(INDEX)/T		Y	260
	CMEAN=TABLE(NINK+3)/T		Y	270
	STD=SQRT(ABS(TABLE(NINK+4)/T-CMEAN*CMEAN))		Y	280
	GO TO 140		Y	290
	130 CONTINUE		Y	300
	IUNDFR=0		Y	310
	IOVER=0		Y	320
	CMEAN=0.0		Y	330
	STD=0.0		Y	340
	140 CONTINUE		Y	350

TABLE B-2, cont.

	TT=TABLE(I)+TABLE(NINK+1)	Y	360
	A7=CFLOT+1.0	Y	370
	WRITE (OUTPUT,740)	Y	380
	WRITE (OUTPUT,750) LINK	Y	390
	IF (LANF.EQ.0) GO TO 150	Y	400
	WRITE (OUTPUT,760) LANE	Y	410
150	CONTINUE	Y	420
	IF (FLOTCH.NF.0) WRITE (OUTPUT,780) A2	Y	430
	INDEXT=LNTHIL/MOSZ+1	Y	440
	IF (MOD(LNTHIL,MOSZ).EQ.0) INDEXT=INDEXT-1	Y	450
	IF (INDEXT.EQ.0) GO TO 160	Y	460
	WRITE (OUTPUT,790) (ITITLE(I),I=1,INDEXT)	Y	470
160	CONTINUE	Y	480
	WRITE (OUTPUT,840) T,CMEAN,STD	Y	490
	WRITE (OUTPUT,800)	Y	500
	II=-1	Y	510
	DO 200 I=1,INDEX	Y	520
	II=II+1	Y	530
	XTABLE(II)=(TABLE(I)/T)*100.0	Y	540
	TAPL(II+1)=(I-1.5)*VINK+VMIN	Y	550
	TAPL(II)=TAPL(II+1)	Y	560
	TAPL(II-1)=TAPL(II)	Y	570
	SUMS=SUMS+XTABLE(II)	Y	580
	IF (SUMS.LT.FPSILON) IGC=TCI200	Y	590
	UL=(I-1)*VINK+VMIN	Y	600
	CUMR=100.0-SUMS	Y	610
	CMEAN1=UL/CMEAN	Y	620
	IF (STD) 170,180,190	Y	630
170	CONTINUE	Y	640
	DEVMEAN=(UL-CMEAN1)/STD	Y	650
	GO TO 190	Y	660
180	CONTINUE	Y	670
	DEVMEAN=INFIN	Y	680
190	CONTINUE	Y	690
	P2=OPLQPR(CMEAN,UL,STD)	Y	700
	WRITE (OUTPUT,850) UL,TABLE(II),XTABLE(II),SUMS,CUMR,CMEAN1,DEVMEAN	Y	710
1,PP		Y	720
	IF (I.EQ.1) WRITE (OUTPUT,290)	Y	730
	IF (I.EQ.INDEX) WRITE (OUTPUT,280)	Y	740
	IF (SUMS.GE.ALLOFIT.AND.I.LT.INDEX-1) GO TO 210	Y	750
200	CONTINUE	Y	760
	GO TO 210	Y	770
210	CONTINUE	Y	780
	WRITE (OUTPUT,270)	Y	790
	J=I+1	Y	800
	DO 220 I=J,INDEX	Y	810
	II=II+3	Y	820
	TAPL(II+1)=(I-1.5)*VINK+VMIN	Y	830
	TAPL(II)=TAPL(II+1)	Y	840
	TAPL(II-1)=TAPL(II)	Y	850
220	CONTINUE	Y	860
230	CONTINUE	Y	870
	RETURN	Y	880
C		Y	890
240	FORMAT (//,20X,164ENTRIES IN TABLE,5X,134MEAN ARGUMENT,5X,8HST	Y	900
	ANDARD,10HDEVATION,/,26X,F10.0,8X,F10.3,13X,F10.3,/)	Y	910
250	FORMAT (10X,F14.3,F15.0,6F15.3)	Y	920
260	FORMAT (//,20X,12410)	Y	930
270	FORMAT (10X,144REMAINING FREQUENCIES ARE ALL ZERO)	Y	940
280	FORMAT (24H*OVERFLOW XXXXXXXXXXXXX)	Y	950
290	FORMAT (10H*UNDERFLOW)	Y	960

TABLE B-2, cont.

700	FORMAT (1H0,1FX, 5HVALUE,7X, 8HOBSERVED,7X, 4HPER CENT,2(5X,	Y	970
	11HCUMULATIVE),7X, 8HMULTIPLE,6X, 9HDEVIATION,4X, 10HCUMULATIVE,	Y	980
	2/,30X, 9HFREQUENCY,7X, 8HOF TOTAL,5X, 10HPERCENTAGE,6X, 9HREMAI	Y	990
	INDER,8X, 7HOF MEAN,6X, 9HFROM MEAN,4X, 11HPROBABILITY,/))	Y	1000
310	FORMAT (40X, 40HTHIS DATA IS ACCOMPANIED BY GRAPH NUMBER,F10.0)	Y	1010
320	FORMAT (///20X, 4HLINK,I3)	Y	1020
330	FORMAT (///20X, 4HLANE,I3)	Y	1030
340	FORMAT (1H1)	Y	1040
C		Y	1050
	END	Y	1060

	FUNCTION QRLOPB(Z,AVG,DEV)	Z	10
	COMP=-1.0	Z	20
	IF (DEV) 110,100,110	Z	30
100	CONTINUE	Z	40
	U=0	Z	50
	GO TO 120	Z	60
110	CONTINUE	Z	70
	U=ABS(Z-AVG)/DEV	Z	80
120	CONTINUE	Z	90
	IF (U-1.6) 130,130,160	Z	100
130	TERP=U*.3989423	Z	110
	GINT=TERP	Z	120
	U=-U*U	Z	130
	R=0.0	Z	140
140	R=R+2.	Z	150
	TERP=TERM*U*(R-1.)/(R*(R+1.))	Z	160
	GINT=GINT+TERP	Z	170
	IF (ABS(TERM)-.0001) 150,150,140	Z	180
150	GINT=.5+GINT*COMP*SIGN(1.,(Z-AVG))	Z	190
	GO TO 190	Z	200
160	PHI=U	Z	210
	UCO=1.	Z	220
	Q=1.	Z	230
	R=0.	Z	240
	Q=0.	Z	250
	ALT=-1.	Z	260
170	R=R+1.	Z	270
	UCO=1./(R*UCO)	Z	280
	TERP=0	Z	290
	Q=QC	Z	300
	Q=Q*U*UCO+TERM	Z	310
	ALT=-ALT	Z	320
	TERP=ALT/(Q*Q)	Z	330
	PHI=PHI+TERM	Z	340
	IF (ABS(TERM)-.0005) 180,180,170	Z	350
180	GINT=.3989423*EXP(-.5*U*U)/PHI	Z	360
	GINT=.5+(.5-GINT)*COMP*SIGN(1.,(Z-AVG))	Z	370
190	CONTINUE	Z	380
	QRLOPB=GINT	Z	390
	RETURN	Z	400
C		Z	410
	END	Z	420

SUBROUTINE STIX4 (TABLE1,TABLE2,VAR,NCALL,VMEAN,STDEV)	AA	10
TABLE1=0.	AA	20
TABLE2=0.	AA	30

TABLE B-2, cont.

F	N*ALL=0	AA	40
	RETURN	AA	50
		AA	60
	END	AA	70
F	SUBROUTINE STI*E (TABLE1,TABLE2,VAR,NCALL,VMEAN,STDEV)	AB	10
	TABLE1=TABLE1+VAR	AB	20
	TABLE2=TABLE2+VAR*VAR	AB	30
	NCALL=NCALL+1	AB	40
	RETURN	AB	50
		AB	60
	END	AB	70
C	SUBROUTINE STI*E (TABLE1,TABLE2,VAR,NCALL,VMEAN,STDEV)	AC	10
	T=FLCAT(NCALL)	AC	20
	IF (T,LC,0.) GO TO 100	AC	30
	VMEAN=TABLE1/T	AC	40
	STDEV=SQRT(APS(TABLE2/T-VMEAN*VMEAN))	AC	50
	RETURN	AC	60
	100 VMEAN=0.	AC	70
	STDEV=0.	AC	80
	ACTION	AC	90
		AC	100
	END	AC	110
C	REAL FUNCTION NORMAL(A,B)	AD	10
	DIMENSION TABLE(80)	AD	20
	COMFASION NAME(2)	AD	30
	DATA TABLE/.0050,.11994,.07967,.05952,.07926,.09871,.11791,.13683,	AD	40
	1.15542,.17364,.19146,.20984,.22575,.24215,.25804,.27317,.28914,.30	AD	50
	2274,.31579,.32894,.34174,.35714,.36413,.37491,.38491,.39435,.40120	AD	60
	7,.41149,.41924,.42547,.43319,.43943,.44520,.45057,.45741,.45994,.4	AD	70
	46407,.46784,.47128,.47441,.47725,.47982,.48214,.48427,.48610,.4877	AD	80
	58,.48928,.49061,.49180,.49288,.49379,.49461,.49574,.49698,.49851,.	AD	90
	499707,.49744,.49781,.49813,.49841,.49875,.49886,.49901,.49918,.499	AD	100
	731,.49947,.49957,.49960,.49966,.49972,.49977,.49981,.49984,.49987,	AD	110
	8,.49990,.49991,.49993,.49994,.49995,.49996/	AD	120
	SIGN=+1.	AD	130
	Z=RNDF(0)	AD	140
	IF (Z,LT,0.5) GO TO 100	AD	150
	Z=Z-.5	AD	160
	SIGN=-1.	AD	170
	100 DO 110 I=1,80	AD	180
	IF (Z,LT,TABLE(I)) GO TO 120	AD	190
	110 CONTINUE	AD	200
	120 CONTINUE	AD	210
	NORMAL=A+SIGN*P*(FLOAT(I)-RNDF(0)-1.0)*0.05	AD	220
	RETURN	AD	230
		AD	240
	END	AD	250
AE	REAL FUNCTION NEGEXP(A)	AE	10
	NEGEXP=0.75-(1.0/A-0.75)*ALOG(RNDF(0))	AE	20

TABLE B-2, cont.

C	RETURN	AE	30
	END	AE	40
		AE	50
	FUNCTION TIMEP(INP:INT)	AF	10
	DATA ISW,X,Y/0,0.0,0.0/	AF	20
	IF (ISW.NE.0) GO TO 100	AF	30
	ISW=1	AF	40
	CALL SECOND (Y)	AF	50
100	CONTINUE	AF	60
	CALL SECOND (X)	AF	70
	Z=X-Y	AF	80
	IF (INPRINT.EQ.0) GO TO 110	AF	90
	WRITE (5,170) Z	AF	100
110	CONTINUE	AF	110
	Y=X	AF	120
	TIMEP=X	AF	130
	RETURN	AF	140
C		AF	150
170	FORMAT (/, 35H ..CP TIME SINCE LAST CALL ...,F8.1, 8H SECOND	AF	160
	15,/)	AF	170
C		AF	180
	END	AF	190

FIGURE D-2.

FIGURE B-2

SIGNET Program Deck Structure

Appendix C

Traffic Flow Data

The following tables give detailed summaries of traffic characteristics during average peak hour conditions on the test network. The data is presented for each of the six 15-minute periods comprising the study period, 4:00 to 5:30 P.M. Traffic volumes, turning probabilities, and lane distributions were obtained through a system of machine and manual counts. Traffic signal settings were obtained by stop-watch.

TABLE C-1

Network Traffic Signal Settings
(All times in Seconds)

<u>Link Number</u>	<u>Indication</u>	<u>Duration</u>	<u>Cycle Length</u>	<u>Offset¹</u>
26	R	32	70	49
	G	35		
	A	3		
27	R	23	60	45
	G + GAL	7		
	G	27		
	A	3		

¹All offsets measured from beginning of red, Link 54 (State Street and Grant Street), to beginning of first phase listed.

<u>Link Number</u>	<u>Indication</u>	<u>Duration</u>	<u>Cycle Length</u>	<u>Offset¹</u>
28	R	19	60	26
	G + GAL	9		
	G	29		
	A	3		
29	G	33	60	53
	A	3		
	R	24		
30	R	36	65	31
	G + GAL	5		
	G	21		
	A	3		
31	G	20	65	0
	A	3		
	R	42		
32	G	55	55	0
33	R	20	55	39
	G	32		
	A	3		
34	G	55	55	0
35	G	29	60	17
	A	4		
	R	27		
36	G	28	75	60
	A	3		
	R	44		
37	G	60	60	0
38	G	33	60	53
	A	3		
	R	24		
39	R	31	60	26
	G	26		
	A	3		
40	G	20	60	45
	A	3		
	R	37		

<u>Link Number</u>	<u>Indication</u>	<u>Duration</u>	<u>Cycle Length</u>	<u>Offset</u>
41	R (All red) G A R	2 15 3 50	70	49
42	R G A	25 22 3	50	24
43	R G A	36 21 3	60	53
44	G	55	55	0
45	G	55	55	0
46	R G A	20 32 3	55	39
47	G A R	20 3 42	65	0
48	R G A	33 24 3	60	17
49	R G A	33 24 3	60	17
50	R G A	20 42 3	65	64
51	R G A	32 31 3	65	40
52	R G A	23 38 4	65	53

<u>Link Number</u>	<u>Indication</u>	<u>Duration</u>	<u>Cycle Length</u>	<u>Offset</u>
53	R G A	35 27 3	65	26
54	R G A	36 26 3	65	0
55	R GAR G A	31 30 11 3	75	60
56	R G + GAL G A	23 10 3 29	65	0
57	R G + GAL G A	20 12 30 3	65	26
58	GAL A GAS A	20 3 38 4	65	53
59	R G A	32 30 3	65	40
60	G A R	17 3 45	65	64
61	R G A	20 42 3	65	64
62	G A R	17 3 45	65	64

<u>Link Number</u>	<u>Indication</u>	<u>Duration</u>	<u>Cycle Length</u>	<u>Offset</u>
63	G A R	29 3 33	65	40
64	G A R	29 3 33	65	40
65	G A R	17 3 45	65	26
66	G A R	17 3 45	65	26
67	G + GAL G A R	8 35 3 29	65	31
68	R G A	41 21 3	65	31
69	R G A R	8 25 3 29	65	31
70	GADR GAL + GADR A R	31 27 3 14	75	60
71	R G A	61 11 3	75	60
72	G A R	17 3 35	55	39
73	G A R	17 3 35	55	39

<u>Link Number</u>	<u>Indication</u>	<u>Duration</u>	<u>Cycle Length</u>	<u>Offset</u>
74	R	55	55	0
75	G	30	60	17
	A	3		
	R	27		
76	R	33	60	17
	G	24		
	A	3		
77	R	36	60	53
	G	21		
	A	3		
78	G	16	60	26
	A	3		
	R	41		
79	G	20	60	45
	A	3		
	R	37		
80	R	33	60	45
	G	24		
	A	3		
81	R	20	70	49
	G	9		
	A	3		
	R	38		
82	R	32	70	49
	G	35		
	A	3		
83	G	22	50	24
	A	3		
	R	25		
84	G	22	50	24
	A	3		
	R	25		

TABLE C-2
TABULATION OF TRAFFIC CHARACTERISTICS, AVERAGE CONDITIONS
4.00 - 4.15 PM

TRAFFIC

LINK NUMBER	FLOW RATE VPH	TURNING PROBABILITIES						LANE DISTRIBUTION		
		1	2	3	4	5	6	1	2	3
NETWORK LINKS										
26	---	.12	--	.45	.39	--	.04	.57	.43	--
27	---	.05	--	.34	.33	--	.28	.39	.33	.28
28	---	.05	--	.37	.35	--	.23	.42	.35	.23
29	---	.10	--	.46	.32	--	.12	.56	.44	--
30	---	.28	--	.64	--	--	.28	.72	.28	--
31	---	.50	--	.27	.23	--	--	.77	.23	--
32	---	--	--	.53	.18	--	.29	.53	.47	--
33	---	.05	--	.50	.45	--	--	.55	.45	--
34	---	--	--	.53	.47	--	--	.53	.47	--
35	---	--	--	.53	.46	--	.01	.53	.47	--
36	---	.24	--	--	--	.74	--	.24	.76	--
37	---	--	--	.42	--	.35	.23	.42	.35	.23
38	---	.17	--	.42	.41	--	--	.59	.41	--
39	---	.04	--	.49	.37	--	.08	.55	.45	--
40	---	.43	--	.26	--	--	.31	.43	.26	.31
41	---	--	.18	.20	--	--	.62	.18	.20	.62
42	---	.15	--	.43	.24	--	.18	.58	.42	--
43	---	.27	--	.39	--	--	.34	.46	.34	--
44	---	--	--	.53	.47	--	--	.53	.47	--
45	---	--	--	.53	.35	--	.12	.53	.47	--
46	---	.27	--	.27	.46	--	--	.54	.46	--
47	---	.28	--	.35	.37	--	--	.63	.37	--
48	---	--	--	.29	--	--	.71	.29	.71	--
49	---	.23	--	.34	.39	--	--	.61	.39	--
50	---	.18	--	.39	.34	--	.09	.57	.43	--

TABLE C-2. CONT.

TRAFFIC

LINK NUMBER	FLOW RATE VPH	TURNING PROBABILITIES					LANE DISTRIBUTION			
		1	2	3	4	5	6	1	2	3
51	---	--	--	.53	.41	--	.0A	.53	.47	--
52	---	.34	--	.66	--	--	--	.34	.66	--
53	---	.01	--	.52	.47	--	--	.53	.47	--
54	---	.10	--	.33	.57	--	--	.43	.57	--
55	---	.79	--	--	.13	--	.0A	.79	.21	--
56	---	.03	--	.30	.37	--	.21	.52	.37	.21
57	---	--	--	.50	.46	--	.04	.50	.46	.04
58	---	--	--	.48	.39	--	.13	.48	.39	.13
59	---	.02	--	.51	.47	--	--	.53	.47	--

INPUT LINKS

60	216	.26	---	.11	---	---	.63	.37	.63	---
61	472	.01	---	.52	.42	---	.05	.53	.47	---
62	128	.38	---	.15	---	---	.87	.53	.47	---
63	356	.17	---	.13	---	---	.70	.17	.13	.70
64	116	.64	---	---	---	---	.36	.64	.36	---
65	52	.54	---	---	---	---	.85	.54	.46	---
66	256	.45	---	.10	---	---	.45	.45	.10	.45
67	340	.01	---	.45	---	---	.54	.46	.54	---
68	84	.84	---	.80	---	---	.16	.84	.16	---
69	140	.24	---	.69	---	---	.07	.93	.07	---
70	680	---	.25	---	---	---	.75	.25	---	.75
71	136	.22	---	.75	---	---	.03	1.00	---	---
72	180	.45	---	---	---	---	.55	.45	.55	---
73	108	.54	---	---	---	---	.46	.54	.46	---
74	0	1.00	---	---	---	---	---	1.00	---	---
75	484	.08	---	.47	.45	---	---	.55	.45	---
76	36	1.00	---	---	---	---	---	1.00	---	---

TABLE C-2. CONT.

TRAFFIC

LINK NUMBER	FLOW RATE VPH	TIPPING PROBABILITIES					LANE DISTRIBUTION		
		1	2	3	4	5	1	2	3
77	20A	.03	--	.53	--	--	.5A	.44	--
78	16A	.53	--	.19	--	--	.72	--	.28
79	21A	.09	--	.77	--	--	.8A	.18	--
80	34A	.18	--	.43	.34	--	.41	.34	.05
81	12	.67	.33	--	--	--	1.00	--	--
82	21A	.0A	--	.48	.42	--	.58	.86	--
83	5A	.12	--	.88	--	--	1.00	--	--
84	10A	--	--	.33	--	--	.33	.67	--

TABLE C-3
TABULATION OF TRAFFIC CHARACTERISTICS, AVERAGE CONDITIONS
4.15 - 4.30 PM

LINK NUMBER	TRAFFIC FLOW RATE VPH	TURNING PROBABILITIES					LANE DISTRIBUTION			
		1	2	3	4	5	6	1	2	3
		NETWORK LINKS								
26	---	.16	--	.41	.39	--	.04	.57	.43	--
27	---	.09	--	.37	.39	--	.23	.62	.35	.23
28	---	.04	--	.49	.35	--	.12	.53	.35	.12
29	---	.02	--	.51	.34	--	.13	.53	.47	--
30	---	.41	--	.37	--	--	.22	.78	.22	--
31	---	.51	--	.26	.23	--	--	.77	.23	--
32	---	--	--	.53	.18	--	.29	.53	.47	--
33	---	.05	--	.50	.85	--	--	.55	.45	--
34	---	--	--	.53	.47	--	--	.53	.47	--
35	---	--	--	.53	.46	--	.01	.53	.47	--
36	---	.24	--	--	--	.78	--	.24	.76	--
37	---	--	--	.39	--	.14	.27	.39	.34	.27
38	---	.21	--	.40	.38	--	.01	.61	.39	--
39	---	.03	--	.51	.42	--	.08	.58	.46	--
40	---	.46	--	.28	--	--	.26	.44	.28	.26
41	---	.18	.20	--	--	--	.62	.38	--	.62
42	---	.06	--	.49	.30	--	.14	.48	.43	--
43	---	.16	--	.40	--	--	.44	.56	.44	--
44	---	--	--	.53	.47	--	--	.53	.47	--
45	---	--	--	.43	.39	--	.12	.53	.47	--
46	---	.27	--	.27	.46	--	--	.54	.46	--
47	---	.32	--	.22	.46	--	--	.54	.46	--
48	---	--	--	.19	--	--	.89	.19	.89	--
49	---	.29	--	.35	.36	--	--	.64	.36	--
50	---	.18	--	.39	.34	--	.09	.57	.43	--

TABLE C-3. CONT.

LINK NUMBER	TRAFFIC FLOW RATE VPH	TURNING PROBABILITIES						LANE DISTRIBUTION		
		1	2	3	4	5	6	1	2	3
51	---	---	---	.53	.41	---	.06	.53	.47	---
52	---	.33	---	.67	---	---	---	.33	.67	---
53	---	.06	---	.49	.49	---	---	.49	.49	---
54	---	.12	---	.31	.57	---	---	.43	.57	---
55	---	.40	---	---	.13	---	.07	.40	.20	---
56	---	.03	---	.41	.42	---	.14	.41	.42	.14
57	---	---	---	.50	.45	---	.05	.50	.45	.05
58	---	---	---	.46	.38	---	.16	.46	.38	.16
59	---	.07	---	.48	.47	---	---	.53	.47	---
INPUT LINKS										
60	204	.26	---	.11	---	---	.43	.37	.63	---
61	464	.07	---	.48	.39	---	.06	.55	.45	---
62	100	.38	---	.15	---	---	.47	.53	.47	---
63	492	.17	---	.11	---	---	.72	.17	.11	.72
64	132	.68	---	---	---	---	.32	.68	.32	---
65	116	.54	---	---	---	---	.46	.54	.46	---
66	388	.45	---	.10	---	---	.45	.45	.10	.45
67	232	---	---	.61	---	---	.30	.61	.30	---
68	88	.03	---	.85	---	---	.12	.88	.12	---
69	44	.34	---	.60	---	---	.06	.94	.06	---
70	604	---	.25	---	---	---	.75	.25	---	.75
71	88	.22	---	.75	---	---	.03	1.00	---	---
72	144	.45	---	---	---	---	.55	.45	.55	---
73	112	.54	---	---	---	---	.46	.54	.46	---
74	0	1.00	---	---	---	---	.00	1.00	---	---
75	544	.07	---	.48	.45	---	---	.55	.45	---
76	40	1.00	---	---	---	---	---	1.00	---	---

TABLE C-3. CONT.

LINK NUMBER	TRAFFIC FLOW RATE VPH	TURNING PROBABILITIES						LANE DISTRIBUTION		
		1	2	3	4	5	6	1	2	3
77	276	.04	--	.49	--	--	.47	.53	.47	--
78	148	.67	--	.11	--	--	.22	.78	--	.22
79	216	.33	--	.51	--	--	.16	.84	.16	--
80	320	.21	--	.37	.38	--	.04	.58	.38	.04
81	64	.38	.07	--	--	--	.69	.31	.69	--
82	488	.04	--	.50	.37	--	.09	.44	.46	--
83	128	.17	--	.88	--	--	--	1.00	--	--
84	140	--	--	.31	--	--	.69	.31	.69	--

TABLE C-4
TABULATION OF TRAFFIC CHARACTERISTICS, AVERAGE CONDITIONS
4.30 - 4.45 PM

LINK NUMBER	TRAFFIC FLOW RATE VPM	TURNING PROBABILITIES					LANE DISTRIBUTION		
		1	2	3	4	5	1	2	3
NETWORK LINKS-									
26	---	.17	---	.42	.30	---	.50	.41	---
27	---	.02	---	.51	.17	---	.53	.17	.30
28	---	.06	---	.48	.34	---	.52	.34	.12
29	---	.08	---	.68	.38	---	.54	.44	---
30	---	.75	---	.30	---	---	.25	.35	---
31	---	.56	---	.23	.23	---	.77	.23	---
32	---	---	---	.53	.18	---	.53	.47	---
33	---	.05	---	.54	.30	---	.81	.30	---
34	---	---	---	.53	.47	---	.53	.47	---
35	---	---	---	.53	.44	---	.53	.47	---
36	---	.11	---	---	---	.80	.11	.80	---
37	---	---	---	.34	---	.34	.34	.34	.27
38	---	.14	---	.43	.41	---	.50	.41	---
39	---	.03	---	.52	.41	---	.55	.45	---
40	---	.40	---	.25	---	---	.40	.25	.24
41	---	---	.52	.30	---	---	.52	.30	.18
42	---	.12	---	.48	.27	---	.87	.43	---
43	---	.10	---	.44	---	---	.54	.44	---
44	---	---	---	.53	.47	---	.53	.47	---
45	---	---	---	.53	.24	---	.53	.47	---
46	---	.12	---	.30	.40	---	.51	.40	---
47	---	.30	---	.34	.34	---	.44	.34	---
48	---	---	---	.15	---	---	.15	.85	---
49	---	.34	---	.31	.34	---	.44	.34	---
50	---	.14	---	.42	.37	---	.57	.42	---

TABLE C-4. CONT.

TRAFFIC										
LINK NUMBER	FLOW RATE VPH	TURNING PROBABILITIES						LANE DISTRIBUTION		
		1	2	3	4	5	6	1	2	3
51	---	---	---	.50	.46	---	.04	.50	.50	---
52	---	.29	---	.71	---	---	---	.29	.71	---
53	---	.04	---	.50	.46	---	---	.54	.46	---
54	---	.04	---	.49	.46	---	---	.55	.45	---
55	---	.41	---	.10	---	---	.09	.01	.09	---
56	---	.03	---	.43	.40	---	.14	.84	.40	.14
57	---	---	---	.51	.45	---	.04	.51	.45	.04
58	---	---	---	.84	.73	---	.11	.54	.33	.11
59	---	.07	---	.46	.47	---	---	.53	.47	---
INPUT LINKS										
60	144	.22	---	.09	---	---	.40	.31	.69	---
61	304	.01	---	.82	.42	---	.05	.53	.47	---
62	100	.31	---	.82	---	---	.27	.73	.27	---
63	364	.14	---	.19	---	---	.63	.19	.19	.43
64	172	.60	---	---	---	---	.40	.40	.40	---
65	100	.52	---	---	---	---	.48	.52	.68	---
66	284	.32	---	.04	---	---	.44	.32	.04	.64
67	272	---	---	.48	---	---	.52	.48	.52	---
68	128	.07	---	.76	.07	---	.10	.83	.17	---
69	172	.40	---	.54	---	---	.04	.94	.04	---
70	680	.01	.33	---	---	---	.64	.34	---	.64
71	180	.16	---	.84	---	---	---	1.00	---	---
72	240	.60	---	---	---	---	.40	.60	.40	---
73	180	.44	---	---	---	---	.54	.44	.54	---
74	0	---	---	1.00	---	---	---	1.00	---	---
75	612	.07	---	.48	.45	---	---	.55	.45	---
76	36	1.00	---	---	---	---	---	1.00	---	---

TABLE C-4. CONT.

TRAFFIC

LINK NUMBER	FLOW RATE VPH	TURNING PROBABILITIES					LANE DISTRIBUTION			
		1	2	3	4	5	6	1	2	3
77	280	.04	--	.42	--	--	.54	.44	.54	--
78	300	.67	--	.11	--	--	.22	.70	--	.22
79	230	.24	--	.43	--	--	.11	.00	.11	--
80	444	.11	--	.45	.41	--	.03	.54	.41	.03
81	400	.62	.15	--	--	--	.23	.77	.23	--
82	300	.06	--	.50	.42	--	.06	.54	.46	--
83	800	.30	--	.70	--	--	--	1.00	--	--
84	104	--	--	.17	--	--	.83	.17	.83	--

TABLE C-5
TABULATION OF TRAFFIC CHARACTERISTICS, AVERAGE CONDITIONS
4.45 - 5.00 PM

LINK NUMBER	TRAFFIC FLOW RATE VPH	TURNING PROBABILITIES					LANE DISTRIBUTION			
		1	2	3	4	5	6	1	2	3
NETWORK LINKS										
26	---	.11	--	.46	.42	--	.01	.57	.43	--
27	---	.06	--	.49	.26	--	.10	.55	.26	.19
28	---	.01	--	.40	.33	--	.17	.50	.33	.17
29	---	.06	--	.49	.31	--	.14	.55	.45	--
30	---	.27	--	.48	--	--	.25	.75	.25	--
31	---	.39	--	.38	.23	--	--	.77	.23	--
32	---	--	--	.53	.20	--	.27	.53	.47	--
33	---	.10	--	.41	.39	--	--	.61	.39	--
34	---	--	--	.53	.47	--	--	.53	.47	--
35	---	--	--	.53	.45	--	.02	.53	.47	--
36	---	.25	--	--	--	.78	--	.25	.75	--
37	---	--	--	.33	--	.34	.28	.33	.39	.28
38	---	.17	--	.42	.41	--	--	.59	.41	--
39	---	.06	--	.51	.43	--	--	.57	.43	--
40	---	.46	--	.26	--	--	.28	.46	.26	.28
41	---	--	.53	.31	--	--	.16	.53	.31	.16
42	---	.08	--	.48	.35	--	.09	.56	.44	--
43	---	.29	--	.40	--	--	.31	.69	.31	--
44	---	--	--	.53	.47	--	--	.53	.47	--
45	---	--	--	.53	.29	--	.18	.53	.47	--
46	---	.06	--	.45	.49	--	--	.51	.49	--
47	---	.36	--	.31	.33	--	--	.67	.33	--
48	---	--	--	.20	--	--	.88	.20	.80	--
49	---	.36	--	.30	.34	--	--	.66	.34	--
50	---	.15	--	.42	.34	--	.09	.57	.43	--

TABLE C-5. CONT.

TRAFFIC													
LINK NUMBER	FLOW RATE VPH	TURNING PROBABILITIES						LANE DISTRIBUTION					
		1	2	3	4	5	6	1	2	3			
51	---	---	---	.52	.46	---	.02	.52	.48	---			
52	---	.40	---	.60	---	---	---	.40	.60	---			
53	---	.08	---	.47	.45	---	---	.58	.45	---			
54	---	.04	---	.50	.46	---	---	.54	.46	---			
55	---	.02	---	.10	---	---	.08	.92	.08	---			
56	---	.03	---	.43	.39	---	.15	.46	.39	.15			
57	---	---	---	.49	.44	---	.07	.49	.44	.07			
58	---	---	---	.56	.37	---	.07	.56	.37	.07			
59	---	.01	---	.52	.47	---	---	.52	.47	---			
INPUT LINKS													
60	200	.19	---	.31	---	---	.50	.50	.50	---			
61	536	.01	---	.52	.44	---	.03	.52	.47	---			
62	192	.50	---	.20	---	---	.30	.70	.30	---			
63	348	.04	---	.22	---	---	.70	.08	.22	.70			
64	312	.63	---	---	---	---	.37	.42	.37	---			
65	128	.47	---	---	---	---	.52	.67	.52	---			
66	372	.45	---	.07	---	---	.48	.45	.07	.48			
67	340	.06	---	.52	---	---	.42	.58	.42	---			
68	172	.08	---	.08	---	---	.11	.80	.11	---			
69	148	.41	---	.42	---	---	.17	.42	.17	---			
70	744	.01	.28	---	---	---	.71	.20	---	.71			
71	112	.22	---	.78	---	---	---	1.00	---	---			
72	276	.56	---	---	---	---	.44	.56	.44	---			
73	324	.40	---	---	---	---	.60	.40	.60	---			
74	0	---	---	1.00	---	---	---	1.00	---	---			
75	612	.06	---	.50	.46	---	---	.50	.46	---			
76	52	1.00	---	---	---	---	---	1.00	---	---			

TABLE C-5. CONT.

LINK NUMBER	TRAFFIC FLOW RATE VPH	TURNING PROBABILITIES						LANE DISTRIBUTION		
		1	2	3	4	5	6	1	2	3
77	340	.02	--	.56	--	--	.42	.58	.42	--
78	580	.65	--	.12	--	--	.23	.77	--	.23
79	166	.15	--	.80	--	--	.05	.95	.05	--
80	460	.20	--	.40	.36	--	.04	.60	.36	.04
81	32	.24	.14	--	--	--	.57	.43	.57	--
82	302	.01	--	.52	.43	--	--	.53	.47	--
83	148	.20	--	.80	--	--	--	1.00	--	--
84	188	--	--	.37	--	--	.63	.37	.63	--

TABULATION OF TRAFFIC CHARACTERISTICS, AVERAGE CONDITIONS
 C.00 - 5.15 PM

LINK NUMBER	TRAFFIC FLOW RATE VPH	TURNING PROBABILITIES					LANE DISTRIBUTION		
		1	2	3	4	5	1	2	3
NETWORK LINKS									
26	---	.17	--	.42	.41	--	.59	.41	--
27	---	.04	--	.40	.27	--	.44	.27	.10
28	---	--	--	.50	.34	--	.50	.34	.14
29	---	.04	--	.49	.34	--	.55	.45	--
30	---	.33	--	.38	--	--	.71	.29	--
31	---	.47	--	.30	.23	--	.77	.23	--
32	---	--	--	.53	.06	--	.53	.47	--
33	---	.02	--	.50	.30	--	.81	.30	--
34	---	--	--	.53	.47	--	.53	.47	--
35	---	--	--	.53	.46	--	.53	.47	--
36	---	.14	--	--	--	.84	.14	.84	--
37	---	--	--	.34	--	.30	.34	.34	.24
38	---	.14	--	.41	.41	--	.50	.41	--
39	---	.04	--	.53	.47	--	.57	.43	--
40	---	.35	--	.20	--	--	.35	.20	.34
41	---	--	.44	.37	--	--	.44	.37	.15
42	---	.12	--	.45	.37	--	.57	.43	--
43	---	.15	--	.41	--	--	.44	.34	--
44	---	--	--	.53	.47	--	.53	.47	--
45	---	--	--	.53	.32	--	.53	.47	--
46	---	.07	--	.44	.49	--	.51	.49	--
47	---	.27	--	.34	.39	--	.41	.39	--
48	---	--	--	.24	--	--	.24	.76	--
49	---	.43	--	.24	.31	--	.49	.31	--
50	---	.07	--	.50	.34	--	.57	.43	--

TABLE C-6. CONT.

LINK NUMBER	TRAFFIC FLOW RATE VPH	TURNING PROBABILITIES						LANE DISTRIBUTION		
		1	2	3	4	5	6	1	2	3
51	---	---	---	.50	.45	---	.05	.50	.50	---
52	---	.34	---	.44	---	---	---	.34	.66	---
53	---	.04	---	.50	.46	---	---	.54	.46	---
54	---	.12	---	.45	.43	---	---	.57	.43	---
55	---	.76	---	.15	---	---	.09	.91	.09	---
56	---	.03	---	.44	.41	---	.12	.47	.41	.12
57	---	---	---	.48	.44	---	.08	.48	.44	.08
58	---	---	---	.56	.27	---	.17	.56	.27	.17
59	---	.06	---	.50	.44	---	---	.56	.44	---
INPUT LINKS										
60	232	.16	---	.29	---	---	.55	.45	.55	---
61	408	.03	---	.51	.42	---	.04	.54	.46	---
62	252	.23	---	.22	---	---	.55	.45	.55	---
63	568	.21	---	.24	---	---	.55	.21	.24	.55
64	396	.64	---	---	---	---	.36	.64	.36	---
65	240	.55	---	---	---	---	.45	.55	.45	---
66	672	.44	---	.11	---	---	.45	.44	.11	.45
67	384	.02	---	.37	---	---	.61	.39	.61	---
68	228	.12	---	.82	---	---	.06	.94	.06	---
69	136	.20	---	.57	---	---	.23	.77	.23	---
70	628	---	.30	---	---	---	.70	.30	---	.70
71	100	.23	---	.77	---	---	---	1.00	---	---
72	212	.61	---	---	---	---	.39	.61	.39	---
73	332	.42	---	---	---	---	.58	.42	.58	---
74	0	---	---	1.00	---	---	---	1.00	---	---
75	744	.04	---	.50	.46	---	---	.54	.46	---
76	24	1.00	---	---	---	---	---	1.00	---	---

TABLE C-6. (CONT.)

TRAFFIC										
LINK NUMBER	FLOW RATE VPH	TURNING PROBABILITIES					LANE DISTRIBUTION			
		1	2	3	4	5	6	1	2	3
77	356	.03	--	.45	--	--	.52	.48	.52	--
78	628	.54	--	.14	--	--	.32	.48	--	.32
79	272	.30	--	.58	--	--	.12	.48	.12	--
80	412	.22	--	.40	.35	--	.03	.62	.35	.03
81	52	.47	.26	--	--	--	.27	.73	.27	--
82	380	.05	.45	.51	.43	--	.06	.51	.49	--
83	204	.04	--	.96	--	--	--	1.00	--	--
84	116	--	--	.42	--	--	.58	.42	.58	--

TABLE C-7
TABULATION OF TRAFFIC CHARACTERISTICS, AVERAGE CONDITIONS
5.15 - 5.30 PM

LINK NUMBER	TRAFFIC FLOW RATE VPH	TURNING PROPORTION						LANE DISTRIBUTION		
		1 2 3 4 5 6						1 2 3		
		NETWORK LINKS								
26	---	.16	--	.43	.41	--	--	.59	.41	--
27	---	.04	--	.50	.27	--	.19	.58	.27	.19
28	---	.02	--	.48	.34	--	.16	.50	.34	.16
29	---	.04	--	.47	.32	--	.12	.56	.44	--
30	---	.34	--	.44	--	--	.23	.78	.22	--
31	---	.56	--	.21	.23	--	--	.77	.23	--
32	---	--	--	.53	.13	--	.34	.53	.47	--
33	---	.01	--	.60	.39	--	--	.61	.39	--
34	---	--	--	.53	.47	--	--	.53	.47	--
35	---	--	--	.53	.45	--	.02	.53	.47	--
36	---	.24	--	--	--	.76	--	.24	.76	--
37	---	--	--	.29	--	.41	.30	.29	.41	.30
38	---	.12	--	.45	.42	--	.01	.57	.43	--
39	---	.09	--	.47	.41	--	.03	.56	.44	--
40	---	.35	--	.29	--	--	.36	.35	.29	.36
41	---	--	.47	.32	--	--	.21	.47	.32	.21
42	---	.12	--	.48	.31	--	.12	.47	.43	--
43	---	.12	--	.55	--	--	.33	.67	.33	--
44	---	--	--	.53	.47	--	--	.53	.47	--
45	---	--	--	.53	.34	--	.13	.53	.47	--
46	---	.07	--	.44	.49	--	--	.51	.49	--
47	---	.30	--	.34	.36	--	--	.64	.36	--
48	---	--	--	.29	--	--	.71	.29	.71	--
49	---	.19	--	.41	.40	--	--	.60	.40	--
50	---	.13	--	.44	.39	--	.04	.57	.43	--

TABLE C-7. CONT.

TRAFFIC		TURNING PROBABILITIES										LANE DISTRIBUTION		
LINK NUMBER	FLOW RATE VPH	1	2	3	4	5	6	1	2	3	1	2	3	
51	---	---	---	.50	.45	---	.05	.50	.50	---	.50	.50	---	
52	---	.36	---	.44	---	---	---	.34	.64	---	.34	.64	---	
53	---	.09	---	.89	.86	---	---	.54	.44	---	.54	.44	---	
54	---	.06	---	.49	.45	---	---	.55	.45	---	.55	.45	---	
55	---	.74	---	.11	---	---	.11	.80	.11	---	.80	.11	---	
56	---	.08	---	.39	.38	---	.18	.44	.38	.18	.44	.38	.18	
57	---	---	---	.49	.44	---	.07	.40	.44	.07	.40	.44	.07	
58	---	---	---	.56	.32	---	.12	.54	.32	.12	.54	.32	.12	
59	---	.03	---	.50	.47	---	---	.53	.47	---	.53	.47	---	
INPUT LINKS														
60	152	.31	---	.19	---	---	.50	.50	.50	---	.50	.50	---	
61	444	.04	---	.50	.42	---	.04	.54	.46	---	.54	.46	---	
62	184	.27	---	.29	---	---	.49	.51	.49	---	.51	.49	---	
63	424	.17	---	.16	---	---	.67	.17	.16	.67	.17	.16	.67	
64	232	.52	---	---	---	---	.48	.52	.48	---	.52	.48	---	
65	180	.77	---	---	---	---	.23	.77	.23	---	.77	.23	---	
66	408	.24	---	.06	---	---	.66	.24	.06	.66	.24	.06	.66	
67	340	.04	---	.37	---	---	.59	.41	.59	---	.41	.59	---	
68	160	.12	---	.84	---	---	.04	.96	.04	---	.96	.04	---	
69	220	.43	---	.50	---	---	.07	.93	.07	---	.93	.07	---	
70	796	---	.35	---	---	---	.65	.35	---	.65	.35	---	.65	
71	108	.20	---	.80	---	---	---	1.00	---	---	1.00	---	---	
72	102	.53	---	---	---	---	.47	.53	.47	---	.53	.47	---	
73	224	.41	---	---	---	---	.59	.41	.59	---	.41	.59	---	
74	0	---	---	1.00	---	---	---	1.00	---	---	1.00	---	---	
75	472	.04	---	.50	.46	---	---	.54	.46	---	.54	.46	---	
76	24	1.00	---	---	---	---	---	1.00	---	---	1.00	---	---	

TABLE C-7. CONT.

TRAFFIC

LINK NUMBER	FLOW RATE VPH	TURNING PROBABILITIES					LANE DISTRIBUTION			
		1	2	3	4	5	6	1	2	3
77	336	.06	--	.57	--	--	.37	.63	.37	--
78	400	.57	--	.12	--	--	.31	.60	--	.31
79	252	.30	--	.58	--	--	.12	.88	.12	--
80	388	.22	--	.40	.35	--	.03	.62	.35	.03
81	64	.33	.40	--	--	--	.27	.73	.27	--
82	320	.03	--	.51	.42	--	.04	.54	.46	--
83	156	.34	--	.66	--	--	--	1.00	--	--
84	168	--	--	.32	--	--	.68	.32	.68	--