

**Economic Evaluation of Toll Plaza Operations.**

by

Francisco J. Torres

Thesis submitted to the Faculty of the  
Virginia Polytechnic Institute and State University  
in partial fulfillment of the requirements for the degree of  
Master of Science  
in  
Civil Engineering (Transportation)

APPROVED:

---

S. Ardekani

---

D.R. Drew

---

A. Trani

October 21, 1988.

Blacksburg, Virginia

# **ECONOMIC EVALUATION OF TOLL PLAZA OPERATIONS**

by

Francisco J. Torres Verdin

S. A. Ardekani, Chairman

Civil Engineering

(ABSTRACT)

The charging systems presently being used in toll facilities impose discomfort and inconvenience to patrons, in terms of delay and stops.

This report presents an analytical method to estimate delay and stops in toll plazas. These variables are then utilized to evaluate user costs and toxic gases emitted in toll plazas. A large amount of information on highway user costs was collected and synthesized to be incorporated in this economic evaluation.

Recommendations on the amount of toll that should be charged to take into account user costs induced by toll plazas are given in this report.

## **Acknowledgements**

I would like to express my gratitude to the National Council of Science and Technology of Mexico (CONACYT) for their financial aid which allowed me to pursue a Master's Degree in the United States.

I express my deepest appreciation to Dr. Siamak A. Ardekani for the continuous guidance and endless patience received from him in the preparation of this thesis. I also would like to thank him and his wife for the invaluable support and help that they provided me during my stay in Blacksburg.

I thank Dr. D.R. Drew and Dr. A. Trani for the help and suggestions given towards the completion of this thesis.

I express my gratefulness to my parents, brothers and sisters, who have always been my major source of encouragement and support.

Finally, I thank my friends in Blacksburg, who made of my stay here one of the most vivid experiences of my life.

# Table of Contents

<b>1.0 Introduction.</b> . . . . .	<b>1</b>
1.1 Objective. . . . .	2
1.2 Methodology. . . . .	3
<b>2.0 Economic Analysis.</b> . . . . .	<b>4</b>
2.1 Toll Collection Systems. . . . .	5
2.2 Toll System Capacity. . . . .	5
2.3 Delay Analysis. . . . .	9
2.3.1 Stopped Delay. . . . .	9
2.3.2 Stopping Delay. . . . .	11
2.4 User Cost Models. . . . .	12
2.4.1 Fuel Consumption. . . . .	12
2.4.2 Oil Consumption . . . . .	13
2.4.3 Tire Wear . . . . .	14
2.4.4 Depreciation . . . . .	14
2.4.5 Maintenance and Repair. . . . .	15
2.4.6 Value of Time . . . . .	16

2.4.7 Accident Cost .....	18
2.5 Toxic Emissions Models. ....	21
2.6 Revenue Estimation. ....	22
2.7 User Cost Evaluation Methodology. ....	23
2.8 Break-even toll. ....	24
<b>3.0 Software Description. ....</b>	<b>27</b>
3.1 Main Menu Module. ....	28
3.2 Edit Data Module. ....	28
3.3 Start a New Problem Module. ....	32
3.4 Files Management Module. ....	35
3.5 Load File Module. ....	35
3.6 Save File Module. ....	38
3.7 Perform Calculations Module. ....	38
<b>4.0 Study of Toll Booth Operations. ....</b>	<b>41</b>
4.1 Sample Application. ....	41
4.1.1 Data. ....	41
4.1.2 Results. ....	43
4.1.3 Graphical Outputs. ....	48
4.1.4 Conclusions on the case analyzed. ....	54
4.2 Cost Effectiveness of the Toll Charged. ....	54
4.2.1 Toll charged to overcome user cost. ....	54
4.2.2 Special cases. ....	56
4.2.3 Toll to overcome administration and user costs. ....	58
4.2.4 Discussion and Recommendations. ....	65
4.3 Poisson Vehicular Demand. ....	65

<b>5.0 Summary and Conclusions.</b> .....	<b>69</b>
5.1 The Economic Analysis .....	69
5.2 The Software. ....	70
<b>REFERENCES</b> .....	<b>72</b>
<b>Appendix A. Consumption and Emissions Rates</b> .....	<b>73</b>
<b>Appendix B. BOOTH Program Listing</b> .....	<b>83</b>
<b>VITA</b> .....	<b>122</b>

## List of Illustrations

Figure 1.	25
Figure 2.	29
Figure 3.	30
Figure 4.	33
Figure 5.	36
Figure 6.	37
Figure 7.	39
Figure 8.	42
Figure 9.	44
Figure 10.	46
Figure 11.	47
Figure 12.	49
Figure 13.	50
Figure 14.	51
Figure 15.	52
Figure 16.	53
Figure 17.	59
Figure 18.	60
Figure 19.	61

Figure 20.	.....	62
Figure 21.	.....	63
Figure 22.	.....	68

## List of Tables

Table 1. Toll booth service rates .....	7
Table 2. Acceleration and Deceleration rates in mph/sec .....	11
Table 3. Vehicular Unit Prices (1980 US dollars) .....	17
Table 4. Accident Costs (1975 dollars) .....	19
Table 5. Accident rates .....	20
Table 6. Default values used in BOOTH .....	34

## **1.0 Introduction.**

Nowadays, toll roads constitute a viable alternative to finance the construction of highways. Among the main advantages of toll roads, and in general of any toll facility, the following can be mentioned:

- Toll funded projects are quicker to implement than tax-supported projects as complete funding is available at the beginning, and also the projects do not necessarily have to comply with federal regulations.
- The way of financing implies that only the actual users of the facility are charged. Besides, this constitutes the most equitable and precise way of pay-as-you-go financing, since the amount of the toll is assessed on the basis of the characteristics of the vehicles and the route itself.
- Because toll facilities are managed as profit generators systems, there exists a great concern to justify to the patron the toll paid, thus resulting in higher maintenance levels.

Despite these advantages, a toll financed facility faces some drawbacks, one of these, probably the most critical, is the toll collection system. Based on how toll facilities are funded, a charging system must be implemented, and the main

component of this is the toll booth, which represents the major source of inconvenience and discomfort for the patrons.

The charging process in a toll booth forces motorists to make a complete stop while paying, preventing them from having a steady and comfortable trip. Additionally, when high vehicular congestion levels prevail, queues are formed before the toll booth, generating delay to the drivers waiting to pay. Both, delay and stops, constitute very negative effects of the toll charging system.

## **1.1 Objective.**

The main objective of this work is to measure the inconvenience that stops and delays impose on users of toll facilities. This inconvenience is estimated in terms of users costs. For the purposes of this research, the following items are considered as the sources of the cost incurred by users:

- Fuel consumption
- Oil consumption
- Tire wear
- Maintenance and repair of vehicles
- Depreciation
- Value of time
- Accidents

Besides these user cost generators, toxic emissions are estimated, thus measuring the negative effect induced by toll facilities to the environment.

Additionally, a cost-benefit analysis is carried out. In this analysis, the sum of the total user cost and the toll system operation cost is compared to the benefit represented by the amount of money collected from the patrons.

## **1.2 Methodology.**

Chapter 2 discusses the operation of toll plazas. This chapter also presents the models for the estimation of users costs, benefits, and toxic emissions.

Chapter 3 describes the features of a software developed to perform the cost-benefit analysis.

Chapter 4 presents an application of the models explained in chapter 2 as well as of the software described in chapter 3. Furthermore a criterion to determine the conditions under which it is cost-effective to charge the users a toll is set forth. Provisions such as a demand modulated toll are considered as well.

Finally, chapter 5 discusses the conclusions and results obtained from the analysis accomplished.

## **2.0 Economic Analysis.**

Like intersections, toll barriers are considered as interrupted flow facilities, where vehicles need to stop when paying for driving along a section of road, or for using a bridge or a tunnel. For the toll facility users, stops and delays in toll plazas constitute a major source of discomfort, which can be translated into a monetary value.

In this chapter, the characteristics of toll systems are shown, and the factors involved in the determination and modification of capacity in toll plazas are explained. A method to compute delay in toll barriers is described. Models used to evaluate user costs such as fuel consumption, oil consumption, tire wear, depreciation, repair, accidents, and time, as functions of delay and number of stops are presented here. Additionally, equations capable of estimating major vehicular toxic emissions are described in this chapter. Finally, the concept of break-even toll is introduced at the end of the chapter.

## **2.1 Toll Collection Systems.**

Toll collection systems are classified into closed and open systems. Closed toll collection systems are preferred in bridges and tunnels. In toll roads, this system is generally used only on intercity routes with a very limited number of access points. This system operates by the dispensing of tickets at identified entry points; then these tickets are surrendered and the toll paid when the vehicles exit the facility. Thus, the toll is assigned based on the distance traveled by each vehicle. Most toll roads with closed collection systems use a combination of main toll plazas at either end, and tolled entrances and exit ramps along the route. Under this collection system, all vehicles are forced to pay.

On the other hand, open collection systems are designed to collect tolls from a portion of the vehicles using the road. Access and exit points are not as restricted as in closed systems. Toll barriers are situated at strategically selected locations across the main roadway, allowing certain local trips to be made without charge. Also, open toll systems are cheaper to build and maintain than closed systems.

## **2.2 Toll System Capacity.**

Capacity in toll plazas depends basically on two main factors; the number of servers (toll booths), and the service rate. The capacity of a conventional toll booth can be determined as follows:

Let vehicles waiting before a toll booth be numbered 0, 1, 2, ..., where 0 is the vehicle in front of the line, 1 is the vehicle behind 0 and so on. The service position is the location of a vehicle when it pays its toll, and the waiting position is the location of the following vehicle in line.

At the time vehicle 1 leaves the service position, the headway between it and vehicle 0 ( $H$ ) is equal to the sum of the headway at the time vehicle 0 left service ( $H_0$ ) and the service time for vehicle 1 ( $S$ ), which is the time needed to pay the toll.  $H_0$  is the sum of a reaction time ( $R$ ) and a move-up time ( $M$ ). The reaction time ( $R$ ) is defined as the elapsed time between the moment vehicle 0 leaves the service position and the moment vehicle 1 begins to move into the service position. The move-up time ( $M$ ) is the time needed to drive from the waiting to the service position. In many situations, when no queue is formed before the toll booth, drivers do not actually stop before entering the service position but enter to service at a low speed, coming to a halt only if the vehicle being served takes particularly long to pay its toll, rendering reaction time almost zero. Based on this, the headway between two consecutive vehicles arriving to a toll booth, can be approximated as:

$$H = R + M + S \quad (2.1)$$

then the capacity of a toll booth is defined as:

$$C = \frac{1}{R + M + S} \quad (2.2)$$

So, besides the service rate ( $S$ ), capacity is also a function of the driver behavior ( $R$ ), and the characteristics of the vehicle ( $M$ ). For heavy vehicles, the values of  $R$  and  $S$  are bigger than for light vehicles. Also capacity depends on the degree of congestion;  $R$  is almost zero when no queues are formed. A very accurate

determination of capacity in a toll booth thus involves the consideration of factors such as vehicular composition, drivers' behavior, and degree of congestion. However, the capacity of a toll booth can be adequately approximated as a percent of the service rate.

Table 1 shows the range of service rate values for toll booths that are commonly used in the design of a toll plaza:

**Table 1. Toll booth service rates**

Manned Booth	600 - 700 vph
Exact Change Booth	700 - 800 vph
Heavy-Vehicle Booth	300 - 400 vph
Tandem Booth	900 -1000 vph

These ranges were obtained from values suggested by Wood and Hamilton(1), Hall and Daganzo(2), and Larsen(3).

The capacity of a toll plaza can be easily determined just by adding the capacity of each one of the booths, however, the distribution of vehicles before the plaza should be considered as well. Reference 1 mentions why the patronage of toll booths in outer lanes is lower than that in the central lanes, even under congested conditions.

Several measures can be taken to increase the capacity of a toll plaza. Among them, the most frequently implemented are:

**Use of tandem toll booths:** Tandem booths are used where toll plazas are constrained in width by buildings or other physical barriers. They consist of two toll takers in a single lane, both serving alternating sets of vehicles simultaneously. In

general, the capacity of a tandem toll booth is less than the equivalent capacity of two toll booths arranged in parallel.

**Use of tokens:** Although the use of tokens significantly increases the capacity of booths by optimizing the charging task, it meets reluctance from motorists who are not willing to accept the inconvenience of the advanced payment required, unless a substantial discount is offered. However, such discounts may reduce revenue dramatically.

**Widening of the toll plaza approach:** By increasing the length of the wide area before the toll plaza, a more even distribution of vehicles in the toll booths is obtained.

**Use of exact change booths:** Much time is saved when the use of exact change avoids the utilization of human operators to charge users. However, exact change booths cannot be used in closed toll systems. Also some studies have shown that it is better to have the exact change booths at the middle lanes rather than at the outer left side.

**Exemption of High Occupancy Vehicles (HOV):** Under congested conditions, high occupancy vehicles ( usually 3 or more passengers) are allowed to clear toll barriers without being charged, hence, without stopping.

**Fixing the amount of the toll:** In many cases, the amount of the toll is not set merely on the basis of the benefit-cost criterion but on the need to provide an easy and fast way to charge the users. Therefore, tolls are usually fixed to nonfractionary values, as a measure which results in an increase in the service rate of the booths.

**Sorting of vehicles:** By assigning a group of booths to one specific type of vehicles, the booth operators deal only with one size of toll, allowing an optimization in the charging process.

Additionally, the capacity of a toll plaza depends also on factors very difficult to estimate, such as, familiarity of drivers with the toll-taking system and even weather conditions. The designer of a toll plaza should attempt to take into account all the factors involved in the evaluation of the toll charging system capacity.

## **2.3 Delay Analysis.**

### **2.3.1 Stopped Delay.**

Stopped delay in toll plazas is defined as the time a vehicle waits to pay before a toll booth. A queue of vehicles is formed before a toll plaza, and delay is incurred, when for a certain period, the arrival rate of vehicles is greater than the capacity. Stopped delay and queue length for a toll plaza system are evaluated here as shown below:

The queue generated during a certain period  $\Delta t$  is computed as:

$$\Delta queue_i = (q_i - cap)\Delta t \quad (2.3)$$

Where:

- $q_i$  = vehicular demand during the time interval  $i$  in vph
- $cap$  = capacity of the toll plaza in vph

Then the queue at any time  $t_i$  is given by:

$$queue_i = \begin{cases} queue_{i-1} + \Delta queue_i & \text{if } queue_{i-1} + \Delta queue_i > 0 \\ 0 & \text{otherwise} \end{cases} \quad (2.4)$$

Then the stopped delay experienced by all the vehicles waiting in the queue between  $t_{i-1}$  and  $t_i$  is :

$$\Delta D_{Si} = \frac{queue_i + queue_{i-1}}{2} \quad (2.5)$$

Therefore the total stopped delay in the system between  $t_0$  and  $t_i$  in veh•hrs is given by:

$$D_{Si} = D_{Si-1} + \Delta D_{Si} \quad (2.6)$$

Additionally, the number of vehicles that arrive at the toll plaza between  $t_{i-1}$  and  $t_i$  is :

$$\Delta Veh_i = q_i \Delta t \quad (2.7)$$

Thus, the total number of vehicles that have arrived just before  $t_i$  is

$$Veh_i = Veh_{i-1} + \Delta Veh_i \quad (2.8)$$

Since all the vehicles arriving at the toll booths are supposed to stop, the value obtained from eq. 2.8 is equivalent to the number of stops made at  $t_i$ .

Then the average stopped delay or stopped delay per vehicle at  $t_i$  is:

$$\bar{D}_{Si} = \frac{D_{Si}}{veh_i} \quad (2.9)$$

### 2.3.2 Stopping Delay.

Besides stopped delay, the delay due to acceleration and deceleration, referred also as stopping delay, is included in this study. Stopping delay is defined as the difference in any additional time beyond the travel time if the plaza were not there. The stopping delay incurred by a vehicle approaching a toll barrier is given as:

$$D_{AC} = \frac{V}{(a + d)} \quad (2.10)$$

Where:

$D_{AC}$  = hours of delay induced to the users when stopping from a speed  $V$  and returning to it after clearing the toll barrier.

$a$  = acceleration rate in mph/hr

$d$  = deceleration rate in mph/hr

$V$  = approach speed in mph.

Pignataro (4) suggests the acceleration and deceleration values shown in Table 2.

**Table 2. Acceleration and Deceleration rates in mph/sec**

	Cars	Light Trucks	Heavy Trucks
Deceleration	10	7	7
Acceleration	5	2	1.5

## 2.4 User Cost Models.

The models presented in this section provide an adequate and reliable estimation of the most important user costs; those related to vehicle operation as well as accidents and value of time.

### 2.4.1 Fuel Consumption.

Perhaps the cost that is easiest for the user to perceive is the one associated with fuel consumption. Patrons consume additional fuel when slowing down and stopping before a toll barrier. Moreover, when a queue is generated, drivers keep the engine of their vehicles idling, thus consuming additional fuel.

Based on this, the model used here to measure the additional fuel consumption due to the presence of toll barriers is as follows:

$$FC = I_F D_S + S_F(V) veh \quad (2.11)$$

Where:

- $FC$  = fuel consumed in gallons
- $I_F$  = fuel consumed per 1000 veh·hrs
- $D_S$  = stopped delay of all the vehicles approaching the toll plaza
- $V$  = approach speed in mph
- $S_F(V)$  = fuel consumed during the deceleration-acceleration cycle from a speed  $V$ , in gal/1000 stops
- $veh$  = vehicles that approach the toll plaza

$I_r$  and  $S_r$  are taken from data obtained by Zaniewsky et al (5). These values are listed in Appendix A

Once the fuel consumed during a certain period is estimated, it is multiplied by the fuel unit price in order to obtain the fuel consumption cost.

## 2.4.2 Oil Consumption

Compared to the other vehicular operation costs, engine oil cost is the lowest. Oil consumption depends on factors, such as vehicle age, characteristics of driving, roadway alignment and length of trip. Speed changes also induce additional oil consumption.

The model used here to evaluate oil consumption, like the one presented before for fuel consumption, takes into account two parameters, idling time and number of stops. That is:

$$OC = I_o D_s + S_o(V) veh \quad (2.12)$$

Where  $OC$  = oil consumed in quarts.

$I_o$  = oil consumed per 1000 veh·hrs

$S_o(V)$  = oil consumed during the deceleration-acceleration cycle from a speed  $V$ , in quarts per 1000 stops

$D_s, V$  and  $veh$  are the same as in eq. (2.11)

$I_o$  and  $S_o$  are taken from Reference 5 and are listed in Appendix A.

Oil cost is obtained by multiplying the  $OC$  value by the cost of a quart of oil.

### 2.4.3 Tire Wear

Tire wear is mainly a function of distance traveled, driving surface, roadway geometry, speed changes, weather conditions, and speed change cycles. This last factor is the only one of relevant importance in this analysis, as it accounts for the stopping maneuver that drivers are forced to make before toll booths. Based on this, the model used to evaluate tire wear is:

$$TW = S_{TW}(V) \text{ veh} \quad (2.13)$$

Where:

$TW$  = Number of sets of tires completely worn

$S_{TW}(V)$  = Number of sets worn during the deceleration-acceleration cycle from a speed  $V$ , in sets per 1000 stops

$S_{TW}$  values are also given in Appendix A and are suggested by Reference 5.

Note that idling time does not induce tire wear. Finally, the tire wear cost is evaluated as the product of  $TW$  and the cost of a set of tires.

### 2.4.4 Depreciation

Among the running costs, depreciation is the most difficult to evaluate. There exists a direct relationship between miles traveled by a vehicle and its depreciation. Also, the age of the car is closely related to the depreciation of a vehicle. Since idling time and stops are minor causes of depreciation, the depreciation due to toll plazas

is not a considerable expense compared to the other costs discussed. The depreciation model is:

$$DP = I_{DP} D_S + S_{DP}(V) veh \quad (2.14)$$

Where:

$DP$  = Number of vehicles completely depreciated.

$I_{DP}$  = Percentage of depreciation of a vehicle per 1000 veh·hr

$S_{DP}(V)$  = Percentage of depreciation due to stopping from and resuming a speed  $V$

$I_{DP}$  and  $S_{DP}$  values are presented in Appendix A and are recommended by Reference 5.

Depreciation cost is obtained as the product of  $DP$  and the value of a new vehicle.

#### **2.4.5 Maintenance and Repair.**

Maintenance and repair expenses are difficult to evaluate, since they depend on the care given by the owner of the vehicle and the specific conditions under which the vehicle operates. Although maintenance expenses increase with the age of the vehicle, a complete repair putting the vehicle into good mechanical condition allows another long period of operation without major repair expenses. In general, the major generator of repair and maintenance expenses is the engine, followed by chassis, electrical equipment, brakes, body and power train, in that order of importance. Tire wear is not mentioned in this group since it is considered separately.

Again, for this study, maintenance and repair cost, is originated by the two major sources of inconvenience to the toll facilities user, delay and stops. The model is:

$$MR = I_{MR} D_S + S_{MR}(V) veh \quad (2.16)$$

Where:

$MR$  = Percentage of running cost per 1000 miles

$I_{MR}$  = Percentage of 1000 miles running cost per 1000 veh·hrs  
of stopped delay

$S_{MR}(V)$  = Percentage of 1000 miles running cost per 1000 stops from  
a speed  $V$

$I_{MR}$  and  $S_{MR}$  percentages as well are given in Appendix A, while the 1000 miles running costs are shown in Table 3. The values provided in Appendix A and in Table 3 are taken from Reference 5.

The maintenance and repair expense is evaluated as the product of the  $MR$  percentage and the corresponding running cost per one thousand miles of travel.

## 2.4.6 Value of Time

Perhaps the most difficult item to evaluate among the ones considered in this analysis is the cost of the time lost by the patrons when waiting before a toll booth. There exists a basic distinction in the way the cost of time is assessed for passenger cars and for trucks. Automobile travel time savings represent nonmarket costs and they are often transformed into greater spatial opportunities, such as additional time for personal business or social activities. On the other hand, for trucks, travel time savings represent market costs.

**Table 3. Vehicular Unit Prices (1980 US dollars)**

Type of Vehicle	Oil (quart)	Maintenance and Repair (Average cost per 1000 mi)	Depreciation (New Vehicle)	Fuel * (gallon)	Tires ** (set)
Small Car	2.5	34.30	6360	1.098	43
Medium Car	2.5	41.60	7501	1.098	68
Large Car	2.5	48.04	9990	1.098	75
Pickup	2.5	52.81	6625	1.098	75
2 Axles SU	1.0	99.00	8673	1.098*	194
3 Axles SU	1.0	140.00	45350	0.886*	465
2 Axles Trailer	1.0	145.00	48687	0.886*	465
3 Axles Trailer	1.0	145.00	51630	0.886*	465

\* Gas Price = 1.221 - 0.123(tax), Diesel Price = 1.01 - 0.124(tax)

\*\* Truck tire cost includes recaps

Based on this, travel time value for a passenger car is a function of trip purpose, traveler's income level, and vehicle occupancy. The Manual on User Benefit Analysis of Highway and Bus-Transit Improvements (6) suggests a one hour value of 3.9 1975 U.S. dollars for an average trip made by a passenger car. This value was obtained based on the average hourly family income prevailing in 1975 as well as the adult occupancy factor observed at that time.

For trucks, the value of the time is assumed to be equivalent to the wage rate of the driver of the vehicle. According to Reference 6, the values hourly values for single unit trucks, and combination trucks, in 1975 dollars, are 7.0 and 8.0, respectively.

The model which evaluates the time value is:

$$TV = uvt (D_S + D_{AC} \cdot veh) \quad (2.16)$$

Where:

$uvt$  = unit value of time (dollars/hour)

Note that the delay due to acceleration-deceleration cycles is also considered in the value of time evaluation.

## 2.4.7 Accident Cost

In this study, only one type of accident is considered, which is the rear-end collision. This conflict is originated by the stopping maneuver before the toll booths.

A technique developed by Brown (7) to predict accidents at an intersection is used here, since the stopping maneuvers are similar at an intersection and before a toll booth.

Brown suggests the following model for an intersection:

$$AC = R k \sqrt{X_i X_j} \quad (2.18)$$

Where:

$AC$  = Number of accidents per million vehicles

$R$  = Is a constant that depends on the type of accident

$k$  = Regression constant which accounts for variations of the vehicular flow in one year

$X_i$  and  $X_j$  = are the two conflicting movements in vph

Brown obtained a mean  $k$  value of 0.0227 with a standard deviation of 0.0017 and a mean standard error of 0.000555.

$R$  is equal to 0.7 for rear-end collisions.

$\sqrt{x_i x_j}$  is equal to the number of vehicles (veh) arriving at the toll plaza. So the number of rear-end collisions before a toll barrier can be estimated as:

$$Acc = 0.01589 \times 10^{-6} \text{ veh} \quad (2.19)$$

Accidents are divided into three major types according to their severity:

1. Fatal.
2. Injury (Nonfatal).
3. Property damage.

Table 4 shows the accident costs cited in the National Highway and Traffic Safety Administration (8).

**Table 4. Accident Costs (1975 dollars)**

Fatal	307 210
Injury	14 600
Property Damage	650

W. Meyers (6) presents accident data obtained from a survey conducted between 1976 and 1978. In this survey data from 21 toll expressways were collected. Table 5 shows the percentages of occurrence computed from values given in Reference 9.

**Table 5. Accident rates**

Class of Severity	Passenger Cars	Light Trucks	Heavy Trucks
Fatal	1.2	1.6	1.6
Injury	32.9	29.7	28.5
Property Damage	65.9	68.7	69.9

The cost per accident is then the summation of the products of cost and percentage of occurrences corresponding to the three types of accidents, as indicated below:

$$C_{Acc} = C_{Fat}P_{Fat} + C_{Inj}P_{Inj} + C_{PD}P_{PD} \quad (2.20)$$

Where:

$C_{Acc}$  = cost per accident in dollars

$C_{Fat}$  = cost of a fatal accident in dollars

$P_{Fat}$  = percentage of occurrence of a fatal accident

$C_{Inj}$  = cost of an injury accident

$P_{Inj}$  = percentage of occurrence of an injury accident

$C_{PD}$  = cost of a property damage accident

$P_{PD}$  = percentage of occurrence of a property damage accident

Then, this unit cost is multiplied by the number of accidents computed using eq. 2.19, and the accident cost is evaluated.

## 2.5 Toxic Emissions Models.

Vehicular emissions constitute a major source of environmental pollution. In this work, models to evaluate carbon monoxide, hydrocarbons, and oxides of nitrogen are presented. However, no attempt is made to estimate the cost of the environment deterioration caused by these vehicular emissions. Carbon monoxide and hydrocarbon emissions result from incomplete combustion of fuel during the operation of internal combustion engines. Oxides of nitrogen are produced when the oxygen and nitrogen in the air used by the internal-combustion engines combine under the heat and pressure of the combustion process.

The emission model for carbon monoxide is:

$$ECO = I_{CO} D_S + S_{CO}(V) \text{ veh} \quad (2.21)$$

Where:

$ECO$  = pounds of carbon monoxide emitted.

$I_{CO}$  = carbon monoxide emissions per 1000 veh·hrs

$S_{CO}(V)$  = carbon monoxide emitted per 1000 stops from a speed  $V$

Hydrocarbons emitted are evaluated according to the following model:

$$EHC = I_{HC} D_S + S_{HC}(V) \text{ veh} \quad (2.22)$$

Where:

$EHC$  = pounds of hydrocarbons emitted.

$I_{HC}$  = hydrocarbon emissions per 1000 veh·hrs

$S_{HC}(V)$  = hydrocarbons emitted per 1000 stops from speed  $V$

The model used to estimate oxide of nitrogen emissions is:

$$ENOX = I_{NOX} D_S + S_{NOX}(V) veh \quad (2.23)$$

Where:

$ENOX$  = pounds of oxide of nitrogen emitted.

$I_{NOX}$  = NOX emissions per 1000 veh•hrs

$S_{NOX}(V)$  = oxides of nitrogen emitted per 1000 stops from speed  $V$

The parameters involved in these emissions models are presented in Appendix A for several types of vehicles. The values due to stopping are taken from Reference 5. Dale(10) suggests the values given in Appendix A for emissions due to idling time.

## 2.6 Revenue Estimation.

Revenue is the amount of money collected by the toll charging system, and is computed as:

$$Rev = \left( \sum_{j=1}^m toll_j \cdot p_j \right) (veh_i - queue_i) \quad (2.24)$$

Where:

$Rev_i$  = total revenue in dollars at the end of the  $i$  time interval.

$toll_j$  = toll for a vehicle of class  $j$ .

$p_j$  = percentage of vehicles of class  $j$ .

$m$  = number of classes of vehicles considered.

$veh_t$  = vehicles that have arrived at  $t$ ,

$queue_t$  = vehicles waiting to pay at  $t$ ,

A value of revenue can be computed for every time interval, and then is compared to the value of total user cost obtained as described in the following section.

## **2.7 User Cost Evaluation Methodology.**

The first step in this methodology is the determination of the capacity for the toll charging system studied. In the analysis, capacity is assumed to be constant during the period of study. Vehicular demand data should be obtained in terms of number of vehicles approaching the toll plaza during constant time intervals.

Once capacity is estimated and vehicular demand data provided, equations 2.3 to 2.10 are used to determine queue length, delay and number of vehicles that arrived at the toll plaza. Then, for every time interval, the number of vehicles and delay values are substituted in the user cost models presented in section 2.4 . The costs obtained from these models are added together, and values of total user cost at the end of each time interval are obtained.

Since the parameters involved in the user cost models are available for eight different classes of vehicles (see Appendix A), the above-described methodology can provide extremely accurate estimations when good information about vehicular composition is available. The general model for evaluating the total user cost is:

$$UC = \sum_{i=1}^n \sum_{j=1}^m Cost_{ij} \cdot p_j \quad (2.25)$$

Where:

$UC$  = total user cost in dollars.

$Cost_{ij}$  = User cost corresponding to an item  $i$  and a vehicle class  $j$ .

$n$  = number of user costs taken into account.

$p_j$  = percentage of vehicles of class  $j$ .

$m$  = number of classes of vehicles considered.

This methodology is illustrated and synthesized in Figure 2.1

Based on the characteristics of the parameters involved and because of the amount of data needed, a computer program was developed to perform the methodology. All the models described in this chapter are incorporated in this program. In the next chapter, the characteristics and capabilities of this software are described.

## 2.8 Break-even toll.

Break-even toll is defined as the amount of toll required to overcome the user and administration costs, and is evaluated as:

$$toll_j = \frac{(UC_j + AC \cdot p_j \cdot t)}{veh_j} \quad (2.24)$$

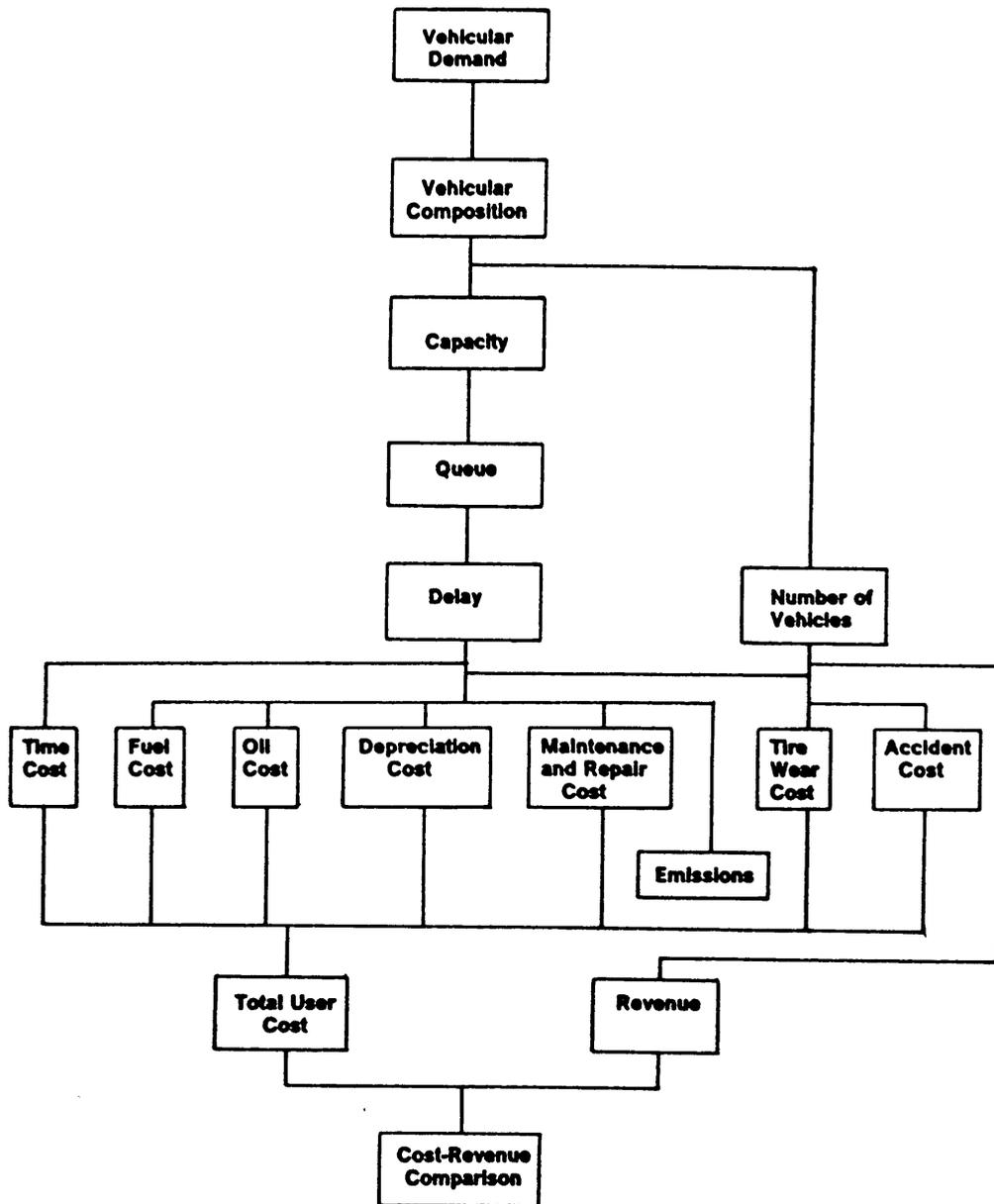


Figure 1. : User Cost Evaluation Methodology

Where:

$toll_{c_j}$  = break-even toll for a vehicle of class  $j$

$UC$  = total user cost for vehicles of class  $j$ , in dollars

$t$  = period of study in hours

$UAC$  = unit administration cost in dollars per hour

$p_j$  = percentage of vehicles of class  $j$ .

$veh_j$  = number of vehicles of class  $j$  that arrived at the toll plaza during  $t$

A further discussion on break-even toll is included in chapter 4.

## **3.0 Software Description.**

A computer program was developed to perform the methodology presented in chapter 2. The features of this program are described in this chapter. The software is written in BASIC and is named BOOTH; it is designed to be used with IBM or IBM compatible microcomputers. The listing of this program is shown in Appendix B.

The objective of BOOTH is to provide a reliable and fast estimation of the user costs and benefits that result from the operation of a toll charging system. A major concern in the development of the program was to make it easy to use. In addition, care was taken to provide results readily understandable. Some other characteristics of BOOTH include the use of color screens, graphic presentation of results, and a system of modules that allow the user to make use of all the capabilities of the program without any difficulty. In the following sections, each one of the modules contained in the program will be described.

### **3.1 Main Menu Module.**

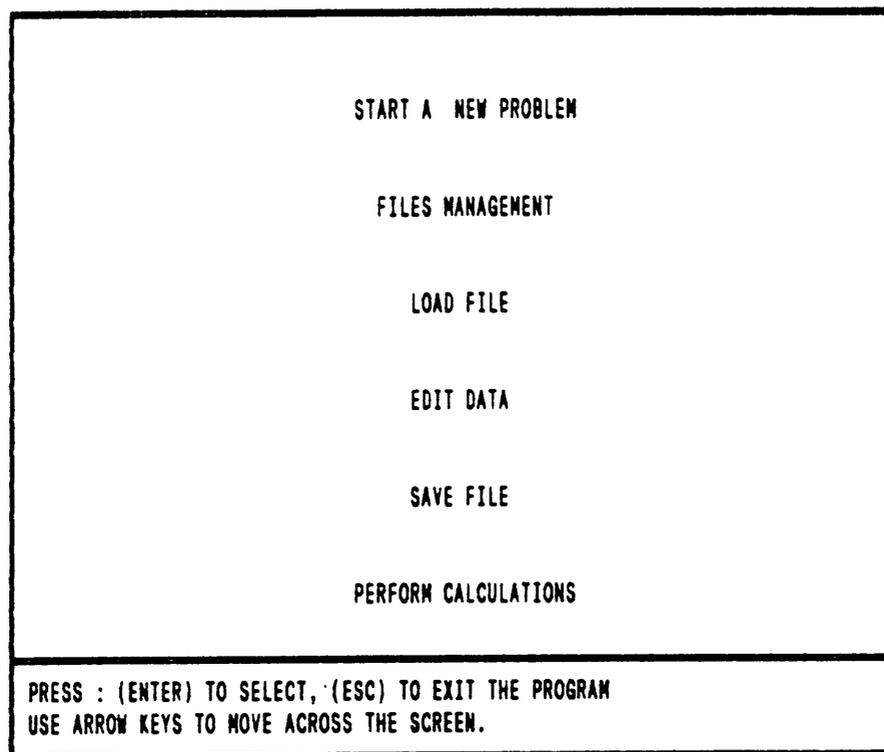
When the program is called from the DOS system, by typing BOOTH, a screen like the one shown in Figure 2 will appear. Also a highlight bar appears around one of the six choices provided by the main menu. The process of selection is made by moving the bar to the choice desired, according to the instructions located at the bottom of the screen. Each choice represents a module the features of which are discussed in the following sections.

### **3.2 Edit Data Module.**

This module was designed to simplify the task of inputting the data into the program. When selecting this module from the main menu, the screen shown in Figure 3 will appear. The process to make the selection of options is exactly the same as the one described for the Main Menu Module.

The Edit Data Module allows the user to view and modify the data simultaneously. Instructions to work in this module are provided at the bottom of the screen. The data required are:

**Period of Study:** This includes initial time, ending time, and time increment, in the hours-minutes system. The program accepts values that will produce as many as one hundred constant time intervals.



**Figure 2. : Main Menu Module**

---

PERIOD OF STUDY  
VEHICULAR DEMAND  
TOLL PLAZA CAPACITY  
VEHICULAR COMPOSITION  
VALUE OF TIME AND SPEED  
ACCIDENTS  
VEHICLE COSTS  
TOLL PLAZA OPERATION

PRESS : (ENTER) TO SELECT, (ESC) TO EXIT TO MAIN MENU  
USE ARROW KEYS TO MOVE ACROSS THE SCREEN.

Figure 3. : Edit Data Module

---

**Vehicular Demand:** The number of vehicles that approached the toll plaza during each one of the constant time intervals are introduced in this section. These values include both, light and heavy vehicles.

**Toll Plaza Capacity:** In this section, the number of booths per type and their capacity are inputted. BOOTH only takes into account two different types of toll booths: exact change and manned.

**Vehicular Composition:** BOOTH distinguishes between heavy and light vehicles. Small, medium, and large cars, as well as pickups are considered in the light vehicles group. On the other hand, two-axle and three-axle single and semitrailer units are classified as heavy vehicles. The percentage of heavy vehicles is introduced in this section. If there exists a percentage of vehicles that are exempted from toll payments (e.g. High Occupancy Vehicles), a value must be entered for this item.

**Speed and Value of Time:** In this section, the value of time for the two classes of vehicles considered must be entered. Also, approach speed, acceleration and deceleration rates for both classes of vehicles are entered in this module. The value of speed is used to determine the cost generated due to stops for items such as fuel, oil, maintenance and repair, depreciation and tire wear.

**Accident Rate and Costs:** The required data in this section are the accident rate for a rear-end collision as well as the cost for each one of the three types of accidents considered. These accident costs are assumed to be the same for the two vehicular groups; however, the percentages of occurrence for each type of accident are different for each group and are fixed values inside the program.

**Vehicle Costs:** In this section, the data required are oil cost per quart, fuel cost per gallon, cost of a new set of tires, new vehicle cost, and maintenance cost. Once again, these values will be entered for the two groups of vehicles taken into account. Table 3 in Chapter 2 shows the 1980 vehicular costs recommended by Reference 5.

**Toll Plaza Operation:** In this section, the data needed to compute toll plaza operation cost and revenue are entered, that is, operation cost per hour and the amount of toll charged for both, light and heavy vehicles.

Figure 4 shows one of the screens that is generated by BOOTH for data entry. This screen corresponds to the Vehicle Costs section. Instructions to enter the data are provided at the bottom of the screen for all the sections contained in the Edit Data module.

### **3.3 *Start a New Problem Module.***

Upon selection of this module, some of the required data values are given to the user by default, intending to suggest some values that might be difficult to obtain. These data include booth capacities, value of time, accident rate and costs, and all the vehicular running costs. The users of the program, however, are encouraged to use values that can better suit their particular needs, whenever these are available. Table 5 lists the default values provided by the program.

VEHICLE COSTS		
	LIGHT VEHICLES	HEAVY VEHICLES
LUBRICATING OIL		
Cost per quart :	4	5
TIRE WEAR		
Set of tires cost :	100	200
MAINTENANCE		
Cost per 1000 miles :	50	100
DEPRECIATION		
New Vehicle Cost :	10000	100000
FUEL		
Cost per gallon :	1	1.1

PRESS: (ENTER) TO STORE, (ESC) TO RETURN TO DATA MENU,  
 (BACKSPACE) TO CORRECT, (INS) TO STORE PREVIOUS VALUE.  
 USE ARROW KEYS TO MOVE ACROSS THE SCREEN.

Figure 4. : Input Data Screen

**Table 6. Default values used in BOUTH**

Manned Booth Service Rate	700 vph
Exact Change Booth Service Rate	800 vph
Accidents per million of vehicles	0.01589
Fatal accident cost	\$ 654 350
Non-fatal accident cost	\$ 31 100
Property damage accident cost	\$ 1 350

	Light Vehicles	Heavy Vehicles
Fuel Cost (dollars/gal)	1.01	0.89
Oil Cost (dollars/quart)	2.5	1.0
Tires (dollars/set)	105	636
1000 miles running cost (dollars)	70.7	212
New vehicles cost(dollars)	12 190	61 736
Value of time (dollars/hr)	8.3	16
Acceleration rate (mph/sec)	5	2
Deceleration rate (mph/sec)	10	7

Booth service rates are the highest values suggested in Table 1 for the two types of booths included in the analysis. The accident rate estimated by Equation 2.19. Accident cost values result from converting the values shown in Table 3 to 1988 values, assuming an inflation rate of 6 percent per year. Unit costs of oil and fuel represent the average costs for these items in 1988. The tires costs, running cost per one thousand miles, and the new vehicle costs are taken from Reference 5. Since Reference 5 provides unit cost values in 1980 dollars, an adjustment was made assuming a yearly inflation rate of six percent between 1980 and 1988. Also, the values of time for average trips proposed in Reference 6 (1975 dollars) are modified using the same inflation rate. Finally, the default acceleration and deceleration values are those recommended in Reference 4 and that are shown in Table 2.

The process of entering the data is exactly the same as for the Edit Data Module.

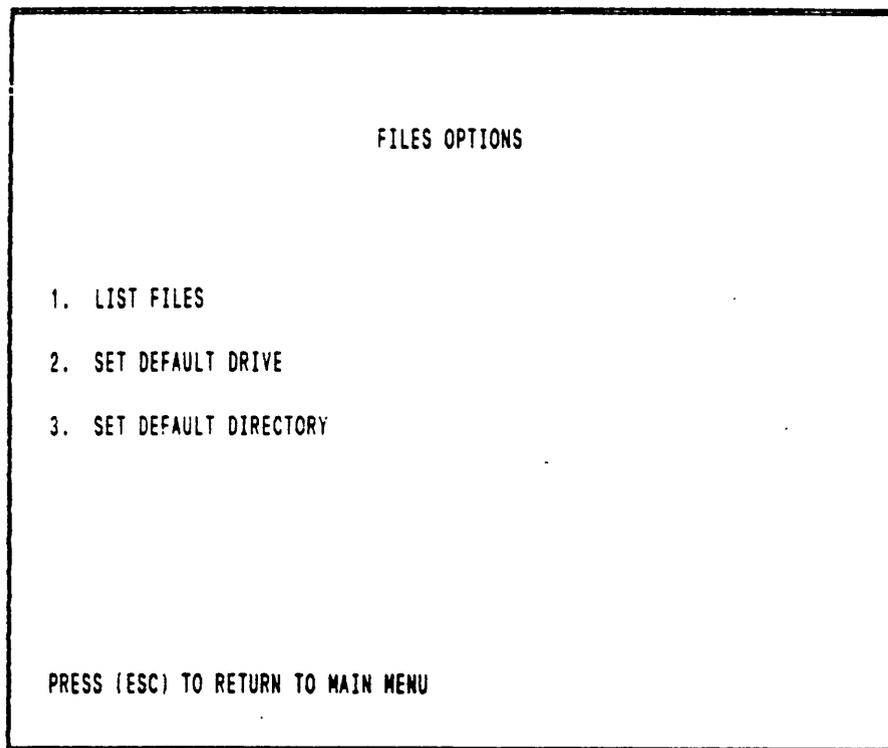
### **3.4 Files Management Module.**

This module allows the user to have full control over the data files that are created interactively through BOOTH. Three basic options are provided, as shown in Figure 5, which is the screen corresponding to this module in the program. Selecting "1. LIST FILES" produces a listing on the monitor of all the stored data files that have been previously created with the data entered by the user, these files have the extension .BTH for "Booth". The selection of "2. CHANGE DRIVE" allows the user to change the default drive designation. When the user of the program desires to keep the files created under a directory, option "3. CHANGE DIRECTORY" must be chosen. Finally, once either of the three basic options is selected, pressing the [ESC] key takes the user back to the Main Menu Module.

### **3.5 Load File Module.**

Upon the selection of this module, a screen like the one shown in Figure 6 will appear. This module allows the user to load the previously created data files. As can be observed, the user does not need to type the extension of the file name, since it is assumed to be .BTH. At the bottom of the screen, the name of a file previously loaded is shown. When no file has been loaded, a message appears indicating that.

Additionally, the program is provided with a run-time error trapping subroutine that gives the user the chance to reenter the name of the file if the one provided is incorrect either because the file cannot be found or a bad denomination was



**Figure 5. : Files Management Module**

---

```
Volume in drive A has no label
Directory of A:\

SALTILLO.BTH  MERIDA  BTH  FORTIN  BTH  DELICIAS.BTH  JAMEI  BTH
CORDOBA  BTH  ORIZABA  BTH  PUEBLA  BTH  IGUALA  BTH  TOLUCA  BTH
CELAYA  BTH  IRAPUATO  BTH  TIJUANA  BTH  ENSENADA  BTH  MAZATLAN  BTH
CULIACAN  BTH  VERACRUZ  BTH  MEXICODF  BTH
      18 File(s)  211968 bytes free

THE CURRENT FILE USED IS : SALTILLO.BTH

NAME OF THE FILE TO LOAD (EXTENSION .BTH ASSUMED )DELICIAS

PRESS ENTER TO RETURN TO MENU
```

Figure 6. : Load File Module

---

specified. Without the use of this subroutine, an immediate halt in the execution of the program would occur.

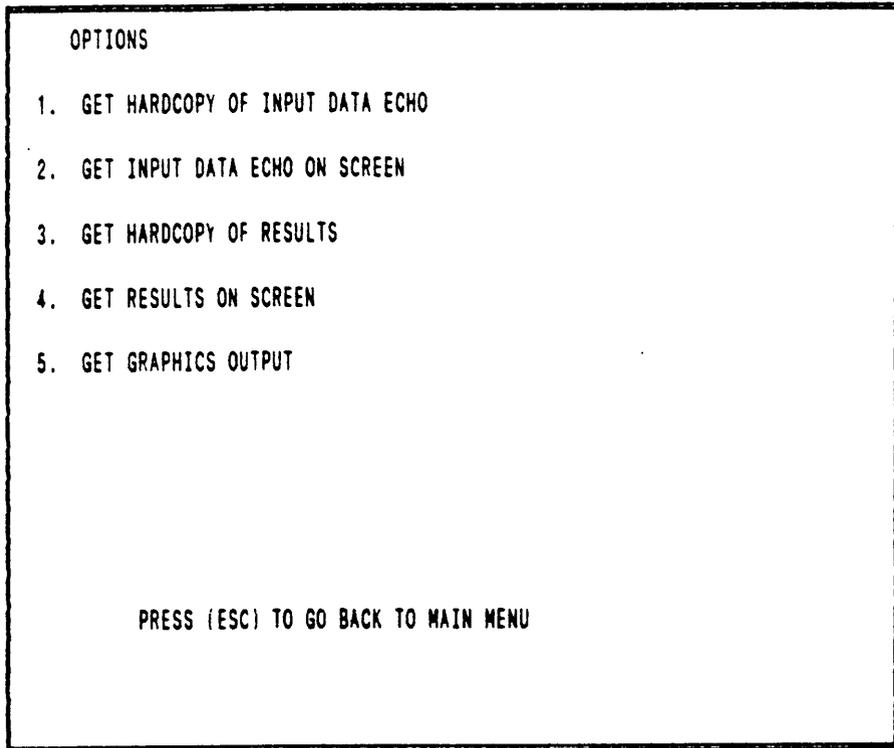
### **3.6 *Save File Module.***

Selecting the "SAVE FILE" option from the Main Menu Module will produce a screen similar to the Load File Module screen. In this case, after having entered or modified data using the Edit Data Module, the user can store the new values either in an already existing file or in a new file. In both cases, the user is required to provide the name of the file in which the new data will be stored. The same error trapping subroutine described in the last section is used here to avoid halting the execution of the program when a wrong file specification is entered.

### **3.7 *Perform Calculations Module.***

Selection of "PERFORM CALCULATIONS" option in the Main Menu produces the screen shown in Figure 7. The first two options in this module allow the user to view the data that are used to perform the calculations. These data can be seen either in an output produced by the printer (Option 1) or on the screen (Option 2). Same applies for the results (Options 3 and 4), which are provided in three different outputs.

The values of consumption rates and emissions of the models described in last chapter are taken from Reference 5 and are listed in Appendix A. For light vehicles,



**Figure 7. : Perform Calculations Module**

---

these values are the arithmetic average of the values for each class of vehicle included in this group (small, medium, and large cars as well as pickups). Same applies for the heavy vehicles group (two-axle and three-axle single and semitrailer units).

In the first output, titled "Measures of Effectiveness", values corresponding to hourly vehicular demand rate ( $q_m$ ), vehicles arrived, queue, and delay are shown for every time interval. Also, the the computed values of average delay and capacity of the toll plaza are displayed in this output. "Cost-Revenue Analysis" is the title of the second output. This output lists the accumulated values of user cost, administration cost, total cost, and gross revenue at the end of every interval. The break-even tolls for light and heavy vehicles are also given in this second output. The third output summarizes the costs incurred by the users at the end of the period of analysis, for all the items considered and for the two groups of vehicles taken into account. Values for gross revenue, net revenue, administration cost, and total user cost are provided. This worksheet also shows the final estimation of the break-even toll. Finally, the estimated values of the toxic emissions produced by the vehicles can be seen at the bottom of the output. Option 5 allows the user to view some of the results through graphs. The user of the program can return to the Main Menu by pressing the [ESC] key. A sample of the outputs obtained using this module will be shown in the next chapter, in which an example case is analyzed.

## **4.0 Study of Toll Booth Operations.**

A sample application of the User Cost Evaluation methodology presented in chapter 2 and the software described in chapter 3 is included in this chapter. Also, the factors involved in the cost effectiveness of the charging process in a toll facility are discussed here.

### **4.1 Sample Application.**

#### **4.1.1 Data.**

The data used for this sample application do not correspond to any real situation, but intend to be as close as possible to what would prevail in a typical toll plaza.

The input data echo produced by BOUTH for this study case is shown in Figure 8. The period of study starts at 7 AM and finishes at 5 PM, a 15 minutes time interval

\*\*\*\*\*  
 \* INPUT DATA ECHO \*  
 \*\*\*\*\*

PERIOD OF STUDY

Initial time 7:00  
 Ending time 17:00  
 Time increment 0:15

TOLL PLAZA CAPACITY

# of manned booths	4	Capacity (veh/hr)	650
# of exact change booths	4	Capacity (veh/hr)	550

VEHICULAR COMPOSITION

Percentage of Heavy Vehicles : 15.0  
 Percentage of High Occupancy Vehicles: 0.0

VALUE OF TIME AND SPEED

	LIGHT VEHICLES	HEAVY VEHICLES
VALUE OF TIME		
Dollars/hour	8.30	16.00
SPEED STOPPED FROM		
AND RESUMED TO (mph):	65.00	55.00
ACCELERATION RATE (mph/sec):	5.00	2.00
DECELERATION RATE (mph/sec):	10.00	7.00

ACCIDENTS RATE AND COSTS

Accident Rate (acc/millions of vehs) .01589  
 Fatal Accident Cost: 654350.00  
 Non-Fatal Accident Cost: 31100.00  
 Property Damage Accident Cost: 1350.00

OPERATION VALUES

Toll Plaza Operation cost (dollars/hour): 500.00  
 Toll per light vehicle (dollars/veh): 0.75  
 Toll per heavy vehicle (dollars/veh): 2.00

VEHICLE COSTS

	LIGHT VEHICLES	HEAVY VEHICLES
LUBRICATING OIL		
Dollars/quart:	2.50	1.00
TIRE WEAR		
Dollars/set of tire:	105.00	636.00
MAINTENANCE AND REPAIR		
Cost per 1000 miles:	70.70	212.00
DEPRECIATION		
Value of a new vehicle:	12190.00	61736.00
FUEL VALUE		
Dollars/gallon:	1.01	0.89

Figure 8. : Input Data Echo.

is used. Concerning the capacity of the booths, the values used are slightly smaller than the values suggested by Reference 3 for manned and exact change booths, thus taking into account the adverse effect that trucks and congestion levels induce in capacity. The speed values reflect the situation prevailing in some toll roads in the U.S., where speed limits for cars and trucks are respectively 65 and 55 miles per hour. The administration cost values were impossible to obtain in the literature searched, thus the value of 500 dollar per hour used in this example constitutes a very rough estimate which is included here to show the importance that this variable has in the economic analysis of toll plazas. The toll values of 0.75 for light vehicles and 2 dollars for heavy vehicles are in the range of what is commonly charged in a toll facility. The acceleration and deceleration rates, accident costs, values of time and vehicular costs used in this example are the ones proposed in the software by default. Since a discussion on these values was presented in the last chapter (Section 3.3), they are not mentioned here.

#### **4.1.2 Results.**

Figure 9 shows the first of the three outputs produced by BOOTH. For every 15-minutes interval, the demand in vehicles per hour is provided, as well as the values for queue, vehicles arrived and delay. The capacity of the toll plaza is 4800 vehicles per hour. Based on the demand volumes and the capacity computed by the program, queues are formed during three periods; the longest one occurring between 7:30 and 10:30, when the queue reaches up to 1730 vehicles. At end of the period analyzed, the total user cost is slightly lower than the gross revenue. The average delay is 0.08 hrs.

\*\*\*\*\*  
 \* MEASURES OF EFFECTIVENESS \*  
 \*\*\*\*\*

The capacity of the Toll Plaza is 4800 vph

t i (hrs)	t f (hrs)	q in (vph)	Vehicles Arrived (veh)	Queue (veh)	Delay (veh*hr)
7.00	7.25	4000	1000	0	0.00
7.25	7.50	4400	2100	0	0.00
7.50	7.75	5200	3400	100	12.50
7.75	8.00	5000	4650	150	43.75
8.00	8.25	4900	5875	175	84.38
8.25	8.50	5040	7135	235	135.63
8.50	8.75	6000	8635	535	231.88
8.75	9.00	5900	10110	810	400.00
9.00	9.25	5920	11590	1090	637.50
9.25	9.50	5980	13085	1385	946.88
9.50	9.75	5920	14565	1665	1328.13
9.75	10.00	5060	15830	1730	1752.50
10.00	10.25	3780	16775	1475	2153.13
10.25	10.50	2616	17429	929	2453.63
10.50	10.75	1740	17864	164	2590.25
10.75	11.00	1468	18231	0	2610.75
11.00	11.25	1400	18581	0	2610.75
11.25	11.50	1648	18993	0	2610.75
11.50	11.75	3284	19814	0	2610.75
11.75	12.00	4912	21042	28	2614.25
12.00	12.25	5644	22453	239	2647.63
12.25	12.50	5504	23829	415	2729.38
12.50	12.75	5476	25198	584	2854.25
12.75	13.00	5660	26613	799	3027.13
13.00	13.25	4304	27689	675	3211.38
13.25	13.50	3900	28664	450	3352.00
13.50	13.75	3428	29521	107	3421.63
13.75	14.00	3860	30486	0	3435.00
14.00	14.25	3940	31471	0	3435.00
14.25	14.50	3812	32424	0	3435.00
14.50	14.75	3756	33363	0	3435.00
14.75	15.00	2568	34005	0	3435.00
15.00	15.25	3184	34801	0	3435.00
15.25	15.50	3744	35737	0	3435.00
15.50	15.75	4508	36864	0	3435.00
15.75	16.00	4416	37968	0	3435.00
16.00	16.25	5300	39293	125	3450.63
16.25	16.50	4996	40542	174	3488.00
16.50	16.75	5712	41970	402	3560.00
16.75	17.00	4776	43164	396	3659.75

The delay per vehicle is 0.08 hrs

Figure 9. : Measures of Effectiveness output.

The second output (Cost-Revenue Analysis) shown in Figure 10 lists the accumulated user, administration and total costs at the end of the time intervals. The revenue collected is also listed. Finally, the last two columns of this output describe the variation of the break-even toll for light and heavy vehicles. A further discussion on the results presented in this output is given in the following section, since the graphs elaborated with the values presented in this output allow a more objective interpretation of the variables involved.

The third output (Figure 10), entitled Cost Summary, shows for the two classes of vehicles considered the cost incurred by the users for every item included in this study. In this example, the major user cost generator is the value of the time ( 35,145 dollars), followed by fuel ( 3,657 dollars), while the remaining costs had a low participation in the amount of the total user cost. The middle section of the Cost Summary output sums up the benefit-cost analysis. The summation of the user and administration costs ( 45,096 dollars) is higher than the benefit represented by the gross revenue( 40,095 dollars). The output also shows the tolls that can be charged to obtain a benefit-cost ratio equal to one; 0.92 dollars per car and 1.74 dollars per truck. While the required toll per car is higher than the one proposed (0.75 dollars), the break-even toll for trucks is less than the one charged (2 dollars). Finally, the estimation for the three major pollutants produced by the vehicles during the period of study is shown at the bottom of the output. Carbon monoxide emissions are the highest among the ones evaluated in this study.

\*\*\*\*\*  
 \* COST-REVENUE ANALYSIS \*  
 \*\*\*\*\*

t i (hrs)	t f (hrs)	User Cost	Administ. Cost	Total Cost	Gross Revenue	Break-even toll Light Heavy Vehicles	
7.00	7.25	64	125	189	938	0.17	0.30
7.25	7.50	135	250	385	1969	0.16	0.30
7.50	7.75	345	375	720	3094	0.19	0.35
7.75	8.00	744	500	1244	4219	0.24	0.45
8.00	8.25	1237	625	1862	5344	0.28	0.53
8.25	8.50	1841	750	2591	6469	0.32	0.61
8.50	8.75	2919	875	3794	7594	0.39	0.74
8.75	9.00	4728	1000	5728	8719	0.50	0.95
9.00	9.25	7245	1125	8370	9844	0.64	1.21
9.25	9.50	10497	1250	11747	10969	0.79	1.51
9.50	9.75	14480	1375	15855	12094	0.96	1.83
9.75	10.00	18890	1500	20390	13219	1.13	2.16
10.00	10.25	23037	1625	24662	14344	1.29	2.47
10.25	10.50	26144	1750	27894	15469	1.41	2.68
10.50	10.75	27565	1875	29440	16594	1.45	2.76
10.75	11.00	27798	2000	29798	17092	1.44	2.73
11.00	11.25	27820	2125	29945	17420	1.42	2.69
11.25	11.50	27847	2250	30097	17806	1.40	2.64
11.50	11.75	27899	2375	30274	18576	1.35	2.55
11.75	12.00	28014	2500	30514	19701	1.28	2.42
12.00	12.25	28445	2625	31070	20826	1.22	2.31
12.25	12.50	29367	2750	32117	21951	1.19	2.25
12.50	12.75	30728	2875	33603	23076	1.18	2.23
12.75	13.00	32582	3000	35582	24201	1.18	2.23
13.00	13.25	34531	3125	37656	25326	1.20	2.27
13.25	13.50	36027	3250	39277	26451	1.21	2.29
13.50	13.75	36792	3375	40167	27576	1.20	2.27
13.75	14.00	36991	3500	40491	28581	1.17	2.22
14.00	14.25	37054	3625	40679	29504	1.14	2.16
14.25	14.50	37115	3750	40865	30398	1.11	2.10
14.50	14.75	37175	3875	41050	31278	1.09	2.05
14.75	15.00	37216	4000	41216	31880	1.07	2.02
15.00	15.25	37267	4125	41392	32626	1.05	1.98
15.25	15.50	37327	4250	41577	33503	1.03	1.94
15.50	15.75	37400	4375	41775	34560	1.00	1.89
15.75	16.00	37470	4500	41970	35595	0.98	1.84
16.00	16.25	37715	4625	42340	36720	0.95	1.79
16.25	16.50	38176	4750	42926	37845	0.93	1.76
16.50	16.75	39002	4875	43877	38970	0.92	1.74
16.75	17.00	40096	5000	45096	40095	0.92	1.74

Figure 10. : User Cost-Revenue Analysis output.

\*\*\*\*\*  
 \* COST ANALYSIS SUMMARY \*  
 \*\*\*\*\*

	LIGHT VEHs.	HEAVY VEHs.	TOTAL
Fuel Cost:	2601.92	1055.58	3657.51
Accident Cost:	11.06	2.11	13.17
Oil Cost:	44.53	3.70	48.22
Maintenance:	256.74	81.87	338.61
Depreciation:	357.32	314.75	672.07
Value of Time:	26186.09	8959.25	35145.34
Tire Wear:	115.19	105.83	221.02
<b>Total User Cost:</b>	<b>29572.85</b>	<b>10523.09</b>	<b>40095.95</b>

Gross Revenue:	40095.00
Administration Cost:	5000.00
User Cost:	40095.95
Net Revenue:	-5000.94
Break-even toll per car :	0.92
Break-even toll per truck:	1.74

Cost Values are given in U.S. dollars

EMISSIONS (pounds)	
Carbon Monoxide (CO) :	25981.5
Hydrocarbons (HC) :	2451.2
Nitrogen Oxides (NOX) :	7640.0

Figure 11. : Cost Summary output.

### **4.1.3 Graphical Outputs.**

Figure 12 illustrates the variation of the vehicular flow arriving at the toll plaza. In this figure, the three periods during which the capacity is less than the vehicular arrival rate can be easily identified. The pattern of the variation of the queue generated is seen in Figure 13. The three different queues created correspond to the congestion periods observed in Figure 12. Figure 13 offers an objective way to compare the gross revenue and the user cost. According to this figure, for the period between 7:00 and 9:30, the total revenue collected is higher than the total cost incurred by the users; however, for almost the rest of the period of analysis, the user cost surpasses the gross revenue, particularly following the congested periods.

Finally Figures 15 and 16 show the variation of the break-even toll for light and heavy vehicles respectively. An interesting fact about these graphs is that both curves have the same shape. As in general, unit user costs are higher for trucks than for cars, it was expected the break-even toll values to be higher for heavy vehicles than for light vehicles. Comparing these two graphs to the queue length graph, it is observed that the maximum points in these curves occur almost at the same time, thus indicating that a close relationship exists between break-even toll and vehicular arrival rate. The difference on the occurrence of the maximum points between the queue curve and the break-even toll curves is due to the decreasing in the demand flow rate. As the vehicular arrival rate decreases, the administration cost is divided by less vehicles per hour than when congestion levels prevail, therefore, the value of the total cost per vehicle remains high until the unit user cost decreases significantly.

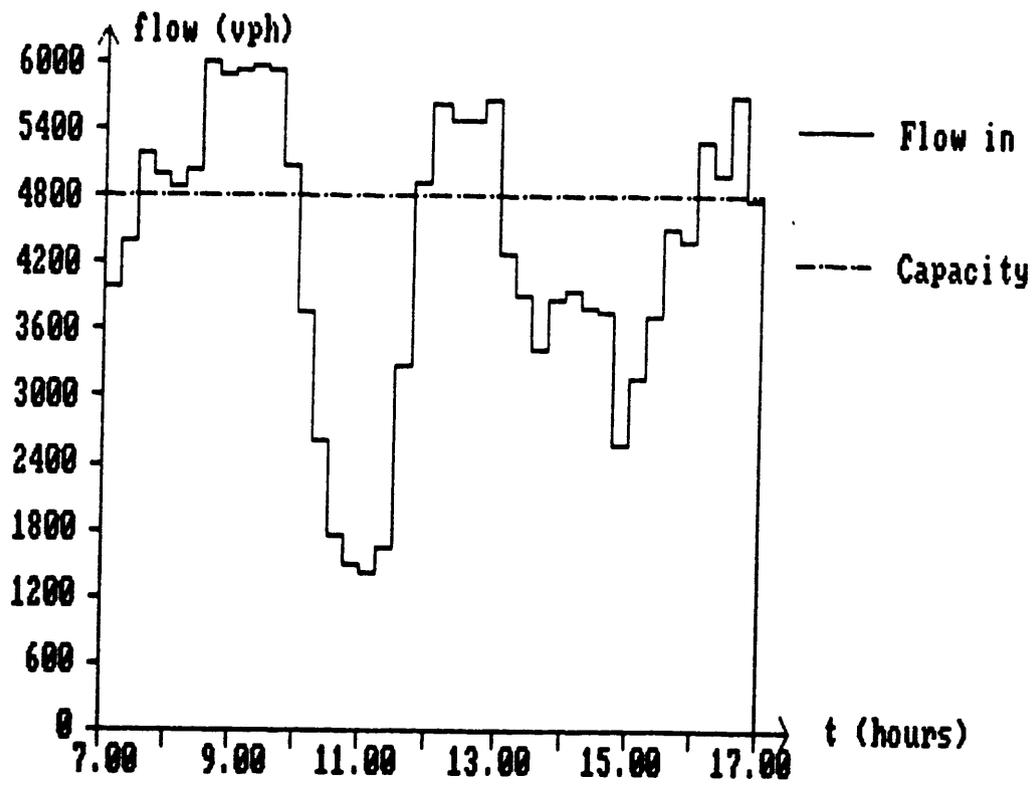


Figure 12. : Vehicular Demand graph.

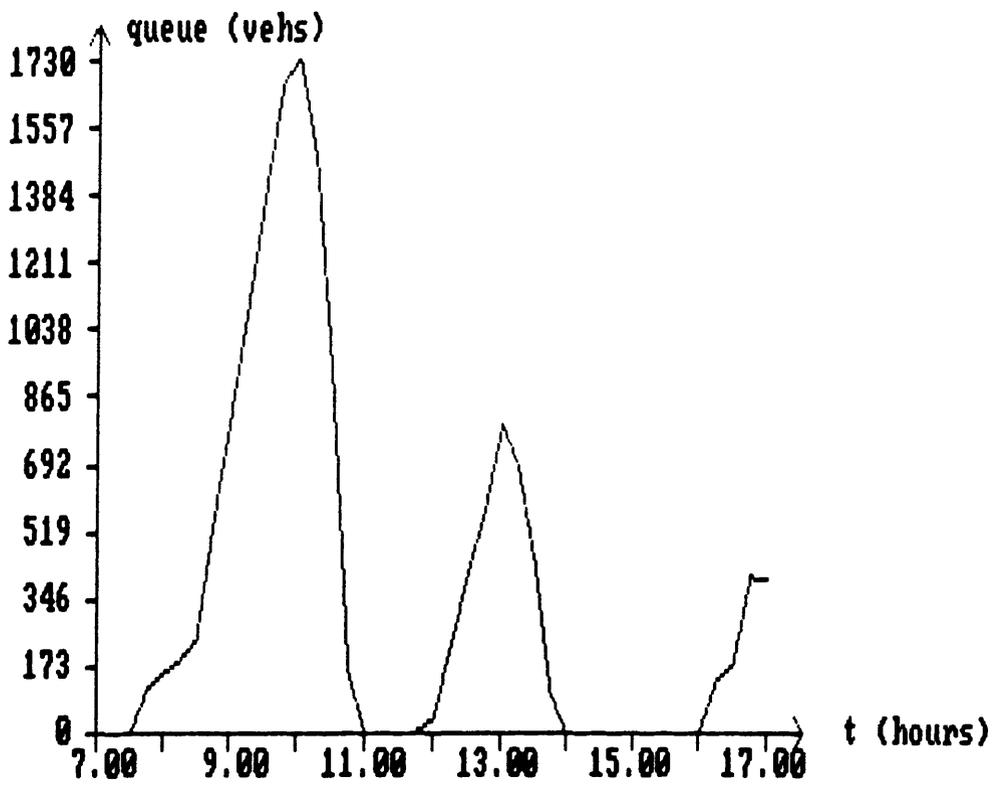


Figure 13. : Queue graph.

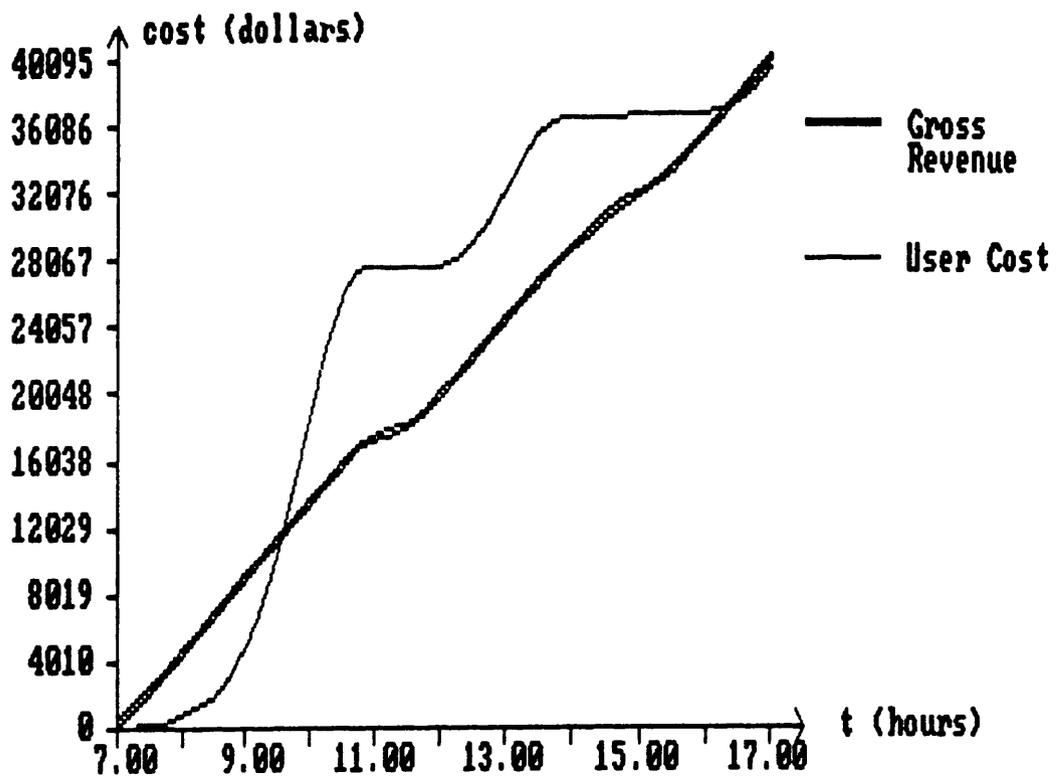


Figure 14. : User Cost-Revenue graph.

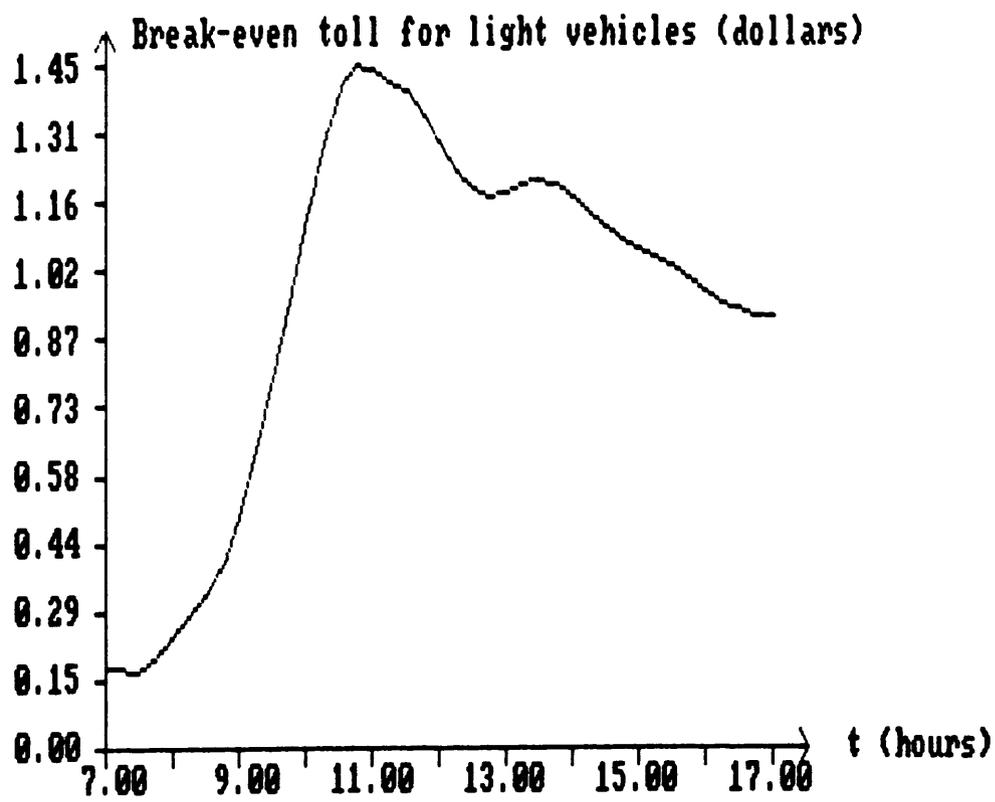


Figure 15. : Break-even toll variation for light vehicles

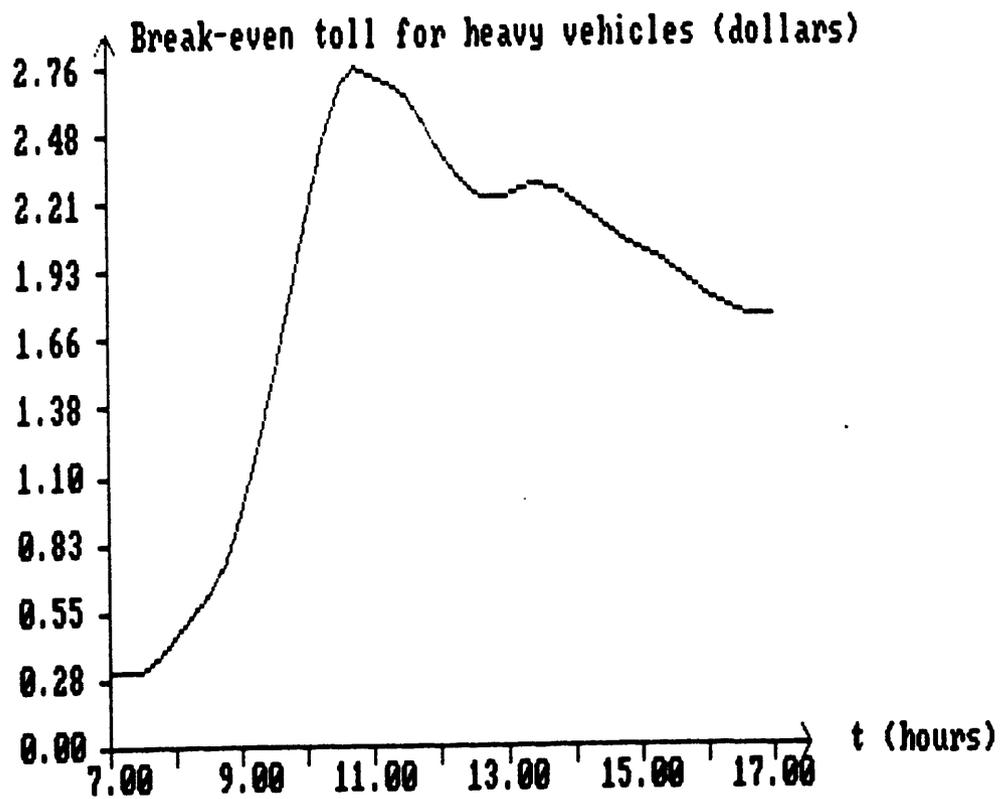


Figure 16. : Break-even toll variation for heavy vehicles

#### **4.1.4 Conclusions on the case analyzed.**

The sample case discussed so far, demonstrates how, even for small values of average delay (less than five minutes), the user cost can surpass the revenues generated.

Particularly significant is the fact that around 85 percent of the user cost is due to the value of the time lost by the drivers when waiting to pay. If this cost had not been considered, the revenue would have easily exceeded the total user cost.

### **4.2 Cost Effectiveness of the Toll Charged.**

In this section, a number of criteria for the amount of toll that should be charged at the toll facilities are discussed. Also, the conditions under which it is recommended to exempt patrons from paying toll are established. Many special situations are analyzed, and a discussion on the efficacy of the models presented is included.

#### **4.2.1 Toll charged to overcome user cost.**

If the economic models presented in Chapter 2 are summarized, the user cost can be expressed as:

$$UC = \alpha veh + \beta D_s \quad (4.1)$$

Where:

- $UC$  = User Cost in US dollars.
- $D_s$  = Stopped delay in veh·hrs.
- $veh$  = Vehicles arrived at the toll plaza.
- $\alpha$  = Constant relating dollars to number of stops.
- $\beta$  = Constant relating dollars to hours of stopped delay.

Based on discussions in chapter 2,  $\alpha$  is a function of the approach speed, as well as consumption rates and unit costs of the items considered; i.e. Fuel, Oil, Depreciation, Tire wear, Maintenance and Repair, Value of time and Accidents. On the other hand, the value of  $\beta$  depends on the idling rates and unit costs of the above-mentioned items.  $\alpha$  and  $\beta$  are computed separately for light and heavy vehicles.

The unit user cost is defined as the amount of money, besides the toll paid, incurred by an average user when arriving at a toll plaza. It is represented by:

$$\overline{UC} = \alpha + \beta \overline{D}_s \quad (4.2)$$

Where:

- $\overline{UC}$  = Unit user cost in US dollars.
- $\overline{D}_s$  = Average stopping delay in hours.

If the toll charged is less than the value obtained from this last equation, patrons should not be charged, since they are already incurring a cost higher than what they are paying for using the facility.

Alternatively, if the payment exemption is not made, the toll to be charged in order to overcome the user cost (  $toll_{uc}$  ) must be set at:

$$toll_{uc} > \alpha + \beta \overline{D}_s \quad (4.3)$$

In this case, a complete delay analysis should be performed, since, as discussed in chapter 2, the average stopped delay depends on the vehicular arrival rate pattern and the capacity of the toll plaza. However, under specific situations such as discussed in the following section, the criterion set by the last equation can be simplified in terms of a few key variables.

#### 4.2.2 Special cases.

The models introduced in this section are developed under the following constraints:

1. Vehicles arrive at the toll plaza at a constant rate during the period of analysis.
2. The capacity of the toll plaza remains constant during the analysis.
3. No queues are formed at the beginning of the analysis.

Graphs for passenger cars are presented for better illustrate the models discussed. The values of approach speed, acceleration and deceleration rates, unit costs and accident rates shown in Figure 8 were used in preparing the graphs presented in this section. Since the value of time is the major source of user cost, it is taken into account as a variable in the graphs presented.

When the vehicle arrival rate is constant, stopping delay can be expressed as:

$$D_s = \frac{queue \cdot t}{2} \quad (4.4)$$

Where:

*queue* = queue length at time *t*

*t* = any time during the period of analysis.

Number of vehicles is then represented as:

$$veh = q_{in} t \quad (4.5)$$

Where  $q_{in}$  is the constant vehicle arrival rate in vph.

Therefore, the equation introduced in chapter 2 to evaluate average stopped delay (Eq. 2.9) can be modified based on the characteristics of the special case discussed in this section, as follows:

$$\bar{D}_s = \frac{queue}{2 q_{in}} \quad (4.6)$$

Moreover, if the queue length is expressed as:

$$queue = (q_{in} - cap) t \quad (4.7)$$

Where  $cap$  is the capacity of the toll plaza. Then, the average stopped delay is:

$$\bar{D}_s = \frac{t}{2} \left( 1 - \frac{cap}{q_{in}} \right) \quad (4.8)$$

Combining equations 4.3 and 4.6, the toll needed to overcome the user cost is:

$$toll_{UC} = \alpha + \frac{\beta}{2} \left( \frac{queue}{q_{in}} \right) \quad (4.9)$$

The parameters  $\alpha$  and  $\beta$  are treated as functions of the value of time, therefore, equation 4.9 represents a family of curves, as shown in Figure 17. From this figure it can be seen that higher toll values correspond to higher queue length and higher values of time.

Combining equations 4.3 and 4.8, the toll needed to overcome the user cost can also be expressed as:

$$toll_{UC} = \alpha + \beta \left( \frac{t}{2} \right) \left( 1 - \frac{cap}{q_{in}} \right) \quad (4.10)$$

If  $t$  is a fixed value, in this case one hour, equation 4.10 represents in a  $(toll_{UC}, q_{in}/cap)$  system, a family of curves for several values of time, as illustrated by Figure 18. Some important conclusions can be drawn from these graphs. The toll is almost the same for different values of time if the demand-capacity ratio is less than one, however, for ratios greater than one, the amount of toll becomes very sensitive to the value of time. Also, it can be seen that the toll value rapidly increases after the flow-capacity ratio reaches one.

If on the other hand,  $\alpha$  and  $\beta$  are fixed values,  $toll_{UC}$  is represented as function of  $t$  for several  $q_{in}/cap$  ratios, as shown in Figures 19 to 21.

It is observed in these graphs, that the toll increases linearly in time. As expected, higher demand-capacity ratios and values of time imply higher tolls.

### 4.2.3 Toll to overcome administration and user costs.

The administration cost incurred in a toll plaza, at  $t$  hours is:

$$AC = UAC t \quad (4.11)$$

Where:

$AC$  = Administration Cost in US dollars.

$UAC$  = Unit Administration Cost in US dollars per hour.

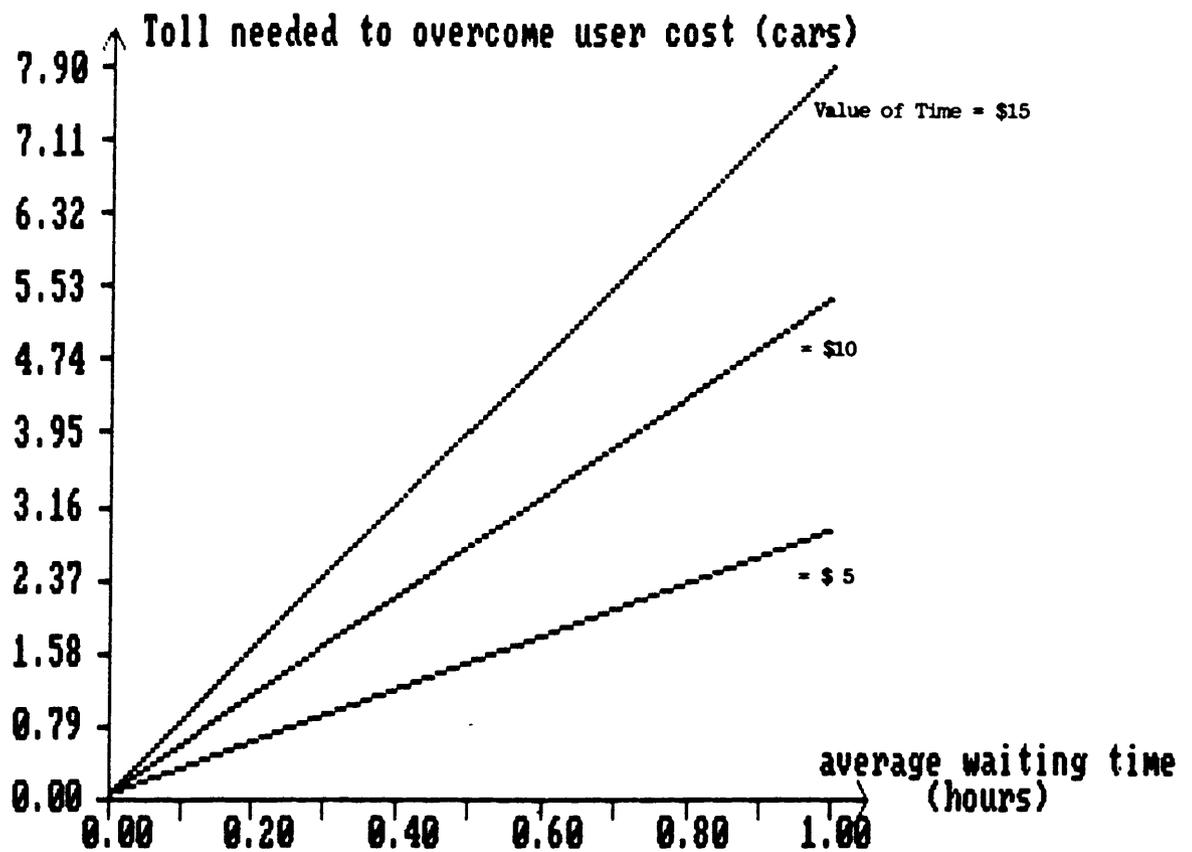


Figure 17. : Toll to be charged as a function of waiting time in queue.

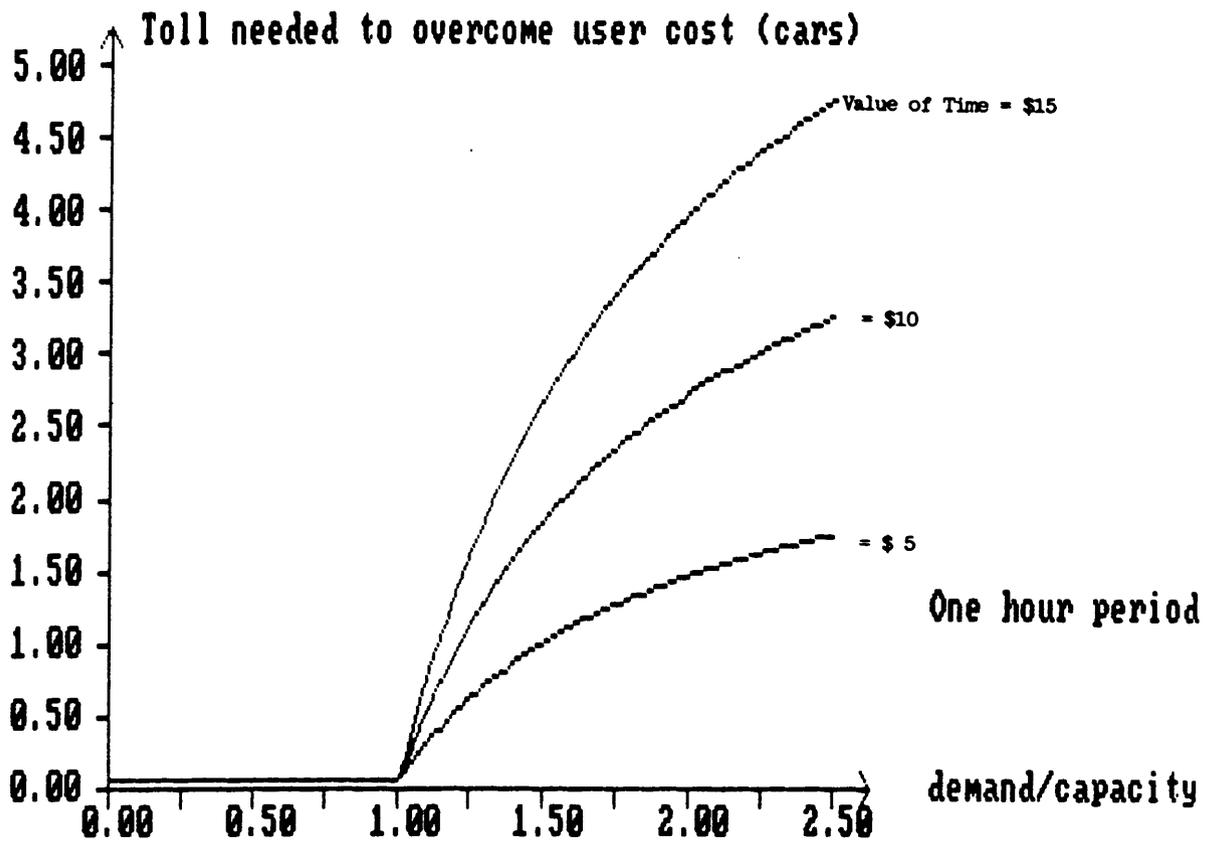


Figure 18. : Toll to be charged as a function of the demand-capacity ratio for some values of time.

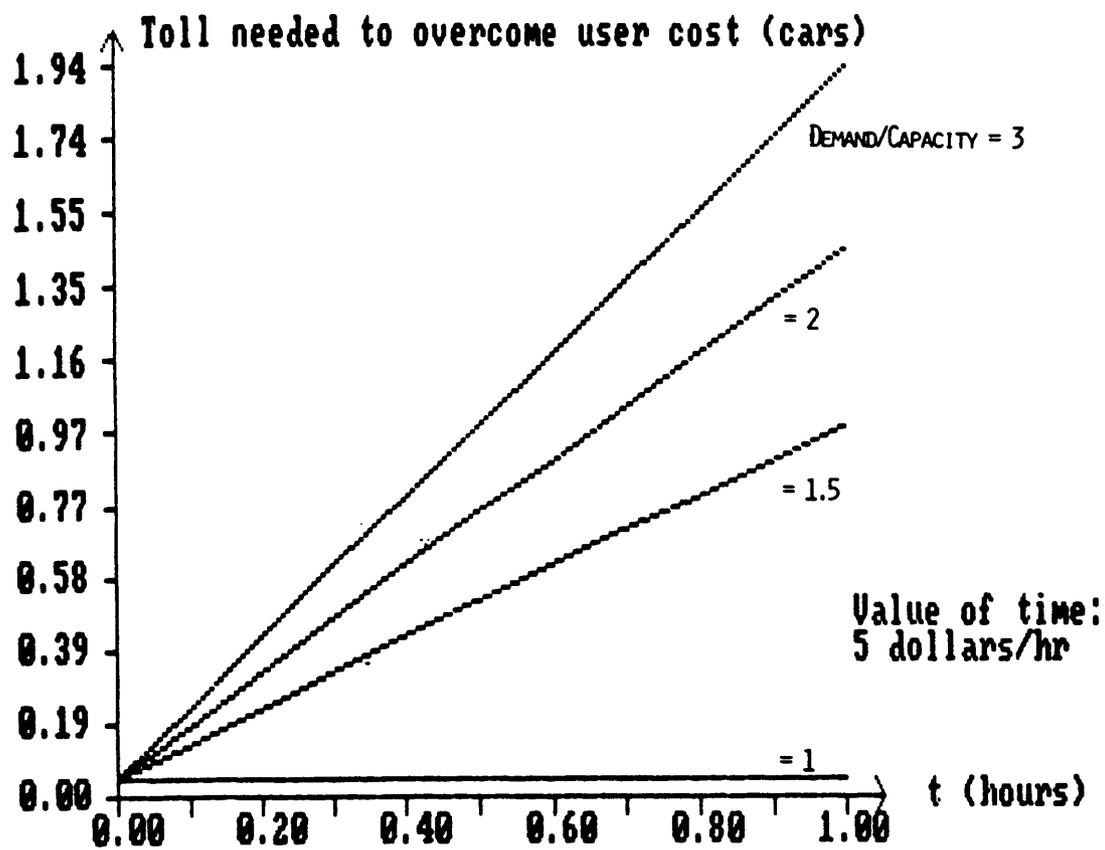


Figure 19. : Toll to be charged as a function of  $t$  for some  $q_m/cap$  values. Value of the time is 5 dollars.

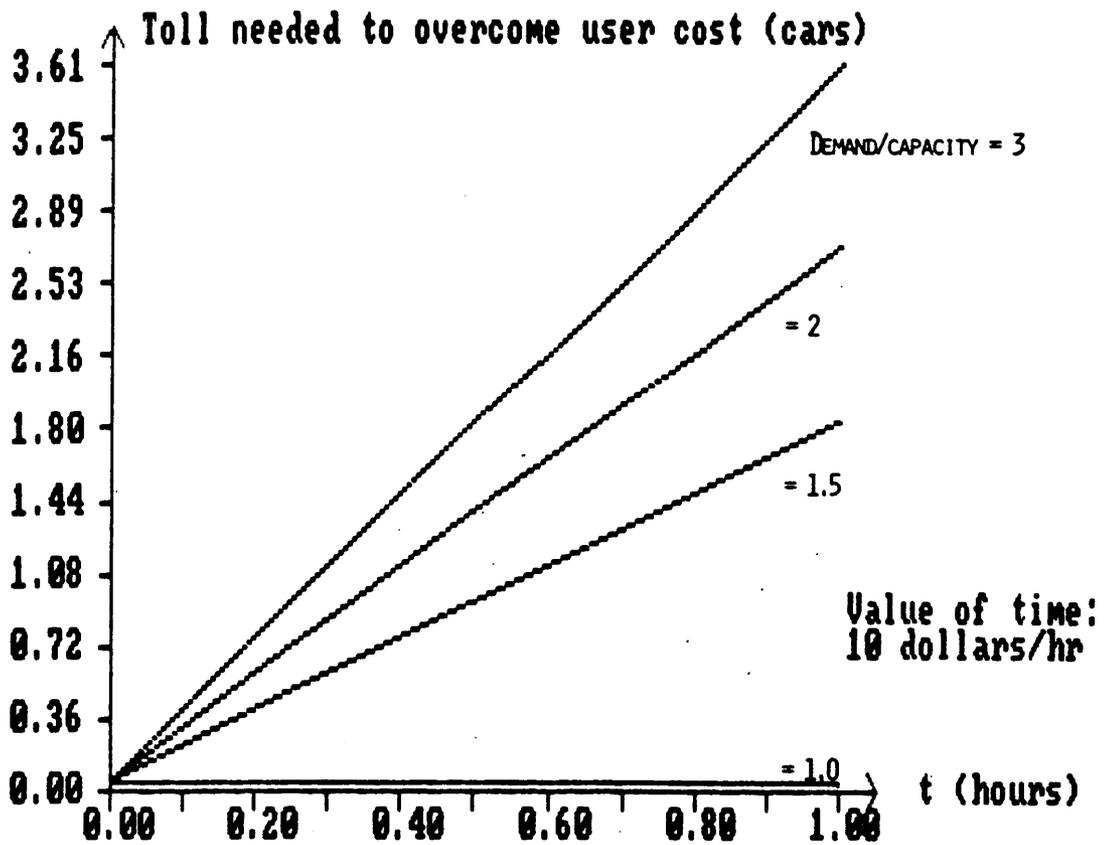


Figure 20. : Toll to be charged as a function of  $t$  for some  $q_n/cap$  values. Value of the time is 10 dollars.

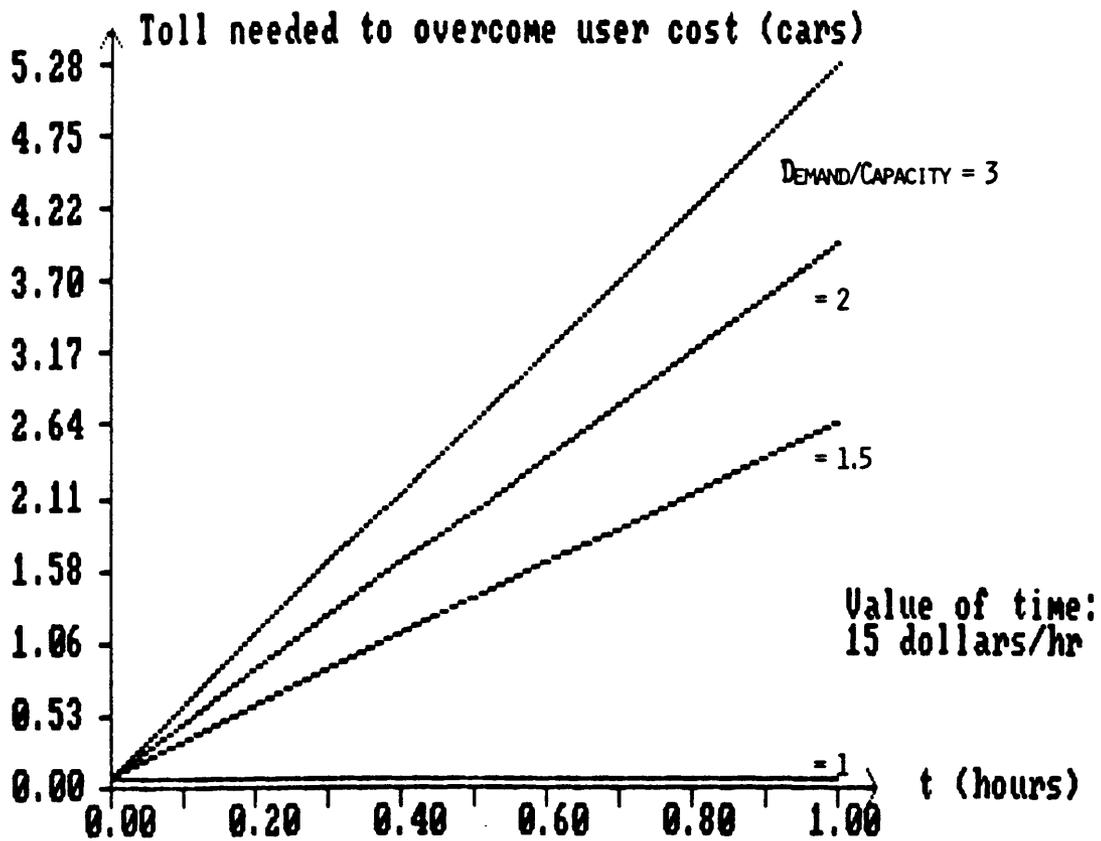


Figure 21. : Toll to be charged as a function of  $t$  for some  $q_m/cap$  values. Value of the time is 15 dollars.

Then, the total cost per vehicle is:

$$\bar{C} = \alpha + \beta \bar{D}_s + \frac{UAC \cdot t}{veh} \quad (4.12)$$

$\bar{C}$  can be thus equated to the toll charged to overcome total cost ( $toll_c$ ) as follows:

$$toll_c = \alpha + \beta \bar{D}_s + \frac{UAC t}{veh} \quad (4.13)$$

This equation represents what was defined in Chapter 2 as the break-even toll. In order to evaluate  $toll_c$  for a general situation, the delay analysis explained in chapter 2 must be performed. Again, if as discussed before,  $q_{in}$  is assumed constant, then Equation 4.13 can be rewritten as:

$$toll_c = \alpha + \beta \left( \frac{t}{2} \right) \left( 1 - \frac{cap}{q_{in}} \right) + \frac{UAC}{q_{in}} \quad (4.14)$$

Because of the last term in this expression, Equation 4.14 cannot be easily simplified unlike the previous case, where only the user cost was taken into account. Therefore, the only possible simplification occurs when no delay is incurred, that is:

$$toll_c = \alpha + \frac{UAC}{q_{in}} \quad (4.15)$$

#### **4.2.4 Discussion and Recommendations.**

The models presented in this section apply to very specific situations. Based on the enormous data required to perform the economic analysis in toll plazas, the computation of the amount of toll needed to overcome the costs incurred is rather involved and cannot be readily synthesized by a family of curves or by an easy-to-use equation. The family of curves provided, however, highlights the relative influence of variables such as the user value of time or the vehicular demand on the magnitude of the toll charged as a decision variable. Therefore, the utilization of the software developed for this research is highly recommended, considering that the use of microcomputers has been widely increased in recent years. If vehicular volumes upstream of the toll plaza are introduced interactively to the program, by using an automatic counter, the toll needed to be charged could be determined almost instantaneously. However, the most likely utility of this software is not as a real-time determinant of toll charges. Rather, it can be used as an effective planning tool to provide input in the decision on how much toll is to be charged at different periods of the day.

### ***4.3 Poisson Vehicular Demand.***

Poisson vehicular arrivals were randomly generated and then inputted to BOUTH. The objective of this was to compare the measures of effectiveness provided by the program and the ones that are obtained from classical Queueing Theory. These measures of effectiveness are average waiting time and average queue length of the

system. As service rate  $\mu$  is considered constant in the program and for this special case, the arrivals are Poisson distributed with rate  $\lambda$ , the case analyzed in this section is an M/D/1 model. For this example:

$$\lambda = 1800 \text{ vph}$$

$$\mu = 1950$$

$$\rho = \lambda/\mu = 0.923$$

For a steady-state system, the length of the queue  $L_q$  is obtained from the Pollaczek-Khintchine formula (Reference 13) :

$$L_q = \frac{\rho^2 + \lambda^2 \sigma_s^2}{2(1 - \rho)} \quad (4.16)$$

Where  $\sigma_s^2$  is the variance of the service time distribution. Letting  $\sigma_s^2 = 0$  gives results for the M/D/1 model. Therefore, for the values of this example,  $L_q = 5.54$  vehicles, then the expected average waiting time is obtained using Little's formula (13) :

$$W_q = \frac{L_q}{\lambda} \quad (4.17)$$

for this case  $W_q = 0.00308$  hours. Figure 22 shows the Measures of Effectiveness output obtained from BOOTH. In this case, the period of the simulation is one hour, for which the arrival rates are Poisson distributed and are randomly generated for every one-minute time interval. The average values of delay and queue can be read from the bottom of the output. As observed, the values computed by the software ( $L_q = 1.57$  vehs and  $W_q = 0.00087$  hours) are quite different from the ones computed using Formulas 4.16 and 4.17. This difference is due to the fact that a much longer period of simulation is required in order to reach a steady-state condition as assumed in derivation of Equation 4.16. Unfortunately, due to computer memory

limitations such a simulation cannot be performed using BOUTH. Currently, memory requirements limit the situation to only 100 time intervals. However, very reasonable results can be obtained from BOUTH for  $\rho$  values greater than one when simulating short periods, whereas Queueing Theory equations derived for steady state do not work for this condition. Furthermore, the current structure of BOUTH does not allow a variability in the service rate. Thus only deterministic service rates can be simulated.

\*\*\*\*\*  
 \* MEASURES OF EFFECTIVENESS \*  
 \*\*\*\*\*

The capacity of the Toll Plaza is 1950 vph

t i (hrs)	t f (hrs)	q in (vph)	Vehicles Arrived (veh)	Queue (veh)	Delay (veh*hr)
12.00	12.02	1620	27	0	0.00
12.02	12.03	1920	59	0	0.00
12.03	12.05	2040	93	1	0.01
12.05	12.07	1800	123	0	0.02
12.07	12.08	2040	157	1	0.04
12.08	12.10	2040	191	3	0.07
12.10	12.12	1320	213	0	0.10
12.12	12.13	1920	245	0	0.10
12.13	12.15	2100	280	2	0.12
12.15	12.17	1200	300	0	0.14
12.17	12.18	1920	332	0	0.14
12.18	12.20	2340	371	6	0.20
12.20	12.22	1500	396	0	0.25
12.22	12.23	1800	426	0	0.25
12.23	12.25	1560	452	0	0.25
12.25	12.27	1380	475	0	0.25
12.27	12.28	1980	508	0	0.25
12.28	12.30	1380	531	0	0.26
12.30	12.32	1860	562	0	0.26
12.32	12.33	1200	582	0	0.26
12.33	12.35	1380	605	0	0.26
12.35	12.37	1800	635	0	0.26
12.37	12.38	2220	672	4	0.30
12.38	12.40	1740	701	1	0.34
12.40	12.42	1980	734	1	0.36
12.42	12.43	2160	770	5	0.42
12.43	12.45	1860	801	3	0.49
12.45	12.47	1260	822	0	0.52
12.47	12.48	1680	850	0	0.52
12.48	12.50	1980	883	0	0.52
12.50	12.52	1860	914	0	0.52
12.52	12.53	1500	939	0	0.52
12.53	12.55	1620	966	0	0.52
12.55	12.57	1620	993	0	0.52
12.57	12.58	1740	1022	0	0.52
12.58	12.60	1440	1046	0	0.52
12.60	12.62	1980	1079	0	0.53
12.62	12.63	1620	1106	0	0.53
12.63	12.65	1440	1130	0	0.53
12.65	12.67	2160	1166	3	0.56
12.67	12.68	1740	1195	0	0.59
12.68	12.70	1620	1222	0	0.59
12.70	12.72	2220	1259	4	0.63
12.72	12.73	1800	1289	2	0.68
12.73	12.75	1500	1314	0	0.70
12.75	12.77	1860	1345	0	0.70
12.77	12.78	1560	1371	0	0.70
12.78	12.80	1680	1399	0	0.70
12.80	12.82	1140	1418	0	0.70
12.82	12.83	1920	1450	0	0.70
12.83	12.85	2460	1491	8	0.77
12.85	12.87	1320	1513	0	0.84
12.87	12.88	1980	1546	0	0.85
12.88	12.90	2340	1585	7	0.91
12.90	12.92	1260	1606	0	0.97
12.92	12.93	2400	1646	7	1.03
12.93	12.95	2280	1684	13	1.20
12.95	12.97	1440	1708	4	1.35
12.97	12.98	2100	1743	7	1.44
12.98	13.00	1800	1773	4	1.54

The delay per vehicle is 0.00087 hrs  
 The average queue is : 1.574997

Figure 22. : Measures of Effectiveness output for Poisson distributed arrivals.

## **5.0 Summary and Conclusions.**

### **5.1 *The Economic Analysis***

The work described in this report is an attempt to measure the cost incurred by drivers when using a toll facility. A major concern was to provide estimation models for all the possible cost generators, even if, as observed for some items, the contribution to the total user cost was relatively small.

It could be quite beneficial for the economic analysis described in chapter 2, if variables such as degree of congestion and vehicular composition are taken into account in the evaluation of the toll plaza capacity. The inclusion of these variables would constitute a research topic in itself, since there is not an extensive body of literature on toll facility operations. Also, if a criterion to assess a unit cost to toxic emissions is eventually set, the effectiveness of the procedure described can be significantly enhanced.

A major contribution of this work has been the collection and synthesis of a large amount of scattered information on highway user costs and to incorporate them in the economic analysis of toll plaza operations.

As seen, the procedure described in this work can be easily applied to determine the cost effectiveness of the charging process in a toll plaza. Furthermore, its utilization could be expanded to a whole set of toll collection points, allowing the evaluation of a complete collection system.

Finally, a main contribution of this work is to offer a methodology for measuring the inconvenience caused to the users by the toll collection systems presently in use. This methodology can be very useful when trying to prove the economic feasibility of an Automatic Vehicle Identification method and to set demand-related variables toll charges for AVI systems. A study carried out by the Hong Kong Government(11) has shown that there are no major technological barriers for the implementation of an AVI system, however its introduction will depend on the economical advantages it could offer as compared to the traditional charging systems. One of these advantages is the reduction or elimination of the user costs evaluated in this research.

## **5.2 The Software.**

The software developed is a very valuable tool to be used in operational analysis and policy decisions on Toll Plazas. It is extremely easy to use and provides results in a very detailed way. Although BOOTH was developed specifically to be used for Toll Plaza Operations, it can be easily adjusted to provide measures of effectiveness such as queue and delay, user cost estimates, and emissions in other Traffic

Engineering applications. Among potential applications are situations which include a downstream reduction in vehicular capacity, such as work zones or intersections. Perhaps, one of the main disadvantages of the program is that the results provided are very sensitive to the accuracy of the data entered. In many situations, and particularly when congestion levels prevail, the value of time is the main generator of the user cost. Unfortunately, this is the most difficult value to evaluate and if it is not properly estimated, the analysis can lead to erroneous results.

One addition to the program could be to considerate the capacity of the toll plaza as a value which can vary during the period of study. Values on consumption and emission rates for heavy trucks with more than three axles are needed. An analysis which takes into account more than two classes of vehicles would be a desirable modification of the program. The incorporation of new and more precise techniques for the estimation of emissions and fuel consumed would increase the reliability of the results given by the program. Simeonidis (Reference 12) presents a method in which instantaneous vehicle speeds and accelerations are transformed to engine torque and RPM which in turn are used to predict fuel consumption and emissions for heavy-duty vehicles. Finally, a great achievement would be to interface program to an automatic vehicle counter to read vehicular data directly.

## References

1. Wood, H.C. and Hamilton, C.S. *Design of Toll Plazas from the Operator's Viewpoint*. HRB, Highway Research Board Proc., Vol 34, 1955.
2. Hall, R. and Daganzo, C. *Tandem Toll Booths for the Golden Gate Bridge*. TRB, Transportation Research Record 910, 1983.
3. Larsen, O. *The Toll Ring in Bergen, Norway* TRB, Transportation Research Record 1107, 1987.
4. Pignataro, L. J. *Traffic Engineering, Theory and Practice*. 1973.
5. Zaniewsky, J.P. , B.C. Butler, G. Cunnigham, G. E. Elkins, M.S. Paggi, and R. Machemehl. *Vehicle Operating Costs, Fuel Consumption, and Pavement Type and Conditions Factors*. Federal Highway Administration, March 1982.
6. *A Manual on User Benefit Analysis of Highway and Bus Transit Improvements, 1977*. American Association of State Highway and Transportation Officials.
7. Brown, R. *A method for Determining the Accident Potential of an Intersection*. Traffic Engineering and Control. Dec. 1981. pp 684 - 651
8. U.S. Department of Transportation, National Highway Traffic Safety Administration, *Societal Costs of Motor Vehicle Accidents*, Preliminary Report (April 1975).
9. Meyers, W.S. *Comparison of Truck and Passenger-Car Accident Rates on Limited Access Facilities*. TRB, Transportation Research Board, 1981.
10. Dale, C. W., *Procedure for Estimating Highway User Costs, Fuel Consumption and Air Pollution*, Office of Traffic Operations, Federal Highway Administration, May 1980
11. Catling, I and G. Roth. *Electronic Road Pricing in Hong Kong: An Opportunity for Road Privatization?*. TRR, Transportation Research Record 1107, 1987.
12. Simeonidis, C. *Emmissions and Fuel Consumption as Functions of Vehicle Speed and Acceleration*. Master of Science Thesis, The University of Texas at Austin, December 1981.
13. Gross, D. and C. Harris. *Fundamentals of Queueing Theory*, second edition, 1985.

# **Appendix A. Consumption and Emissions Rates**

As suggested by Reference 5, except when mentioned.

Idling vehicular operation cost components and emissions per 1000 hours.

Type of Vehicle	Oil (quarts)	Maintenance and Repair (Percent of Average cost per 1000 mi)	Depreciation (Percent of New Price)	Fuel (gallons)	Carbon Monoxide (tons)	Hydrocarbons (tons)	Nitrogen Oxides (tons)
Small Car	5.8	57	0.81	271	1.105	0.073	0.023
Medium Car	5.8	57	0.81	563	1.105	0.073	0.023
Large Car	5.8	59	0.81	563	1.105	0.073	0.023
Pickup	3.5	60	0.50	756	1.105	0.073	0.023
2 Axles SU	3.2	23	1.10	1198	1.105*	0.073*	0.023*
3 Axles SU	3.46	26	1.10	398	1.105*	0.073*	0.023*
2 Axles Trailer	3.46	24	0.38	470	1.105*	0.073*	0.023*
3 Axles Trailer	3.46	24	0.38	470	1.105*	0.073*	0.023*

\* Rates assumed equal to the ones for light vehicles.

Note: Emission rates are suggested by Dale (10).

Excess fuel consumption (gal/1000 stops) for a speed stopped from and and returned to.

speed (mph)	small car	medium car	large car	pickup	2 axles S U	3 axles S U	2-S2 semi	3-S2 semi
5	0.53	1.00	1.16	1.13	2.96	4.35	8.10	8.96
10	0.97	1.98	2.35	2.23	5.96	8.94	15.40	17.50
15	1.43	3.02	3.63	3.41	9.58	14.20	23.00	26.90
20	2.01	4.18	5.08	4.77	14.40	20.50	32.50	38.80
25	2.70	5.43	6.68	6.24	20.30	28.20	42.60	51.70
30	3.50	6.81	8.46	7.85	27.80	37.90	53.40	65.90
35	4.63	8.68	10.80	9.99	37.30	50.00	65.70	82.00
40	5.91	10.70	13.30	12.35	48.60	64.80	79.60	100.00
45	7.35	12.90	15.90	14.80	61.90	82.80	95.90	121.00
50	8.94	15.30	18.70	17.60	76.10	103.00	115.00	114.00
55	10.90	17.90	21.70	20.90	90.70	124.00	135.00	168.00
60	13.10	20.80	24.90	25.15	106.00	146.00	156.00	192.00
65	15.60	24.30	28.30	30.20	123.00	168.00	179.00	217.00
70	18.70	29.20	32.00	38.60	140.00	190.00	205.00	243.00

Excess oil consumption (qts/1000 stops) for a speed stopped from and and returned to.

speed (mph)	small car	medium car	large car	pickup	2 axles S U	3 axles S U	2-S2 semi	3-S2 semi
5	0.005	0.005	0.005	0.005	0.017	0.034	0.034	0.070
10	0.009	0.008	0.008	0.008	0.028	0.058	0.058	0.118
15	0.012	0.011	0.011	0.010	0.038	0.078	0.078	0.159
20	0.015	0.013	0.013	0.012	0.047	0.097	0.097	0.198
25	0.017	0.015	0.015	0.014	0.056	0.116	0.116	0.237
30	0.019	0.017	0.017	0.016	0.066	0.135	0.135	0.276
35	0.022	0.020	0.020	0.019	0.077	0.156	0.156	0.318
40	0.025	0.023	0.023	0.022	0.088	0.178	0.178	0.360
45	0.028	0.026	0.026	0.025	0.099	0.200	0.200	0.401
50	0.032	0.029	0.029	0.028	0.111	0.225	0.225	0.442
55	0.036	0.032	0.032	0.031	0.126	0.256	0.256	0.493
60	0.040	0.036	0.035	0.034	0.148	0.299	0.288	0.562
65	0.046	0.042	0.039	0.039	0.182	0.361	0.361	0.662
70	0.054	0.051	0.044	0.048	0.247	0.462	0.462	0.828

Excess tire wear (percent worn/1000 stops) for a speed stopped and returned to.

speed (mph)	small car	medium car	large car	pickup	2 axles S U	3 axles S U	2-S2 semi	3-S2 semi
5	0.02	0.03	0.03	0.03	0.04	0.03	0.02	0.03
10	0.08	0.11	0.12	0.11	0.16	0.11	0.09	0.11
15	0.18	0.24	0.27	0.24	0.36	0.34	0.20	0.25
20	0.32	0.42	0.48	0.42	0.63	0.42	0.35	0.44
25	0.50	0.64	0.74	0.64	0.97	0.65	0.54	0.68
30	0.71	0.90	1.05	0.90	1.38	0.93	0.77	0.97
35	0.90	1.14	1.33	1.13	1.72	1.16	0.96	1.21
40	1.11	1.40	1.65	1.39	2.10	1.42	1.18	1.49
45	1.34	1.68	2.00	1.66	2.52	1.71	1.42	1.80
50	1.59	1.98	2.37	1.95	2.97	2.03	1.68	2.14
55	1.85	2.29	2.77	2.26	3.45	2.37	1.96	2.50
60	2.13	2.62	3.19	2.58	3.95	2.73	2.26	2.88
65	2.43	2.97	3.64	2.93	4.46	3.11	2.57	3.29
70	2.76	3.35	4.11	3.36	4.93	3.49	2.88	3.70

Excess depreciation (percent of price of new vehicle/1000 stops) for a speed stopped from and and returned to.

speed (mph)	small car	medium car	large car	pickup	2 axles S U	3 axles S U	2-S2 semi	3-S2 semi
5	0.002	0.002	0.001	0.001	0.002	0.003	0.001	0.001
10	0.004	0.003	0.002	0.002	0.004	0.005	0.002	0.002
15	0.006	0.004	0.003	0.003	0.005	0.007	0.003	0.003
20	0.008	0.005	0.004	0.004	0.006	0.009	0.004	0.004
25	0.010	0.006	0.005	0.005	0.007	0.011	0.005	0.005
30	0.012	0.007	0.006	0.006	0.008	0.013	0.006	0.006
35	0.014	0.008	0.007	0.007	0.009	0.015	0.007	0.007
40	0.016	0.009	0.008	0.008	0.011	0.017	0.008	0.008
45	0.018	0.010	0.009	0.009	0.013	0.019	0.009	0.009
50	0.020	0.012	0.010	0.010	0.015	0.021	0.010	0.010
55	0.022	0.014	0.011	0.011	0.017	0.024	0.011	0.011
60	0.024	0.016	0.012	0.012	0.020	0.027	0.012	0.012
65	0.027	0.018	0.013	0.013	0.024	0.031	0.013	0.013
70	0.030	0.021	0.014	0.015	0.030	0.037	0.015	0.015

Excess maintenance and repair cost (percent of avg cost/1000 stops) for a speed stopped from and and returned to.

speed (mph)	small car	medium car	large car	pickup	2 axles S U	3 axles S U	2-S2 semi	3-S2 semi
5	0.04	0.04	0.03	0.04	0.03	0.02	0.03	0.03
10	0.15	0.15	0.13	0.15	0.11	0.09	0.11	0.11
15	0.33	0.33	0.30	0.33	0.24	0.22	0.25	0.26
20	0.58	0.58	0.54	0.58	0.43	0.40	0.45	0.48
25	0.91	0.84	0.91	0.69	0.65	0.69	0.72	0.78
30	1.31	1.31	1.21	1.31	1.02	0.96	1.06	1.16
35	1.75	1.76	1.63	1.75	1.41	1.33	1.47	1.63
40	2.25	2.27	2.11	2.25	1.87	1.77	1.95	2.20
45	2.81	2.84	2.65	2.81	2.40	2.29	2.51	2.88
50	3.42	3.46	3.24	3.42	3.00	2.89	3.16	3.67
55	4.08	4.13	3.88	4.08	3.68	3.57	3.89	4.58
60	4.78	4.85	4.56	4.78	4.44	4.33	4.71	5.61
65	5.51	5.61	5.28	5.52	5.28	5.18	5.63	6.78
70	6.26	6.40	6.03	6.28	6.20	6.12	6.64	8.09

Excess carbon monoxide emissions (tons/million of stops) for a speed stopped from and returned to.

speed (mph)	small car	medium car	large car	pickup	2 axles S U	3 axles S U	2-S2 semi	3-S2 semi
5	0.34	0.31	0.33	0.31	0.64	0.18	0.18	0.08
10	0.70	0.66	0.67	0.66	2.00	0.56	0.56	0.25
15	1.20	1.20	1.20	1.20	4.40	1.20	1.20	0.55
20	2.10	2.10	2.10	2.10	7.80	2.20	2.20	0.98
25	3.30	3.40	3.40	3.40	12.00	3.50	3.50	1.60
30	5.00	5.20	5.20	5.20	18.00	5.20	5.20	2.30
35	7.30	7.60	7.70	7.50	25.00	7.30	7.20	3.20
40	10.00	11.00	11.00	11.00	34.00	9.80	9.80	4.30
45	14.00	14.00	15.00	14.00	44.00	13.00	13.00	5.60
50	18.00	19.00	20.00	18.00	57.00	17.00	17.00	7.20
55	23.00	24.00	25.00	24.00	74.00	22.00	22.00	9.20
60	29.00	30.00	32.00	29.00	96.00	28.00	28.00	12.00
65	36.00	37.00	40.00	35.00	130.00	36.00	37.00	15.00
70	43.00	43.00	48.00	39.00	180.00	49.00	51.00	20.00

Excess hydrocarbons emissions (tons/million of stops) for a speed stopped from and and returned to.

speed (mph)	small car	medium car	large car	pickup	2 axles S U	3 axles S U	2-S2 semi	3-S2 semi
5	0.00	0.00	0.00	0.00	0.05	0.07	0.03	0.02
10	0.02	0.01	0.01	0.01	0.14	0.22	0.08	0.06
15	0.08	0.07	0.07	0.07	0.31	0.49	0.18	0.13
20	0.22	0.21	0.21	0.21	0.55	0.88	0.32	0.23
25	0.46	0.44	0.45	0.44	0.86	1.40	0.51	0.37
30	0.82	0.80	0.81	0.80	1.30	2.10	0.74	0.54
35	1.30	1.30	1.30	1.30	1.80	2.90	1.00	0.75
40	2.00	2.00	2.00	2.00	2.40	3.90	1.40	1.00
45	2.90	2.90	2.90	2.90	3.10	5.10	1.90	1.30
50	4.00	4.00	4.00	4.00	4.00	6.70	2.40	1.70
55	5.40	5.40	5.30	5.40	5.20	8.60	3.10	2.20
60	7.10	7.00	7.00	7.00	6.80	11.00	4.10	2.80
65	9.00	9.00	8.90	8.70	9.00	14.00	5.40	3.60
70	11.00	11.00	11.00	9.80	13.00	19.00	7.30	4.70

Excess nitrous oxides emissions (tons/million of stops) for a speed stopped from and and returned to.

speed (mph)	small car	medium car	large car	pickup	2 axles S U	3 axles S U	2-S2 semi	3-S2 semi
5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	0.08	0.06	0.07	0.06	0.16	0.64	1.00	1.40
15	0.45	0.41	0.42	0.41	0.35	1.40	2.20	3.20
20	1.20	1.20	1.20	1.20	0.62	2.50	4.00	5.60
25	2.60	2.50	2.50	2.50	0.97	4.10	6.30	9.00
30	4.70	4.60	4.60	4.60	1.40	6.00	9.30	13.00
35	7.60	7.50	7.50	7.50	2.00	8.40	13.00	18.00
40	12.00	11.00	11.00	11.00	2.70	11.00	18.00	25.00
45	17.00	17.00	16.00	17.00	3.50	15.00	23.00	32.00
50	23.00	23.00	23.00	23.00	4.50	25.00	39.00	53.00
55	31.00	31.00	31.00	31.00	5.90	25.00	39.00	53.00
60	41.00	41.00	40.00	41.00	7.60	32.00	51.00	67.00
65	52.00	52.00	51.00	51.00	10.00	42.00	67.00	87.00
70	64.00	62.00	65.00	57.00	14.00	56.00	92.00	120.00

## Appendix B. BOOTH Program Listing

```
'$INCLUDE: 'COMUNES.BAS'
OPEN "DATA.XOL" FOR OUTPUT AS #1
WRITE #1, 0
CLOSE #1
calcok = 0
DRIVE$ = "A"
DIR$ = ""
irenglon = 1
CLS
COLOR 9,0,0
LOCATE 9,1
Print"          BOOTH IS A PROGRAM DESIGNED TO  "
Print"          ESTIMATE THE ECONOMIC FACTORS INVOLVED"
Print"          IN THE OPERATION OF TOLL PLAZAS."
PRINT
PRINT"          BOOTH WAS DESIGNED BY "
PRINT"          Francisco Javier Torres-Verdin"
PRINT"          210 Patton Hall, Civil Engineering Department"
PRINT"          Virginia Polytechnic Institute and State University"
PRINT"          Blacksburg VA, 24061"
PRINT"
for I = 1 to 60
locate 4,(10 + i)
print""
next i
for I = 1 to 16
locate 4+i,(70)
print""
next i
for I = 1 to 60
locate 20,(70 - i)
print""
next i
for I = 1 to 16
locate (20-i),10
print""
next i
correcto =0
      GOSUB PAUSA
CLS
BEEP
```

```

LOCATE 10, 1
PRINT " MAKE SURE THAT NEITHER THE (CAPS LOCK) NOR THE"
PRINT " THE (NUM LOCK) KEY IS PRESSED WHILE USING THIS"
PRINT " PROGRAM."
GOSUB PAUSA
CHAIN"OPTIONS"

```

```

PAUSA:
LOCATE 24, 51, 1 : PRINT " PRESS ENTER TO CONTINUE "
PAUSA$ = "": WHILE PAUSA$ = "": PAUSA$ = INKEY$: WEND
CLS
RETURN

```

```

'
' OPTIONS PROGRAM
'
'

```

```

'$INCLUDE: 'COMUNES.BAS'

```

```

DIM tag$(7),title$(20)

```

```

CLS

```

```

KEY 15, CHR$(&H0) + CHR$(&H1)' Escape
KEY 16, CHR$(&H0) + CHR$(&H1C)' Enter
tag$(1) = " START A NEW PROBLEM "
tag$(2) = " FILES MANAGEMENT "
tag$(3) = " LOAD FILE "
tag$(4) = " EDIT DATA "
tag$(5) = " SAVE FILE "
tag$(6) = " PERFORM CALCULATIONS "
uu = 1
rengloni = 4
nrenglon = 6
drenglon = 3
salida$ = " THE PROGRAM"

```

```

MAINMENU:

```

```

GOSUB FRAME

```

```

MENU:

```

```

KEY(11) ON
KEY(14) ON
KEY(15) ON
KEY(16) ON
WHILE uu = 1
COLOR 2, 0, 0
ON KEY(11) GOSUB UP
ON KEY(14) GOSUB DOWN
ON KEY(15) GOSUB FINAL
ON KEY(16) GOSUB KEEP
WEND
CLS

```

```

KEY(16) OFF

```

```

LOCATE 5, 10, 1

```

```

INPUT "DID YOU SAVE YOUR FILES (Y/N)"; ysave$

```

```

ysave$ = (ysave$)

```

```

IF ysave$ = "Y" OR ysave$ = "y" THEN

```

```

FOR i = 11 TO nkeys

```

```

KEY(i) OFF

```

```

NEXT i

```

```

CLS

```

```

KILL "DATA.XOL"
END
  ELSE
    KEY(16) ON
    uu = 1
    irenglon = 5
    GOTO MAINMENU
  END IF
/*****
/*****
/*****

FINAL:
  uu = 0
RETURN
UP:
COLOR 2, 0, 0
LOCATE drenglon * (irenglon - 1) + rengloni, 30
PRINT tag$(irenglon)
IF irenglon = 1 THEN irenglon = nrenglon ELSE irenglon = irenglon - 1
COLOR 0, 2, 0
LOCATE drenglon * (irenglon - 1) + rengloni, 30
PRINT tag$(irenglon)
RETURN
DOWN:
COLOR 2, 0, 0
LOCATE drenglon * (irenglon - 1) + rengloni, 30
PRINT tag$(irenglon)
IF irenglon = nrenglon THEN irenglon = 1 ELSE irenglon = irenglon + 1
COLOR 0, 2, 0
LOCATE drenglon * (irenglon - 1) + rengloni, 30
PRINT tag$(irenglon)
RETURN
KEEP:
  SELECT CASE irenglon
    CASE 1
      calcok = 1
      irenglon = 1
      CHAIN "EDIT"
    CASE 2
      irenglon = 2
      CHAIN "FILES"
    CASE 3
      calcok = 1
      irenglon = 3
      CHAIN "LOAD"
    CASE 4
      irenglon = 4
      CHAIN "EDIT"
    CASE 5
      irenglon = 5
      CHAIN "SAVE"
    CASE 6
      irenglon = 6

```

```

        CHAIN "CASEA"
    END SELECT
RETURN

EDITA:
FRAME:
CLS
COLOR 2, 0, 0
    FOR i = 1 TO nrenglon
LOCATE drenglon * (i - 1) + rengloni, 30, 0
PRINT tag$(i)
    NEXT i
LOCATE 3, 2, 1
PRINT title$(1)
LOCATE 1, 1, 1: PRINT CHR$(201);
FOR i = 2 TO 79: PRINT CHR$(205); : NEXT i
PRINT CHR$(187)
    FOR i = 2 TO 23
LOCATE i, 1, 1: PRINT CHR$(186)
LOCATE i, 80, 1: PRINT CHR$(186)
    NEXT i
LOCATE 21, 1, 1
PRINT CHR$(204);
FOR i = 2 TO 79: PRINT CHR$(205); : NEXT i
PRINT CHR$(185)
LOCATE 22, 2, 1
PRINT " PRESS : (ENTER) TO SELECT, (ESC) TO EXIT"; salida$
LOCATE 23, 2, 1
PRINT " USE ARROW KEYS TO MOVE ACROSS THE SCREEN. ";
LOCATE 24, 1, 1
PRINT CHR$(200);
FOR i = 2 TO 79: PRINT CHR$(205); : NEXT i
PRINT CHR$(188);
COLOR 0, 2, 0
LOCATE drenglon * (irenglon - 1) + rengloni, 30
PRINT tag$(irenglon)
RETURN

PAUSA:
    LOCATE 24, 51, 1 : PRINT " PRESS ENTER TO CONTINUE "
    PAUSA$ = "": WHILE PAUSA$ = "": PAUSA$ = INKEY$: WEND
    CLS
RETURN
'
'   FILES MANAGEMENT PROGRAM
'
'
'$INCLUDE: 'COMUNES.BAS'
CLS
ARCHIVOS:
GOSUB MARCO
GOSUB CHOICES
    ARCH$ = ""
    WHILE ARCH$ = ""
        ARCH$ = INKEY$
    WEND

```

```

SELECT CASE ARCH$
CASE "1"
  GOSUB FLIST
CASE "2"
  GOSUB FDRIVE
CASE "3"
  GOSUB FDIR
CASE ELSE
  IF ARCH$ = CHR$(27) THEN
    CHAIN "OPTIONS"
  ELSE
    LOCATE 18,5
    PRINT"INVALID CHOICE, TRY AGAIN!"
    GOTO ARCHIVOS
  END IF
END SELECT

```

```

FDRIVE:
COLOR 4, 0, 4
CLS
LOCATE 5, 1, 1
DRIVE$ = DRIVE$
PRINT "  THE DEFAULT DRIVE IS (A,B or C):  ";
COLOR 0, 4, 0
PRINT " "; DRIVE$; " "
COLOR 4, 0, 4
AR1: LOCATE 10, 5
INPUT "  THE NEW DEFAULT DRIVE IS (A,B or C) ", DRIVE$
IF DRIVE$ = "a" THEN DRIVE$ = "A"
IF DRIVE$ = "b" THEN DRIVE$ = "B"
IF DRIVE$ = "c" THEN DRIVE$ = "C"
IF DRIVE$ <> "A" AND DRIVE$ <> "B" AND DRIVE$ <> "C" THEN
  BEEP
  LOCATE 11, 5
  PRINT "                INVALID DRIVE "
  GOTO AR1
END IF
CLS
LOCATE 5, 1
PRINT SPC(6); "THE DEFAULT DRIVE IS"; SPC(18);
COLOR 0, 4, 0
PRINT " "; DRIVE$; " "
COLOR 4, 0, 0
LOCATE 22, 1
PRINT "  PRESS ANY KEY TO CONTINUE"
BACK$ = ""
WHILE BACK$ = "": BACK$ = INKEY$: WEND
CLS
RETURN ARCHIVOS

```

```

FDIR:
COLOR 4, 0, 4
CLS
LOCATE 15, 1
PRINT "  THE CURRENT DIRECTORY IS ";

```

```

COLOR 0, 4, 0
PRINT "\";DIRR$
COLOR 4, 0, 0
LOCATE 17, 1
LOCATE 19, 1
INPUT " THE NEW DEFAULT DIRECTORY IS \", DIRR$
IF DIRR$ = "" THEN DIR$ = DIRR$ ELSE DIR$ = DIRR$ + "\"
CLS
LOCATE 15, 1
PRINT " THE CURRENT DIRECTORY IS :";
COLOR 0, 4, 0
PRINT "\";DIRR$
COLOR 4, 0, 0
LOCATE 22, 1
PRINT " PRESS ANY KEY TO CONTINUE"
BACK$ = ""
WHILE BACK$ = "": BACK$ = INKEY$: WEND
CLS
RETURN ARCHIVOS

```

```

FLIST:
COLOR 4, 0, 4
CLS
shell$ = " dir/w " + DRIVE$ + ":\\" + DIR$ + "*.BTH"
SHELL shell$
LOCATE 22, 1
PRINT " PRESS ANY KEY TO CONTINUE"
BACK$ = ""
WHILE BACK$ = "": BACK$ = INKEY$: WEND
CLS
RETURN ARCHIVOS

```

```

MARCO:
COLOR 4, 0, 0
LOCATE 1, 1, 1: PRINT CHR$(201);
FOR i = 2 TO 79: PRINT CHR$(205); : NEXT i
PRINT CHR$(187)
FOR i = 2 TO 23
  LOCATE i, 1, 1: PRINT CHR$(186)
  LOCATE i, 80, 1: PRINT CHR$(186)
NEXT i
LOCATE 24, 1, 1
PRINT CHR$(200);
FOR i = 2 TO 79: PRINT CHR$(205); : NEXT i
PRINT CHR$(188);
LOCATE 15, 1, 1
RETURN

```

```

CHOICES:
LOCATE 5, 20, 1
PRINT " FILES OPTIONS"
LOCATE 10, 5
PRINT "1. LIST FILES"
LOCATE 12, 5
PRINT "2. SET DEFAULT DRIVE"
LOCATE 14, 5

```

```

PRINT "3. SET DEFAULT DIRECTORY"
LOCATE 22, 5
PRINT "PRESS (ESC) TO RETURN TO MAIN MENU"
LOCATE 16,15
RETURN
'
'   LOAD FILE PROGRAM
'
'
'$INCLUDE: 'COMUNES.BAS'

CLS
ON ERROR GOTO ERRORES
  shell$ = " dir/w " + DRIVE$ + ":\" + DIR$ + "*.BTH"
SHELL shell$
LOCATE 18, 10
  IF file1$ < > "" THEN
    PRINT " THE CURRENT FILE USED IS : "; file1$
  ELSE
    PRINT " NO FILE IS CURRENTLY LOADED"
  END IF
LOCATE 22, 10
PRINT " PRESS ENTER TO RETURN TO MENU"
LOAD1:
  LOCATE 20, 10
  INPUT " NAME OF THE FILE TO LOAD (EXTENSION .BTH ASSUMED )", file$
  IF file$ < > "" THEN
    calcok = 1
    file1$ = file$ + ".BTH"
    file2$ = DRIVE$ + ":\" + DIR$ + file$ + ".BTH"
    OPEN file2$ FOR INPUT AS #1
      INPUT #1, nqi, dt, t0
        tmax = nqi * dt + t0
        ht0 = INT(t0)
        mt0 = CINT(60 * (t0 - ht0))
        htmax = INT(tmax)
        mtmax = CINT(60 * (tmax - htmax))
        hdt = INT(dt)
        mdt = CINT(60 * (dt - hdt))
        irmax = (tmax - t0) / dt

      INPUT #1, nqi
        qmax = 0
        tq(1) = t0
    FOR iq = 1 TO nqi
      INPUT #1, qi(iq), tq(iq + 1)
      IF qi(iq) > qmax THEN qmax = qi(iq)
    NEXT iq

    INPUT #1, srec, necb, nomb, srb
    INPUT #1, ptrucks, phov
    INPUT #1, cpFuel, tpFuel
    INPUT #1, tspstop, cspstop
    INPUT #1, accrate, fatalcost, nofatcost, propdcost
    INPUT #1, cpOil, tpOil
    INPUT #1, csetTire, tsetTire
    INPUT #1, cRepvalue, tRepvalue

```

```

INPUT #1, carvalue, truckvalue
INPUT #1, cvaluetime, tvaluetime
INPUT #1, opcost, feecar, feetruck
INPUT #1, tacc, cacc, tdec, cdec
CLOSE #1
OPEN "DATA.XOL" FOR OUTPUT AS #2
    FOR iq = 1 TO nqi
        WRITE #2, qi(iq), tq(iq + 1)
    NEXT iq
CLOSE #2
END IF

ERRORES:
CONST FILENOTFOUND = 53
CONST BADFILENAME = 64
SELECT CASE ERR
CASE FILENOTFOUND
LOCATE 21, 50, 1
BEEP
PRINT " FILE NOT FOUND, TRY AGAIN "
LOCATE 20, 10, 1
PRINT "
RESUME LOAD1
CASE BADFILENAME, 52
LOCATE 21, 50, 1
BEEP
PRINT " BAD FILE NAME, TRY AGAIN "
LOCATE 20, 10, 1
PRINT "
RESUME LOAD1
CASE ELSE
ON ERROR GOTO 0
END SELECT
CHAIN "OPTIONS"
END
'
' EDIT DATA PROGRAM
'
'$INCLUDE: 'COMUNES.BAS'
'$DYNAMIC
DIM tag$(8)
DIM title$(24)
DIM num$(1, 1), pal$(1, 1), dato$(1, 1)
KEY 15, CHR$(&H0) + CHR$(&H1)' Escape
KEY 16, CHR$(&H0) + CHR$(&H1C)' Enter
uu = 1
iirenglon = 1
rengloni = 5
nrenglon = 8
drenglon = 2
salida$ = " TO MAIN MENU"
tag$(1) = " PERIOD OF STUDY "
tag$(2) = " VEHICULAR DEMAND "
tag$(3) = " TOLL PLAZA CAPACITY "
tag$(4) = " VEHICULAR COMPOSITION "

```

```

tag$(5) = " VALUE OF TIME AND SPEED "
tag$(6) = "     ACCIDENTS     "
tag$(7) = "     VEHICLE COSTS     "
tag$(8) = "     TOLL PLAZA OPERATION     "
salida$ = " TO MAIN MENU"
    IF irenglon = 1 THEN
KEY(16) OFF
    ht0 = 0: hdt = 0: htmax = 0
    mt0 = 0: mdt = 0: mtmax = 0
    nqi = 0: nqi = 0
    CLS
    BEEP
    LOCATE 10, 1, 0
    PRINT SPC(25); " Warning: This program gives some of the "
    PRINT SPC(25); "data values by default, intending to provide"
    PRINT SPC(25); "a hint to the user about values that might"
    PRINT SPC(25); "be difficult to obtain."
    PRINT SPC(25); " Cost values are given in 1988 dollars, and"
    PRINT SPC(25); "are applicable to a general situation. It is"
    PRINT SPC(25); "recommended that the user input values which"
    PRINT SPC(25); "will fit better to the specific situation he"
    PRINT SPC(25); "or she is interested to analyze"
GOSUB PAUSA
    srec = 800: srb = 700
    cpFuel = 1.01: tpFuel = .89
    accrate = 1.589E-02: fatalcost = 654350
    nofatcost = 31100: propdcost = 1350
    cpOil = 2.5: tpOil = 1!
    csetTire = 105: tsetTire = 636
    cRepvalue = 70.7: tRepvalue = 212
    carvalue = 12190: truckvalue = 61736
    cvaluetime = 8.3: tvaluetime = 16
    necb = 0 : nmb = 0
    ptrucks = 0: phov = 0
    tspstop = 55: cspstop = 65
    opcost = 0: feecar = 0: feetruck = 0
    cacc = 5: tacc = 2
    cdec = 10: tdec = 7
    KEY(16) ON
    ELSE
    OPEN "DATA.XOL" FOR INPUT AS #1
        FOR iq = 1 TO nqi
            INPUT #1, qi(iq), tq(iq + 1)
        NEXT iq
    CLOSE #1
    END IF

MENU:
GOSUB FRAME
KEY(11) ON
KEY(14) ON
KEY(15) ON
KEY(16) ON
WHILE uu = 1
COLOR 14, 0, 0
ON KEY(11) GOSUB UP

```

```

ON KEY(14) GOSUB DOWN
ON KEY(15) GOSUB FINAL
ON KEY(16) GOSUB KEEP
WEND
OPEN "DATA.XOL" FOR OUTPUT AS #2
  FOR iq = 1 TO nqi
    WRITE #2, qi(iq), tq(iq + 1)
  NEXT iq
CLOSE #2

CHAIN "OPTIONS"
END

FINAL:
  uu = 0
RETURN

UP:
COLOR 14, 0, 0
LOCATE drenglon * (iirenglon - 1) + rengloni, 30
PRINT tag$(iirenglon)
IF iirenglon = 1 THEN iirenglon = nrenglon ELSE iirenglon = iirenglon - 1
COLOR 0, 14, 0
LOCATE drenglon * (iirenglon - 1) + rengloni, 30
PRINT tag$(iirenglon)
RETURN

DOWN:
COLOR 14, 0, 0
LOCATE drenglon * (iirenglon - 1) + rengloni, 30
PRINT tag$(iirenglon)
IF iirenglon = nrenglon THEN iirenglon = 1 ELSE iirenglon = iirenglon + 1
COLOR 0, 14, 0
LOCATE drenglon * (iirenglon - 1) + rengloni, 30
PRINT tag$(iirenglon)
RETURN

KEEP:
  ON iirenglon GOSUB GENERAL, FLOWIN, FLOWOUT, VEHCOMP, FUEL, ACCIDENTS, COST,
OPERATION
RETURN MENU

FRAME:
CLS
COLOR 14, 0, 0
  FOR i = 1 TO nrenglon
    LOCATE drenglon * (i - 1) + rengloni, 30, 0
    PRINT tag$(i)
  NEXT i
LOCATE 3, 2, 1
PRINT title$(1)
LOCATE 1, 1, 1: PRINT CHR$(201);
FOR i = 2 TO 79: PRINT CHR$(205); : NEXT i
PRINT CHR$(187)
  FOR i = 2 TO 23
    LOCATE i, 1, 1: PRINT CHR$(186)
  
```

```

LOCATE i, 80, 1: PRINT CHR$(186)
NEXT i
LOCATE 21, 1, 1
PRINT CHR$(204);
FOR i = 2 TO 79: PRINT CHR$(205); : NEXT i
PRINT CHR$(185)
LOCATE 22, 2, 1
PRINT " PRESS : (ENTER) TO SELECT, (ESC) TO EXIT"; salida$
LOCATE 23, 2, 1
PRINT " USE ARROW KEYS TO MOVE ACROSS THE SCREEN. ";
LOCATE 24, 1, 1
PRINT CHR$(200);
FOR i = 2 TO 79: PRINT CHR$(205); : NEXT i
PRINT CHR$(188);
COLOR 0, 14, 0
LOCATE drenglon * (iirenglon - 1) + rengloni, 30
PRINT tag$(iirenglon)
RETURN

```

GENERAL:

```

REDIM num$(4, 2), num(4, 2), dato$(4, 2)
irmax = 3
ecrans = 0
dataca$ = "tiempo"
ri = 8 ' Initial Row
rows = 3' Number of rows per screen
cols = 2' Number of columns per row
dr = 3' Space between rows
dc = 30
mot = 25
mnc = 2
title$(4) = " PERIOD OF STUDY"
ci = 5 ' Initial Column
ici = 1
dato$(1, 1) = " INITIAL TIME hours:"
dato$(2, 1) = " ENDING TIME hours:"
dato$(3, 1) = " TIME INCREMENT hours:"
FOR k = 1 TO 3
dato$(k, 2) = " mins:"
NEXT k
num(1, 1) = ht0: num(2, 1) = htmax: num(3, 1) = hdt
num(1, 2) = mt0: num(2, 2) = mtmax: num(3, 2) = mdt
GOSUB EDITOR
ht0 = num(1, 1): htmax = num(2, 1): hdt = num(3, 1)
mt0 = num(1, 2): mtmax = num(2, 2): mdt = num(3, 2)
IF mt0 >= 60 THEN mt0 = 0
IF mtmax >= 60 THEN mtmax = 0
IF mdt >= 60 THEN mdt = 0
t0 = ht0 + mt0 / 60
tmax = htmax + mtmax / 60
dt = hdt + mdt / 60
IF dt <> 0 THEN nqi = (tmax - t0) / dt
nqi = CINT(nqi)
tmax = nqi * dt + t0
RETURN

```

FLOWIN:

```

REDIM num$(nqi, 1), num(nqi, 1), dato$(nqi, 1)
IF dt <= 0 THEN RETURN
irmax = (tmax - t0) / dt
ecrans = 1
dataca$ = "flow"
ri = 5 ' Initial Row
rows = 7' Number of rows per screen
ci = 10 ' Initial Column
cols = 1 ' Number of columns per row
dr = 2' Space between rows
mnc = 6
mot = 20
ici = 1
title$(4) = "                      VEHICULAR DEMAND VALUES"
FOR i = 1 TO nqi
tta = t0 + dt * (i - 1): ttb = t0 + dt * (i)
ttah = INT(tta): ttbh = INT(ttb)
ttam = CINT((tta - ttah) * 60): ttbm = CINT((ttb - ttbh) * 60)
IF ttah < 10 THEN ttah$ = " " + STR$(ttah) ELSE ttah$ = STR$(ttah)
IF ttbh < 10 THEN ttbh$ = " " + STR$(ttbh) ELSE ttbh$ = STR$(ttbh)
ttam$ = STR$(ttam)
ttbm$ = STR$(ttbm)
IF ttam < 10 THEN ca$ = "0" ELSE ca$ = ""
IF ttbm < 10 THEN cb$ = "0" ELSE cb$ = ""
dato$(i, 1) = "from " + MID$(ttah$, 2, 2) + ":" + ca$ + MID$(ttam$, 2, 2) + " to " +
MID$(ttbh$, 2, 2) + ":" + cb$ + MID$(ttbm$, 2, 2) + " "
num(i, 1) = qi(i)
NEXT i

GOSUB EDITOR
tq(1) = t0
FOR i = 1 TO nqi
qi(i) = num(i, 1)
tq(i + 1) = i * dt + t0
num(i, 1) = 0
NEXT i
RETURN

```

FLOWOUT:

```

REDIM num$(2, 2), num(2, 2), dato$(2, 2)
irmax = 2
ecrans = 0
dataca$ = "flow"
ri = 8 ' Initial Row
rows = 2' Number of rows per screen
cols = 2' Number of columns per row
ci = 5 ' Initial Column
dr = 4' Space between rows
mnc = 6
mot = 25
ici = 1
dc = 30
title$(4) = "                      TOLL PLAZA CAPACITY "
dato$(1, 1) = "          # of exact change booths"
dato$(1, 2) = "          Capacity (veh/hr)"

```

```

    dato$(2, 1) = "          # of manned booths"
    dato$(2, 2) = "          Capacity (veh/hr)"
    num(1, 2) = srec
    num(1, 1) = necb
    num(2, 1) = nmb
    num(2, 2) = srb
GOSUB EDITOR
    srec = num(1, 2)
    necb = num(1, 1)
    nmb = num(2, 1)
    srb = num(2, 2)
    qo = (srec * necb + nmb * srb) * dt
RETURN

VEHCOMP:
REDIM num$(2, 1), num(2, 1), dato$(2, 1)
    irmax = 2
    ecrans = 0
    ri = 8
    rows = 2
    cols = 1
    dr = 3
    mot = 50
    mnc = 3
    title$(4) = "          VEHICULAR COMPOSITION"
    ci = 3
    ici = 1
    dato$(1, 1) = "          Percentage of Heavy Vehicles      :"
    dato$(2, 1) = "          Percentage of High Occupancy Vehicles:"
    num(1, 1) = ptrucks
    num(2, 1) = phov
GOSUB EDITOR
    ptrucks = num(1, 1)
    phov = num(2, 1)
    title$(4) = ""
RETURN

FUEL:
REDIM num$(4, 2), num(4, 2), dato$(4, 2)
    irmax = 4
    ecrans = 0
    ri = 6
    rows = 4
    cols = 2
    dr = 3
    dc = 30
    mot = 22
    mnc = 5
    title$(4) = "          VALUE OF TIME AND SPEED      "
    title$(6) = "          Value of Time (dollars/hr)  "
    title$(9) = "          Speed stopped from (mph)    "
    title$(12) = "          Acceleration rate (mph/sec) "
    title$(15) = "          Deceleration rate (mph/sec) "
    ci = 3
    ici = 1
    dato$(1, 1) = "          Light Vehicles      "

```

```

dato$(1, 2) = "          Heavy Vehicles "
dato$(2, 1) = "          Light Vehicles "
dato$(2, 2) = "          Heavy Vehicles "
dato$(3, 1) = "          Light Vehicles "
dato$(3, 2) = "          Heavy Vehicles "
dato$(4, 1) = "          Light Vehicles "
dato$(4, 2) = "          Heavy Vehicles "
num(1, 1) = cvaluetime: num(1, 2) = tvaluetime
num(2, 1) = cspstop: num(2, 2) = tspstop
num(3, 1) = cacc: num(3, 2) = tacc
num(4, 1) = cdec: num(4, 2) = tdec
GOSUB EDITOR
cvaluetime = num(1, 1): tvaluetime = num(1, 2)
cspstop = num(2, 1): tspstop = num(2, 2)
cacc = num(3, 1): tacc = num(3, 2)
cdec = num(4, 1): tdec = num(4, 2)
REDIM title$(25)
RETURN

ACCIDENTS:
REDIM num$(4, 1), num(4, 1), dato$(4, 1)
irmax = 4
ecrans = 0
ri = 6
rows = 4
cols = 1
dr = 3
dc = 38
mot = 50
mnc = 8
ci = 3
ici = 1
title$(4) = "          ACCIDENT RATE AND COSTS "
dato$(1, 1) = "          Rear-End Collisions per million of vehs:"
dato$(2, 1) = "          Fatal Accident Cost :          "
dato$(3, 1) = "          Non-Fatal Accident Cost :      "
dato$(4, 1) = "          Property Damage Accident Cost : "
num(1, 1) = accrate
num(2, 1) = fatalcost
num(3, 1) = nofatcost
num(4, 1) = propdcost
GOSUB EDITOR
accrate = num(1, 1)
fatalcost = num(2, 1)
nofatcost = num(3, 1)
propdcost = num(4, 1)
RETURN

COST:
REDIM num$(5, 2), num(5, 2), dato$(5, 2)
irmax = 5
ecrans = 0
ri = 7
rows = 5
cols = 2
dr = 3

```

```

dc = 32
mot = 22
mnc = 8
ci = 3
ici = 1
dato$(1, 1) = "          Cost per quart      : "
dato$(1, 2) = "          "
dato$(2, 1) = "          Set of tires cost    : "
dato$(2, 2) = "          "
dato$(3, 1) = "          Cost per 1000 miles : "
dato$(3, 2) = "          "
dato$(4, 1) = "          New Vehicle Cost   : "
dato$(4, 2) = "          "
dato$(5, 1) = "          Cost per gallon    : "
dato$(5, 2) = "          "
title$(4) = "          VEHICLE COSTS "
title$(6) = "          LIGHT VEHICLES      HEAVY VEHICLES"
title$(7) = "          LUBRICATING OIL "
title$(10) = "          TIRE WEAR "
title$(13) = "          MAINTENANCE"
title$(16) = "          DEPRECIATION"
title$(19) = "          FUEL          "
num(1, 1) = cpOil: num(1, 2) = tpOil
num(2, 1) = csetTire: num(2, 2) = tsetTire
num(3, 1) = cRepvalue: num(3, 2) = tRepvalue
num(4, 1) = carvalue: num(4, 2) = truckvalue
num(5, 1) = cpFuel: num(5, 2) = tpFuel
GOSUB EDITOR
FOR i = 4 TO 19: title$(i) = "": NEXT i
cpOil = num(1, 1): tpOil = num(1, 2)
csetTire = num(2, 1): tsetTire = num(2, 2)
cRepvalue = num(3, 1): tRepvalue = num(3, 2)
carvalue = num(4, 1): truckvalue = num(4, 2)
cpFuel = num(5, 1): tpFuel = num(5, 2)
RETURN

OPERATION:
REDIM num$(3, 1), num(3, 1), dato$(3, 1)
  irmax = 3
  ecrans = 0
  ri = 6
  rows = 3
  cols = 1
  dr = 3
  mot = 45
  mnc = 5
  ci = 3
  ici = 1
  title$(4) = "          TOLL PLAZA OPERATION "
  dato$(1, 1) = "          Toll Plaza Operation cost (dollars/hour): "
  dato$(2, 1) = "          Toll per light vehicle (dollars/veh): "
  dato$(3, 1) = "          Toll per heavy vehicle (dollars/veh): "
  num(1, 1) = opcost: num(2, 1) = feecar: num(3, 1) = feetruck
GOSUB EDITOR
  opcost = num(1, 1): feecar = num(2, 1): feetruck = num(3, 1)
RETURN

```

```

PAUSA:
  LOCATE 24, 1, 1
  PRINT SPC(52); "ZDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD?"
  PRINT SPC(52); "3 PRESS ENTER TO CONTINUE 3"
  PRINT SPC(52); "@DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDY"
  PAUSA$ = "": WHILE PAUSA$ = "": PAUSA$ = INKEY$: WEND
  CLS
RETURN

```

```

EDITOR:
  CLS
  COLOR 3, 0, 0
  '
  ' Parameters to be read by the Subroutine
  '
  ' ecrans = 1
  ' dataca$ = "flow"
  ' rows Number of rows per screen
  ' cols = Number of columns per row
  ' dr = Space between rows
  ' ici = ' Column from which the editing process is initiated
  ' dc = ' Space between columns
  ' mot = 'Width of word
  ' mnc = 'Number of characters per number read
  '   ir = 1' ir is the index for the rows
  '   ic = 1 ' ic is the index for the columns
  '   i1 = 1
  '   coli = ci + (ici - 1) * dc - 1
  ' Create Frame
  GOSUB MARCO
  GOSUB TABLA
  BORRA$ = ""
  FOR i = 1 TO mnc
    BORRA$ = "" + BORRA$
  NEXT i
  '
  ' MAIN SUBROUTINE
  '
  '
  row = ri
  col1 = ci + mot + 1

  "/>

```

```

    nedit = 1
    WHILE nedit = 1
    INICIO:
        ind = 1
        SELECT CASE (INKEY$)
    CASE "1"
        CALL CREANUM("1")
    CASE "2"
        CALL CREANUM("2")
    CASE "3"
        CALL CREANUM("3")
    CASE "4"
        CALL CREANUM("4")
    CASE "5"
        CALL CREANUM("5")
    CASE "6"
        CALL CREANUM("6")
    CASE "7"
        CALL CREANUM("7")
    CASE "8"
        CALL CREANUM("8")
    CASE "9"
        CALL CREANUM("9")
    CASE "0"
        CALL CREANUM("0")
    CASE "."
        CALL CREANUM(".")
    CASE ELSE
        ind = 0
        ON KEY(11) GOSUB ARRIBA
        ON KEY(12) GOSUB IZQUIERDA
        ON KEY(13) GOSUB DERECHA
        ON KEY(14) GOSUB ABAJO
        ON KEY(15) GOSUB FIN
        ON KEY(16) GOSUB GUARDA
        ON KEY(17) GOSUB RETROCEDE
        ON KEY(18) GOSUB REPITE
        ON KEY(20) GOSUB ADELANTE
        ON KEY(19) GOSUB ATRAS
        END SELECT
        WEND
    '
    ' HERE IT RETURNS TO THE PLACE WHERE THE SUBROUTINE WAS CALLED
    RETURN

    MARCO:
    LOCATE 3, 2, 1
    FOR i = 1 TO 24: PRINT title$(i): NEXT i
    LOCATE 1, 1, 1
    PRINT CHR$(201);
    FOR i = 2 TO 79: PRINT CHR$(205); : NEXT i
    PRINT CHR$(187)
    FOR i = 2 TO 23
    LOCATE i, 1, 1: PRINT CHR$(186)
    LOCATE i, 80, 1: PRINT CHR$(186)
    NEXT i

```

```

LOCATE 20, 1, 1
PRINT CHR$(204);
FOR i = 2 TO 79: PRINT CHR$(205); : NEXT i
PRINT CHR$(185)
LOCATE 21, 2, 1
PRINT " PRESS: (ENTER) TO STORE, (ESC) TO RETURN TO DATA MENU, "
LOCATE 22, 2, 1
PRINT " (BACKSPACE) TO CORRECT, (INS) TO STORE PREVIOUS VALUE."
LOCATE 23, 2, 1
PRINT " USE ARROW KEYS TO MOVE ACROSS THE SCREEN. ";
IF ecrans = 1 THEN PRINT " PgUp AND PgDn TO CHANGE SCREEN"
LOCATE 24, 1, 1
PRINT CHR$(200);
FOR i = 2 TO 79: PRINT CHR$(205); : NEXT i
PRINT CHR$(188);
RETURN

```

ARRIBA:

```

num$(ir, ic) = ""
LOCATE row, col - 1, 1
GOSUB BORRA
PRINT num(ir, ic)
IF ir = i1 THEN
ir = i1 + rows - 1
row = ri + (rows - 1) * dr
ELSE
ir = ir - 1
row = row - dr
END IF
LOCATE row, col - 1, 1
PRINT num(ir, ic)
LOCATE row, col, 1
unpto = 0
RETURN

```

ABAJO:

```

num$(ir, ic) = ""
LOCATE row, col - 1, 1
GOSUB BORRA
PRINT num(ir, ic)
IF ir = i1 + rows - 1 OR ir = irmax THEN
ir = i1
row = ri
ELSE
ir = ir + 1
row = row + dr
END IF
LOCATE row, col - 1, 1
PRINT num(ir, ic)
LOCATE row, col, 1
unpto = 0
RETURN

```

IZQUIERDA:

```

num$(ir, ic) = ""
LOCATE row, col - 1, 1

```

```

GOSUB BORRA
PRINT num(ir, ic)
IF ic = ici THEN
  ic = cols
  col = col1 + (cols - 1) * dc
ELSE
  ic = ic - 1
  col = col - dc
END IF
LOCATE row, col - 1, 1
PRINT num(ir, ic)
LOCATE row, col, 1
unpto = 0
RETURN

DERECHA:
num$(ir, ic) = ""
LOCATE row, col - 1, 1
GOSUB BORRA
PRINT num(ir, ic)
IF ic = cols THEN
  ic = ici
  col = coli + mot + 2
ELSE
  ic = ic + 1
  col = col + dc
END IF
LOCATE row, col - 1, 1
PRINT num(ir, ic)
LOCATE row, col, 1
unpto = 0
RETURN

REPITE:
IF ir > 1 THEN num(ir, ic) = num(ir - 1, ic)
GOSUB GUARDA
RETURN

GUARDA:
IF num$(ir, ic) <> "" THEN num(ir, ic) = VAL(num$(ir, ic))
num$(ir, ic) = ""
IF ic = cols THEN
  num$(ir, ic) = ""
  LOCATE row, col - 1, 1
  GOSUB BORRA
  PRINT num(ir, ic)
  ic = ici
  col = coli + mot + 2
  GOSUB ABAJO
ELSE
  GOSUB DERECHA
END IF
RETURN

BORRA:

```

```

nc = 0
cx = POS(1)
ry = CSRLIN
PRINT BORRA$
LOCATE ry, cx
RETURN

```

RETROCEDE:

```

IF num$(ir, ic) = "" THEN RETURN
LOCATE row, POS(1) - 1, 1

PRINT " ";
LOCATE row, POS(1) - 1, 1
extra$ = MID$(num$(ir, ic), nc, 1)
IF extra$ = "." THEN unpto = 0
nc = nc - 1
num$(ir, ic) = MID$(num$(ir, ic), 1, nc)

```

RETURN

TABLA:

```

LOCATE 24, 60, 1
IF ecrans = 1 THEN ecran = CINT(((i1 - 1) / (rows - 1)) + 1)
IF ecrans = 1 THEN PRINT " SCREEN"; ecran;
FOR i = i1 TO i1 + rows - 1
  FOR j = 1 TO cols
    IF i <= irmax THEN
      wi = LEN(dato$(i, j))
      IF wi < mot THEN wi = mot + 1
      dato$(i, j) = MID$(dato$(i, j), wi - mot, mot + 1)
      LOCATE ri + ((i - i1 + 1) - 1) * dr, ci + (j - 1) * dc - 1, 1
      PRINT dato$(i, j);
      GOSUB BORRA
    END IF
    LOCATE CSRLIN, POS(1), 1
    IF i = i1 AND j = ici THEN
      iii = CSRLIN
      jjj = POS(1) + 1
    END IF
  IF i <= irmax THEN PRINT num(i, j)
NEXT j
NEXT i

```

SIG:

```

LOCATE iii, jjj, 1
row = ri
col = jjj
ir = i1
ic = ici

```

RETURN

ADELANTE:

```

ii1 = i1 + rows - 1
IF ii1 >= nqi THEN RETURN
CLS
GOSUB MARCO
i1 = ii1
ir = i1

```

```

GOSUB TABLA
RETURN

ATRAS:
CLS
GOSUB MARCO
i1 = i1 - rows + 1
IF i1 < 1 THEN i1 = 1
ir = i1
GOSUB TABLA
RETURN

FIN:
CLS
title$(1) = ""
nedit = 0
KEY(17) OFF
KEY(18) OFF
KEY(19) OFF
KEY(20) OFF
RETURN

SUB CREANUM (n$) STATIC
SHARED num$(), nc, unpto, mnc, BORRA$, ir, ic

    IF num$(ir, ic) = "" THEN

nc = 0
cx = POS(1)
ry = CSRLIN
PRINT BORRA$
LOCATE ry, cx
    END IF
    nc = nc + 1
    IF nc <= mnc THEN
IF n$ = "." THEN
    unpto = unpto + 1
    IF unpto = 1 THEN
        PRINT ".";
        num$(ir, ic) = num$(ir, ic) + "."
    ELSE
        nc = nc - 1
    END IF
ELSE
    PRINT n$;
    num$(ir, ic) = num$(ir, ic) + n$
END IF
    END IF

END SUB ' '    CALCULATIONS PROGRAM '

'$INCLUDE: 'COMUNES.BAS'
'$DYNAMIC
DIM COST(1), revenue(1), que(1),fcmin(1),ftmin(1)
DIM speed(15)

```

```
DIM tCO(15), tHC(15), tNOX(15), tFuel(15), tOil(15), tTire(15), tDep(15), tRep(15)
DIM cCO(15), cHC(15), cNOX(15), cFuel(15), cOil(15), cTire(15), cDep(15), cRep(15)
shell "graphics"
```

```
'Speed stopped from and returned to Values
```

```
speed(1) = 0
speed(2) = 5
speed(3) = 10
speed(4) = 15
speed(5) = 20
speed(6) = 25
speed(7) = 30
speed(8) = 35
speed(9) = 40
speed(10) = 45
speed(11) = 50
speed(12) = 55
speed(13) = 60
speed(14) = 65
speed(15) = 70
```

```
' Stopping costs values for trucks
```

```
tFuel(1) = 0!: tOil(1) = 0!: tTire(1) = 0!: tRep(1) = 0!
tFuel(2) = 6.09: tOil(2) = .039: tTire(2) = .03: tRep(2) = .03
tFuel(3) = 11.45: tOil(3) = .066: tTire(3) = .12: tRep(3) = .11
tFuel(4) = 18.42: tOil(4) = .088: tTire(4) = .26: tRep(4) = .24
tFuel(5) = 26.55: tOil(5) = .11: tTire(5) = .46: tRep(5) = .44
tFuel(6) = 35.7: tOil(6) = .131: tTire(6) = .71: tRep(6) = .71
tFuel(7) = 46.25: tOil(7) = .153: tTire(7) = 1.01: tRep(7) = 1.05
tFuel(8) = 58.75: tOil(8) = .177: tTire(8) = 1.26: tRep(8) = 1.46
tFuel(9) = 73.25: tOil(9) = .201: tTire(9) = 1.55: tRep(9) = 1.95
tFuel(10) = 90.4: tOil(10) = .225: tTire(10) = 1.86: tRep(10) = 2.52
tFuel(11) = 109.53: tOil(11) = .251: tTire(11) = 2.21: tRep(11) = 3.18
tFuel(12) = 129.43: tOil(12) = .283: tTire(12) = 2.57: tRep(12) = 3.93
tFuel(13) = 150!: tOil(13) = .327: tTire(13) = 2.96: tRep(13) = 4.77
tFuel(14) = 171.75: tOil(14) = .392: tTire(14) = 3.36: tRep(14) = 5.72
tFuel(15) = 194.5: tOil(15) = .5: tTire(15) = 3.75: tRep(15) = 6.76
```

```
' Stopping costs values for cars
```

```
cFuel(1) = 0!: cOil(1) = 0!: cTire(1) = 0!: cRep(1) = 0!
cFuel(2) = .96: cOil(2) = .005: cTire(2) = .03: cRep(2) = .04
cFuel(3) = 1.88: cOil(3) = .008: cTire(3) = .11: cRep(3) = .15
cFuel(4) = 2.87: cOil(4) = .011: cTire(4) = .23: cRep(4) = .32
cFuel(5) = 4.01: cOil(5) = .013: cTire(5) = .41: cRep(5) = .57
cFuel(6) = 5.26: cOil(6) = .015: cTire(6) = .63: cRep(6) = .89
cFuel(7) = 6.66: cOil(7) = .017: cTire(7) = .89: cRep(7) = 1.29
cFuel(8) = 8.53: cOil(8) = .02: cTire(8) = 1.13: cRep(8) = 1.72
cFuel(9) = 10.55: cOil(9) = .023: cTire(9) = 1.39: cRep(9) = 2.22
cFuel(10) = 12.73: cOil(10) = .026: cTire(10) = 1.67: cRep(10) = 2.78
cFuel(11) = 15.14: cOil(11) = .03: cTire(11) = 1.97: cRep(11) = 3.39
cFuel(12) = 17.85: cOil(12) = .033: cTire(12) = 2.29: cRep(12) = 4.04
cFuel(13) = 20.99: cOil(13) = .036: cTire(13) = 2.63: cRep(13) = 4.74
cFuel(14) = 24.6: cOil(14) = .042: cTire(14) = 2.99: cRep(14) = 4.98
```

cFuel(15) = 29.63: cOil(15) = .049: cTire(15) = 3.4: cRep(15) = 6.24

' Depreciation and Emission values for trucks

tDep(1) = 0!: tHC(1) = 0!: tNOX(1) = 0!: tCO(1) = 0!  
tDep(2) = .002: tHC(2) = .04: tNOX(2) = .26: tCO(2) = .27  
tDep(3) = .003: tHC(3) = .13: tNOX(3) = .8: tCO(3) = .84  
tDep(4) = .005: tHC(4) = .28: tNOX(4) = 1.79: tCO(4) = 1.84  
tDep(5) = .006: tHC(5) = .5: tNOX(5) = 3.18: tCO(5) = 3.3  
tDep(6) = .007: tHC(6) = .79: tNOX(6) = 5.09: tCO(6) = 5.15  
tDep(7) = .008: tHC(7) = 1.17: tNOX(7) = 7.43: tCO(7) = 7.68  
tDep(8) = .01: tHC(8) = 1.61: tNOX(8) = 10.35: tCO(8) = 10.68  
tDep(9) = .011: tHC(9) = 2.18: tNOX(9) = 14.18: tCO(9) = 14.48  
tDep(10) = .013: tHC(10) = 2.85: tNOX(10) = 18.38: tCO(10) = 18.9  
tDep(11) = .014: tHC(11) = 3.7: tNOX(11) = 23.63: tCO(11) = 24.55  
tDep(12) = .016: tHC(12) = 4.78: tNOX(12) = 25.1: tCO(12) = 31.8  
tDep(13) = .018: tHC(13) = 6.18: tNOX(13) = 39.4: tCO(13) = 41!  
tDep(14) = .02: tHC(14) = 8!: tNOX(14) = 51.5: tCO(14) = 54.5  
tDep(15) = .024: tHC(15) = 11!: tNOX(15) = 70.5: tCO(15) = 75!

' Depreciation and Emission values for cars

cDep(1) = 0!: cHC(1) = 0: cNOX(1) = 0: cCO(1) = 0!  
cDep(2) = .002: cHC(2) = 0: cNOX(2) = 0: cCO(2) = .27  
cDep(3) = .003: cHC(3) = .012: cNOX(3) = .07: cCO(3) = .84  
cDep(4) = .004: cHC(4) = .074: cNOX(4) = .42: cCO(4) = 1.84  
cDep(5) = .005: cHC(5) = .21: cNOX(5) = 1.2: cCO(5) = 3.3  
cDep(6) = .007: cHC(6) = .45: cNOX(6) = 2.5: cCO(6) = 5.15  
cDep(7) = .008: cHC(7) = .81: cNOX(7) = 4.6: cCO(7) = 7.68  
cDep(8) = .009: cHC(8) = 1.3: cNOX(8) = 7.53: cCO(8) = 10.68  
cDep(9) = .01: cHC(9) = 2!: cNOX(9) = 11.25: cCO(9) = 14.48  
cDep(10) = .012: cHC(10) = 2.9: cNOX(10) = 16.75: cCO(10) = 18.9  
cDep(11) = .013: cHC(11) = 4!: cNOX(11) = 23!: cCO(11) = 24.55  
cDep(12) = .015: cHC(12) = 5.38: cNOX(12) = 31!: cCO(12) = 31.8  
cDep(13) = .016: cHC(13) = 7!: cNOX(13) = 40.75: cCO(13) = 41!  
cDep(14) = .018: cHC(14) = 8.7: cNOX(14) = 51.5: cCO(14) = 54.5  
cDep(15) = .02: cHC(15) = 9.8: cNOX(15) = 62!: cCO(15) = 75!

'  
' **FIXED RATES**  
'

' Accidents

tpfatal = 1.6/100: tpnofat = 29.1/100: tppropd = 69.9/100  
cpfatal = 1.2/100: cpnofat = 32.9/100: cppropd = 65.9/100

' Fuel

cidleFuel = 538: tidleFuel = 634

' Maintenance and Repair

cidleRepptm = 58: tidleRepptm = 24

' Oil

cidleOil = 5.23: tidleOil = 3.4

' Depreciation

cpctDep = .73: tpctDep = .74

' Emissions

cidleNOX = 0.05: tidleNOX = 0.05  
cidleCO = 2.43: tidleCO = 2.43

```

        cidleHC = 0.16: tidleHC = 0.16

OPEN "DATA.XOL" FOR INPUT AS #1
  FOR iq = 1 TO nqi
    INPUT #1, qi(iq), tq(iq + 1)
  NEXT iq
CLOSE #1

OPCIONES:
CLS
COLOR 5, 0, 0
PRINT
PRINT SPC(6); "OPTIONS "
PRINT
PRINT " 1. GET HARDCOPY OF INPUT DATA ECHO"
PRINT
PRINT " 2. GET INPUT DATA ECHO ON SCREEN"
PRINT
PRINT " 3. GET HARDCOPY OF RESULTS"
PRINT
PRINT " 4. GET RESULTS ON SCREEN"
PRINT
PRINT " 5. GET GRAPHICS OUTPUT"
PRINT
IF calcok = 0 THEN
  BEEP
  PRINT SPC(35); " Warning: To perform calculations either"
  PRINT SPC(35); "a file must be loaded or data have to be"
  PRINT SPC(35); "inputted. Please press (ESC) Key to go back"
  PRINT SPC(35); "to the Main Menu."
END IF

LOCATE 20,1
PRINT "      PRESS (ESC) TO GO BACK TO MAIN MENU"

LOCATE 1, 1, 1: PRINT CHR$(201);
FOR i = 2 TO 79: PRINT CHR$(205); : NEXT i
PRINT CHR$(187)
FOR i = 2 TO 23
  LOCATE i, 1, 1: PRINT CHR$(186)
  LOCATE i, 80, 1: PRINT CHR$(186)
NEXT i
LOCATE 24, 1, 1
PRINT CHR$(200);
FOR i = 2 TO 79: PRINT CHR$(205); : NEXT i
PRINT CHR$(188);
LOCATE 15, 1, 1
ASKY:
U$ = ""
WHILE U$ = ""
  U$ = INKEY$
WEND
  LOCATE 20, 4, 1
  SELECT CASE U$
    CASE "1"
      yew$ = "lpt1"

```

```

lp = 40
GOSUB DATAECHO
GOTO OPCIONES
CASE "2"
yew$ = "con"
lp = 8
GOSUB DATAECHO
GOTO OPCIONES
CASE "3"
yew$ = "lpt1"
lp = 40
GOSUB CALCULOS
GOSUB RESULTADOS
GOTO OPCIONES
CASE "4"
yew$ = "con"
lp = 8
GOSUB CALCULOS
GOSUB RESULTADOS
GOTO OPCIONES
CASE "5"
GOSUB CALCULOS
GOSUB GRAFICA
GOTO OPCIONES
CASE ELSE
IF U$ = CHR$(27) THEN
CHAIN "OPTIONS"
ELSE
LOCATE 15, 15, 1
BEEP
PRINT " Invalid Choice!"
GOTO ASKY
END IF
END SELECT

```

```

PAUSA:
LOCATE 23, 51, 1 : PRINT " PRESS (ENTER) TO CONTINUE, (ESC) TO RETURN TO OPTIONS"

```

```

PAUSA$ = ""
WHILE PAUSA$ = ""
PAUSA$ = INKEY$
IF PAUSA$ = CHR$(27) THEN RETURN OPCIONES
WEND
CLS
RETURN

```

```

DATAECHO:
CLS
CLOSE #5
OPEN yew$ FOR OUTPUT AS #5
PRINT #5,
PRINT #5,
PRINT #5,
PRINT #5, SPC(25); "*****"
PRINT #5, SPC(25); "** INPUT DATA ECHO **"
PRINT #5, SPC(25); "*****"

```

```

PRINT #5,
PRINT #5,
PRINT #5, SPC(18); "PERIOD OF STUDY"
PRINT #5,
mt0$ = STR$(mt0); mtmax$ = STR$(mtmax); mdt$ = STR$(mdt)
IF mt0 < 10 THEN mt0$ = "0" + MID$(mt0$, 2, 1) ELSE mt0$ = MID$(mt0$, 2, 2)
IF mtmax < 10 THEN mtmax$ = "0" + MID$(mtmax$, 2, 1) ELSE mtmax$ = MID$(mtmax$,
2, 2)
IF mdt < 10 THEN mdt$ = "0" + MID$(mdt$, 2, 1) ELSE mdt$ = MID$(mdt$, 2, 2)
PRINT #5, SPC(10); "Initial time ";
PRINT #5, USING "##"; ht0;
PRINT #5, ":";
PRINT #5, mt0$
PRINT #5, SPC(10); "Ending time ";
PRINT #5, USING "##"; htmax;
PRINT #5, ":";
PRINT #5, mtmax$
PRINT #5, SPC(10); "Time increment ";
PRINT #5, USING "##"; hdt;
PRINT #5, ":";
PRINT #5, mdt$
PRINT #5,
PRINT #5,
PRINT #5, SPC(18); "TOLL PLAZA CAPACITY"
PRINT #5,
PRINT #5, SPC(10); "# of manned booths ";
PRINT #5, USING "##"; nmb;
PRINT #5, " Capacity (veh/hr)";
PRINT #5, USING "####"; srb
PRINT #5, SPC(10); "# of exact change booths";
PRINT #5, USING "##"; necb;
PRINT #5, " Capacity (veh/hr)";
PRINT #5, USING "####"; srec
PRINT #5,
IF yew$ = "con" THEN GOSUB PAUSA
PRINT #5,
PRINT #5, SPC(18); "VEHICULAR COMPOSITION"
PRINT #5,
PRINT #5, SPC(10); "Percentage of Heavy Vehicles ";
PRINT #5, USING "##.#"; ptrucks
PRINT #5, SPC(10); "Percentage of High Occupancy Vehicles:";
PRINT #5, USING "##.#"; phov
PRINT #5,
PRINT #5,
PRINT #5, SPC(18); "VALUE OF TIME AND SPEED "
PRINT #5,
PRINT #5, SPC(35); " LIGHT VEHICLES"; SPC(5); "HEAVY VEHICLES"
PRINT #5,
PRINT #5, SPC(10); "VALUE OF TIME"
PRINT #5, SPC(10); "Dollars/hour ";
PRINT #5, USING "#####.##"; cvaluetime, tvaluetime
PRINT #5, SPC(10); "SPEED STOPPED FROM "
PRINT #5, SPC(10); "AND RESUMED TO (mph): ";
PRINT #5, USING "#####.##"; cspstop, tspstop
PRINT #5, SPC(10); "ACCELERATION RATE (mph/sec): ";
PRINT #5, USING "#####.##"; cacc, tacc

```

```

PRINT #5, SPC(10); "DECELERATION RATE (mph/sec): ";
PRINT #5, USING "#####.##"      "; cdec, tdec

PRINT #5,
PRINT #5,
IF yew$ = "con" THEN GOSUB PAUSA
PRINT #5,
PRINT #5,
PRINT #5, SPC(18); "ACCIDENTS RATE AND COSTS"
PRINT #5,
PRINT #5, SPC(10); "Accident Rate (acc/millions of vehs) "; accrate
PRINT #5, SPC(10); "Fatal Accident Cost:      ";
PRINT #5, USING "#####.##"; fatalcost
PRINT #5, SPC(10); "Non-Fatal Accident Cost:   ";
PRINT #5, USING "#####.##"; nofatcost
PRINT #5, SPC(10); "Property Damage Accident Cost:";
PRINT #5, USING "#####.##"; propdcost
PRINT #5,
PRINT #5,
PRINT #5,
PRINT #5, SPC(18); "OPERATION VALUES"
PRINT #5,
PRINT #5, SPC(10); "Toll Plaza Operation cost (dollars/hour): ";
PRINT #5, USING "#####.##"; opcost
PRINT #5, SPC(10); "Toll per light vehicle (dollars/veh):   ";
PRINT #5, USING "###.##"; feecar
PRINT #5, SPC(10); "Toll per heavy vehicle (dollars/veh):   ";
PRINT #5, USING "###.##"; feetruck
IF yew$ = "lpt1" THEN PRINT #5, CHR$(12)
IF yew$ = "con" THEN GOSUB PAUSA
PRINT #5,
PRINT #5,
PRINT #5,
PRINT #5, SPC(18); "VEHICLE COSTS"
PRINT #5,
PRINT #5, SPC(33); " LIGHT VEHICLES"; SPC(3); "HEAVY VEHICLES"
PRINT #5,
PRINT #5, SPC(10); "LUBRICATING OIL"
PRINT #5, SPC(10); "Dollars/quart:      ";
PRINT #5, USING "#####.##"      "; cpOil, tpOil
PRINT #5, SPC(10); "TIRE WEAR"
PRINT #5, SPC(10); "Dollars/set of tire:   ";
PRINT #5, USING "#####.##"      "; csetTire, tsetTire
PRINT #5, SPC(10); "MAINTENANCE AND REPAIR"
PRINT #5, SPC(10); "Cost per 1000 miles:   ";
PRINT #5, USING "#####.##"      "; cRepvalue, tRepvalue
PRINT #5, SPC(10); "DEPRECIATION"
PRINT #5, SPC(10); "Value of a new vehicle:  ";
PRINT #5, USING "#####.##"      "; carvalue, truckvalue
PRINT #5, SPC(10); "FUEL VALUE"
PRINT #5, SPC(10); "Dollars/gallon:      ";
PRINT #5, USING "#####.##"      "; cpfuel, tpfuel
IF yew$ = "con" THEN GOSUB PAUSA
IF yew$ = "lpt1" THEN PRINT #5, CHR$(12)
RETURN

```

CALCULOS:

redim que(nqi + 1), COST(nqi + 1), revenue(nqi + 1),fcmin(nqi + 1),ftmin(nqi + 1)

\*\*\*\*\*  
 \*\* C A L C U L A T I O N S \*\*  
 \*\*\*\*\*

```

    qmax = 0
    qo = (srec * necb + nmb * srb) * dt
    IF qo > qmax THEN qmax = qo
  FOR iq = 1 TO nqi
    IF qi(iq) > qmax THEN qmax = qi(iq)
    tq(iq + 1) = iq * dt + t0
  NEXT iq
  k = 2
  t = t0 + dt
  que(1) = 0!
  quemax = 0
  fcmin(1) = 0
  ftmin(1) = 0
  delay = 0
  veh = 0
  fcmax = 0
  fmax = 0

  COST(1) = 0!
  revenue(1) = 0!
  costmax = 0!
  *****Computation of Costs per unit
  totalcost = 0
  *****Vehicles percentages
  pc = (1 - ptrucks / 100)
  pt = ptrucks / 100
  *****Moving delay
  demc = cspstop / (( cacc + cdec ) * 3600.0 )
  demt = tspstop / (( tacc + tdec ) * 3600.0 )

  ***** Computation of Cost Constant for Trucks
  ***** Stops
  CALL CURVA(speed(), tFuel(), tspstop, tstopFuel, 1!)
  CALL CURVA(speed(), tOil(), tspstop, tstopOil, 1!)
  CALL CURVA(speed(), tTire(), tspstop, tstopTire, 1!)
  CALL CURVA(speed(), tRep(), tspstop, tstopRep, 1!)
  CALL CURVA(speed(), tDep(), tspstop, tstopDep, 1!)
  tAcpu = (tpfatal * fatalcost + tpnofat * nofatcost + tppropd * propdcost) * accrate
  alfatFuel = tstopFuel * tpFuel / 1000 * pt
  alfatOil = tstopOil * tpOil / 1000 * pt
  alfatTire = tstopTire * tsetTire / 100000 * pt
  alfatRep = tstopRep * tRepvalue / 100000 * pt
  alfatDep = tstopDep * truckvalue / 100000 * pt
  alfatAcc = tAcpu / 1000000! * pt
  alfatTime = tvaluetime * demt * pt
  alfat = alfatFuel + alfatOil + alfatTire + alfatRep
  alfat = alfat + alfatDep + alfatAcc + alfatTime

  ***** Delay
  betatFuel = tidleFuel * tpFuel / 1000 * pt
  betatOil = tidleOil * tpOil / 1000 * pt
  
```

```

betatRep = tidleRepptm * tRepvalue / 100000 * pt
betatDep = tpctDep * truckvalue / 100000 * pt
betatTime = tvalueTime * pt
betat = betatFuel + betatOil + betatRep + betatDep + betatTime

```

```

////////// Computation of Cost Constant for Cars

```

```

////////// Stops

```

```

CALL CURVA(speed(), cFuel(), cspstop, cstopFuel, 1!)
CALL CURVA(speed(), cOil(), cspstop, cstopOil, 1!)
CALL CURVA(speed(), cTire(), cspstop, cstopTire, 1!)
CALL CURVA(speed(), cRep(), cspstop, cstopRep, 1!)
CALL CURVA(speed(), cDep(), cspstop, cstopDep, 1!)
cAcpu = (cpfatal * fatalcost + cpnofat * nofatcost + cppropd * propdcost) * accrate
alfacFuel = cstopFuel * cpFuel / 1000 * pc
alfacOil = cstopOil * cpOil / 1000 * pc
alfacTire = cstopTire * csetTire / 100000 * pc
alfacRep = cstopRep * cRepvalue / 100000 * pc
alfacDep = cstopDep * carvalue / 100000 * pc
alfacAcc = cAcpu / 1000000! * pc
alfacTime = demc * cvalueTime * pc
alfac = alfacFuel + alfacOil + alfacTire + alfacRep
alfac = alfac + alfacTime + alfacDep + alfacAcc

```

```

////////// Delay

```

```

betacFuel = cidleFuel * cpFuel / 1000 * pc
betacOil = cidleOil * cpOil / 1000 * pc
betacRep = cidleRepptm * cRepvalue / 100000 * pc
betacDep = cpctDep * carvalue / 100000 * pc
betacTime = cvalueTime * pc
betac = betacFuel + betacOil + betacRep + betacDep + betacTime

```

```

//////////

```

```

beta = betat + betac
betaFuel = betacFuel + betatFuel
betaOil = betacOil + betatOil
betaTime = betacTime + betatTime
betaDep = betacDep + betatDep
betaRep = betacRep + betatRep

```

```

alfa = alfac + alfat
alfaFuel = alfacFuel + alfatFuel
alfaOil = alfacOil + alfatOil
alfaTire = alfacTire + alfatTire
alfaDep = alfacDep + alfatDep
alfaRep = alfacRep + alfatRep
alfaAcc = alfacAcc + alfatAcc
alfaTime = alfatTime + alfacTime

```

```

FOR i = 1 TO nqi

```

```

veh = veh + qi(i) * (1 - phov / 100)
que(i + 1) = qi(i) * (1 - phov / 100) - qo + que(i)
IF que(i + 1) < 0! THEN que(i + 1) = 0!
IF que(i + 1) > quemax THEN quemax = que(i + 1)
delay = ((que(i) + que(i + 1)) / 2) * dt + delay
carcosttotal = (betac * delay + alfac * veh)
truckcosttotal = (betat * delay + alfat * veh)
IF pc <> 0 then fcmin(i + 1) = (carcosttotal + opcost*(t - t0)*pc)/(veh*pc)

```

```

If pt <> 0 THEN ftmin(i + 1) = (truckcosttotal + opcost*(t - t0)*pt)/(veh*pt)
IF fcmin(i + 1) > fcmx THEN fcmx = fcmin(i + 1)
IF ftmin(i + 1) > ftmax THEN ftmax = ftmin(i + 1)
avgd = delay / veh
COST(i + 1) = alfa * veh + beta * delay
revenue(i + 1) = (feecar * pc + feetruck * pt) * ( veh - que(i + 1))
IF COST(i + 1) > costmax THEN costmax = COST(i + 1)
IF revenue(i + 1) > costmax THEN costmax = revenue(i + 1)
t = t + dt
NEXT i
tmax = t0 + nqi * dt
fcmin(1) = fcmin(2)
ftmin(1) = ftmin(2)

*****Computation of Fuel consumption cost
CALL CURVA(speed(), tFuel(), tspstop, tstopFuel, 1!)
CALL CURVA(speed(), cFuel(), cspstop, cstopFuel, 1!)
costFueLt = tpFuel * (tstopFuel * veh + tidleFuel * delay) * pt / 1000
costFueLc = cpFuel * (cstopFuel * veh + cidleFuel * delay) * pc / 1000
costFuel = (costFueLt + costFueLc)

*****Accidents Cost
costAccit = tpfatal * fatalcost + tpnofat * nofatcost + tppropd * propdcost
costAccit = costAccit * accrate / 1000000! * veh * pt
costAccic = cpfatal * fatalcost + cpnofat * nofatcost + cppropd * propdcost
costAccic = costAccic * accrate / 1000000! * veh * pc
costAcci = costAccit + costAccic

*****Maintenance Cost
CALL CURVA(speed(), tRep(), tspstop, tstopRep, 1!)
CALL CURVA(speed(), cRep(), cspstop, cstopRep, 1!)
costRept = tRepvalue * (tstopRep * veh + tidleRepptm * delay) * pt / 100000
costRepc = cRepvalue * (cstopRep * veh + cidleRepptm * delay) * pc / 100000
costRep = (costRept + costRepc)

*****Oil Cost
CALL CURVA(speed(), tOil(), tspstop, tstopOil, 1!)
CALL CURVA(speed(), cOil(), cspstop, cstopOil, 1!)
costOilt = tpOil * (tstopOil * veh + tidleOil * delay) * pt / 1000
costOilc = cpOil * (cstopOil * veh + cidleOil * delay) * pc / 1000
costOil = (costOilt + costOilc)

*****Depreciation Cost
CALL CURVA(speed(), tDep(), tspstop, tstopDep, 1!)
CALL CURVA(speed(), cDep(), cspstop, cstopDep, 1!)
costDept = truckvalue * (tstopDep * veh + tpctDep * delay) * pt / 100000
costDepc = carvalue * (cstopDep * veh + cpctDep * delay) * pc / 100000
costDep = (costDept + costDepc)

*****Value of Time
costTimec = cvaluetime * ( delay + demc ) * pc
costTimet = tvaluetime * ( delay + demt ) * pt
costTime = (costTimec + costTimet)

*****Computation of Tire Wear cost
CALL CURVA(speed(), tTire(), tspstop, tstopTire, 1!)

```

```

CALL CURVA(speed(), cTire(), cspstop, cstopTire, 1!)
costTiret = tsetTire * (tstopTire * veh) * pt / 100000
costTirec = csetTire * (cstopTire * veh) * pc / 100000
costTire = (costTiret + costTirec)

*****Total Cost
totalcarcost = costFuelc + costOilc + costDepc + costRepc + costTimec
totalcarcost = totalcarcost + costTirec + costAccic

totaltruckcost = costFuelt + costOilt + costDept + costRept + costTimet
totaltruckcost = totaltruckcost + costTiret + costAccit

totalcost = costFuel + costOil + costDep + costRep + costTime
totalcost = totalcost + costTire + costAcci
*****Revenue
totrevenue = (veh - que(nqi + 1)) * (pc * feecar + pt * feetruck)
***** Operation Cost
oopcost = opcost * (tmax - t0)
*****Profit
profit = totrevenue - oopcost
***** Minimum Unit Toll Required
costo = totalcost + oopcost
IF (feecar + feetruck) > 0 THEN
IF pc <> 0 then feecarmin = (totalcarcost + oopcost*pc)/(veh*pc)
If pt <> 0 THEN feetruckmin = (totaltruckcost + oopcost*pt)/(veh*pt)
END IF
***** Toxic Emissions
CALL CURVA(speed(), cCO(), cspstop, cstopCO, 1!)
CALL CURVA(speed(), cHC(), cspstop, cstopHC, 1!)
CALL CURVA(speed(), cNOX(), cspstop, cstopNOX, 1!)
CALL CURVA(speed(), tCO(), tspstop, tstopCO, 1!)
CALL CURVA(speed(), tHC(), tspstop, tstopHC, 1!)
CALL CURVA(speed(), tNOX(), tspstop, tstopNOX, 1!)

cNOXemissions = cstopNOX * veh * 2.2 / 1000 + cidleNOX * delay
tNOXemissions = tstopNOX * veh * 2.2 / 1000 + tidleNOX * delay
NOXemissions = cNOXemissions + tNOXemissions

cCOemissions = cstopCO * veh * 2.2 / 1000 + cidleCO * delay
tCOemissions = tstopCO * veh * 2.2 / 1000 + tidleCO * delay
COemissions = cCOemissions + tCOemissions

cHCemissions = cstopHC * veh * 2.2 / 1000 + cidleHC * delay
tHCemissions = tstopHC * veh * 2.2 / 1000 + tidleHC * delay
HCemissions = cHCemissions + tHCemissions

RETURN

RESULTADOS:
CLOSE #5
OPEN yew$ FOR OUTPUT AS #5

*****
/* Printing of Queue and Delay */
*****

```

```

CLS
delay = 0
veh = 0
t = t0 + dt
FOR i = 1 TO nqi
IF INT((i - 1) / lp) = (i - 1) / lp THEN GOSUB CABEZA
PRINT #5, " ";
PRINT #5, USING " ##.## "; t - dt, t;
PRINT #5, USING " ##### "; qi(i) / dt;
veh = veh + qi(i) * (1 - phov / 100)
PRINT #5, USING " ##### "; veh;
PRINT #5, USING " ##### "; que(i + 1);
delay = ((que(i) + que(i + 1)) / 2) * dt + delay
PRINT #5, USING " #####.## "; delay
t = t + dt
NEXT i
PRINT #5,
PRINT #5,

PRINT #5, SPC(10); "The delay per vehicle is "; : PRINT #5, USING "#####.##"; avgd;
PRINT #5, " hrs"
IF yew$ = "con" THEN GOSUB PAUSA
IF yew$ = "lpt1" THEN PRINT #5, CHR$(12)
IF yew$ = "lpt1" THEN PRINT #5,
CLS
delay = 0
veh = 0
t = t0 + dt
fcmin = 0
ftmin = 0
FOR i = 1 TO nqi
IF INT((i - 1) / lp) = (i - 1) / lp THEN GOSUB HEAD
PRINT #5, " ";
PRINT #5, USING "####.## "; t - dt, t;
veh = veh + qi(i) * (1 - phov / 100)
delay = ((que(i) + que(i + 1)) / 2) * dt + delay
adcost = opcost*(t - t0)
costotal = cost(i + 1) + adcost

PRINT #5, USING " #####"; COST(i + 1);
PRINT #5, USING " #####"; adcost;
PRINT #5, USING " #####"; costotal;
PRINT #5, USING " #####"; revenue(i + 1);
PRINT #5, USING " ##.##"; fcmin(i + 1);
PRINT #5, USING " ##.##"; ftmin(i + 1)
t = t + dt
NEXT i
PRINT #5,
PRINT #5,
PRINT #5," Values are given in US dollars"

IF yew$ = "con" THEN GOSUB PAUSA
IF yew$ = "lpt1" THEN PRINT #5, CHR$(12)

```

```

IF yew$ = "lpt1" THEN PRINT #5,

PRINT #5, SPC(25); "*****"
PRINT #5, SPC(25); "** COST ANALYSIS SUMMARY **"
PRINT #5, SPC(25); "*****"
PRINT #5,
PRINT #5, SPC(26); " LIGHT VEHS. "; " HEAVY VEHS. "; " TOTAL"
PRINT #5,
PRINT #5, SPC(7); "Fuel Cost: ";
PRINT #5, USING "#####.##"; costFuelc, costFueLt, costFuel
PRINT #5,
PRINT #5, SPC(7); "Accident Cost: ";
PRINT #5, USING "#####.##"; costAccic, costAccit, costAcci
PRINT #5,
PRINT #5, SPC(7); "Oil Cost: ";
PRINT #5, USING "#####.##"; costOilc, costOilt, costOil
PRINT #5,
PRINT #5, SPC(7); "Maintenance: ";
PRINT #5, USING "#####.##"; costRepc, costRept, costRep
PRINT #5,
PRINT #5, SPC(7); "Depreciation: ";
PRINT #5, USING "#####.##"; costDepc, costDept, costDep
PRINT #5,
PRINT #5, SPC(7); "Value of Time: ";
PRINT #5, USING "#####.##"; costTimec, costTimet, costTime
PRINT #5,
PRINT #5, SPC(7); "Tire Wear: ";
PRINT #5, USING "#####.##"; costTirec, costTiret, costTire
PRINT #5,
PRINT #5,
PRINT #5, SPC(7); "Total User Cost:";
PRINT #5, USING "#####.##"; totalcarcost, totaltruckcost, totalcost
IF yew$ = "con" THEN GOSUB PAUSA
PRINT #5,
PRINT #5,
PRINT #5,
PRINT #5,
PRINT #5, SPC(20); "Gross Revenue: ";
PRINT #5, USING "#####.##"; totrevenue
PRINT #5,
PRINT #5, SPC(20); "Administration Cost: ";
PRINT #5, USING "#####.##"; oopcost
PRINT #5,
PRINT #5, SPC(20); "User Cost: ";
PRINT #5, USING "#####.##"; totalcost
PRINT #5,
PRINT #5, SPC(20); "Net Revenue: ";
PRINT #5, USING "#####.##"; profit - totalcost
PRINT #5,
PRINT #5, SPC(20); "Break-even toll per car :";
PRINT #5, USING "#####.##"; feecarmin
PRINT #5,
PRINT #5, SPC(20); "Break-even toll per truck:";

```

```

PRINT #5, USING "#####.##"; feetruckmin
PRINT #5,
PRINT #5,
PRINT #5, "      Cost Values are given in U.S. dollars"
PRINT #5,
PRINT #5,
  IF yew$ = "con" THEN GOSUB PAUSA
PRINT #5, SPC(30); " EMISSIONS (pounds)"
PRINT #5,
PRINT #5, SPC(20); "Carbon Monoxide (CO) : ";
PRINT #5, USING "#####.##"; COemissions
PRINT #5,
PRINT #5, SPC(20); "Hydrocarbons (HC) : ";
PRINT #5, USING "#####.##"; HCemissions
PRINT #5,
PRINT #5, SPC(20); "Nitrogen Oxides (NOX) : ";
PRINT #5, USING "#####.##"; NOXemissions
PRINT #5,
IF yew$ = "lpt1" THEN PRINT #5, CHR$(12)
GOSUB PAUSA
RETURN

```

'Subroutine Used to print the heading

```

CABEZA:
  IF (i - 1) <> 0 THEN
    IF yew$ = "con" THEN GOSUB PAUSA
    IF yew$ = "lpt1" THEN PRINT #5, CHR$(12)
  END IF
PRINT #5,
PRINT #5,
PRINT #5,
PRINT #5, SPC(25); "*****"
PRINT #5, SPC(25); "** MEASURES OF EFFECTIVENESS **"
PRINT #5, SPC(25); "*****"
PRINT #5,
  IF i = 1 THEN
PRINT #5,
PRINT #5, "  The capacity of the Toll Plaza is ";
PRINT #5, USING "##### "; qo/dt;
PRINT #5, " vph"
PRINT #5,
  END IF
PRINT #5,
PRINT #5, "      t      t      q      Vehicles      Queue      Delay"
PRINT #5, "      i      f      in      Arrived      (veh)      (veh*hr)"
PRINT #5, "      (hrs)   (hrs)   (vph)   (veh)      "
PRINT #5,
RETURN

```

HEAD:

```

  IF (i - 1) <> 0 THEN
    IF yew$ = "con" THEN GOSUB PAUSA
    IF yew$ = "lpt1" THEN PRINT #5, CHR$(12)
  END IF
PRINT #5,
PRINT #5,

```

```

PRINT #5,
PRINT #5, SPC(25); "*****"
PRINT #5, SPC(25); "** COST-REVENUE ANALYSIS **"
PRINT #5, SPC(25); "*****"
PRINT #5,
  IF i = 1 THEN
PRINT #5,
PRINT #5,
  END IF
PRINT #5,
PRINT #5, "    t    t    User  Administ.  Total  Gross  Break-even toll "
PRINT #5, "    i    f    Cost    Cost    Cost  Revenue  Light Heavy "
PRINT #5, "    (hrs) (hrs)                               Vehicles "
PRINT #5,
RETURN

```

GRAFICA:

```

*****
** G R A P H I C S      M O D U L E **
*****

CLS
LOCATE 10, 1
PRINT " Note: To make a printout of the graphics showed in the screen, both"
PRINT "    keys (Ctrl) and (PrtSc), should be pressed at the same time. "
PRINT "    Although the message (PRESS ENTER TO CONTINUE) will not appear "
PRINT "    while the graphics are shown, you should do so in order to go to"
PRINT "    the next screen."
GOSUB PAUSA
SCREEN 2
' Flow Graphic
dift = (tmax - t0) / 2
WINDOW (t0 - .96 * dift, 0 - .17 * qmax / dt)-(tmax + 1.01 * dift, 1.07 * qmax / dt)
' Plotting of Flowin Values
FOR i = 1 TO nqi
  IF i = 1 THEN LINE (t0, 0)-(t0, qi(i) / dt)
  LINE (tq(i), qi(i) / dt)-(tq(i + 1), qi(i) / dt)
  IF i <> nqi then
    LINE (tq(i + 1), qi(i) / dt)-(tq(i + 1), qi(i + 1) / dt)
  ELSE
    LINE (tq(i + 1), qi(i) / dt)-(tq(i + 1), 0)
  END IF
NEXT i
LINE (tmax + dift / 10, .9 * qmax / dt)-(tmax + dift / 3, .9 * qmax / dt)
LOCATE 4, 69
PRINT "Flow in"
' Plotting of Flowout Values
LINE (tq(1), qo / dt)-(tq(nqi + 1), qo / dt), , , &HFFCD
LINE (tq(nqi + 1), qo / dt)-(tq(nqi + 1), 0), , , &HFFCD
LINE (tmax + dift / 10, .7 * qmax / dt)-(tmax + dift / 3, .7 * qmax / dt), , , &HFFCD
LOCATE 8, 69
PRINT "Capacity"

' Plotting of axes
CALL AXCES(t0, tmax, dt, qmax / dt, dift, "flow (vph)")
PAUSA$ = "": WHILE PAUSA$ = "": PAUSA$ = INKEY$: WEND

```

```

' Queue Graphic
CLS
IF quemax = 0 THEN quemax = 1000
WINDOW (t0 - .96 * dift, -.17 * quemax)-(tmax + 1.01 * dift, 1.07 * quemax)
t = t0
FOR i = 1 TO nqi
  LINE (t, que(i))-(t + dt, que(i + 1))
  t = t + dt
NEXT i
CALL AXXES(t0, tmax, dt, quemax, dift, "queue (veh)")
PAUSA$ = "": WHILE PAUSA$ = "": PAUSA$ = INKEY$: WEND

' Break-even toll Graphic
CLS
IF fcmax = 0 THEN fcmax = 1.000
WINDOW (t0 - .96 * dift, -.17 * fcmax)-(tmax + 1.01 * dift, 1.07 * fcmax)
t = t0
FOR i = 1 TO nqi
  LINE (t, fcmin(i))-(t + dt, fcmin(i + 1))
  t = t + dt
NEXT i
CALL AXXES(t0, tmax, dt, fcmax, dift, "Break-even toll for light vehicles (dollars)")
PAUSA$ = "": WHILE PAUSA$ = "": PAUSA$ = INKEY$: WEND

' Break-even toll Graphic for trucks
CLS
IF ftmax = 0 THEN ftmax = 1.000
WINDOW (t0 - .96 * dift, -.17 * ftmax)-(tmax + 1.01 * dift, 1.07 * ftmax)
t = t0
FOR i = 1 TO nqi
  LINE (t, ftmin(i))-(t + dt, ftmin(i + 1))
  t = t + dt
NEXT i
CALL AXXES(t0, tmax, dt, ftmax, dift, "Break-even toll for heavy vehicles (dollars)")
PAUSA$ = "": WHILE PAUSA$ = "": PAUSA$ = INKEY$: WEND

' Cost Graphic
CLS
mil = 1000
IF costmax = 0 THEN costmax = 1000
wc1 = t0 - .96 * dift
wc2 = -.17 * costmax
wc3 = tmax + 1.01 * dift
wc4 = 1.07 * costmax
sep = costmax / 100
WINDOW (wc1, wc2)-(wc3, wc4)
t = t0
FOR i = 1 TO nqi
  LINE (t, COST(i))-(t + dt, COST(i + 1))
  LINE (t, revenue(i))-(t + dt, revenue(i + 1)), 2
  LINE (t, revenue(i) + sep)-(t + dt, revenue(i + 1) + sep), 2
  t = t + dt
NEXT i
LINE (tmax + dift / 10, .7 * costmax)-(tmax + dift / 3, .7 * costmax)
LOCATE 8, 69

```

```

PRINT "User Cost"
LINE (tmax + dift / 10, .9 * costmax)-(tmax + dift / 3, .9 * costmax), 2
LINE (tmax + dift / 10, .9 * costmax + sep)-(tmax + dift / 3, .9 * costmax + sep), 2
LOCATE 4, 69
PRINT "Gross"
Locate 5,69
Print "Revenue"
  CALL AXYES(t0, tmax, dt, costmax, dift, "cost (dollars)")
  PAUSA$ = "": WHILE PAUSA$ = "": PAUSA$ = INKEY$: WEND
  SCREEN 0
  CLS
RETURN

REM $STATIC
SUB AXYES (a0, amax, da, kmax, difa, C$) STATIC
SHARED nqi
'Draw X axes
LINE (a0, 0)-(amax + .1 * difa, 0)
'Draw Y axes
LINE (a0, 0)-(a0, 1.05 * kmax)

'Draw arrow for X
a00 = amax + .1 * difa
LINE (a00, 0)-(a00 - difa / 40, kmax / 40)
LINE (a00, 0)-(a00 - difa / 40, -kmax / 40)
'Draw arrow for Y
k00 = 1.05 * kmax
LINE (a0, k00)-(a0 - difa / 40, k00 - kmax / 40)
LINE (a0, k00)-(a0 + difa / 40, k00 - kmax / 40)
'Draw marks for Y axes
dh = kmax / 10
FOR i = 1 TO 11
LOCATE 22 - 2 * (i - 1), 9
  IF kmax < 100 THEN
PRINT USING "#####.##"; kmax / 10 * (i - 1)
  ELSE
PRINT USING "#####.###"; kmax / 10 * (i - 1)
  END IF
LINE (a0, (i - 1) * dh)-(a0 - difa / 40, (i - 1) * dh)
NEXT i
'Draw marks for X axes
FOR i = 1 TO 11
LOCATE 23, 18 + (i - 1) * 4
IF INT((i - 1) / 2) = (i - 1) / 2 THEN
  PRINT USING "###.###"; a0 + (i - 1) * difa / 5
END IF
LINE (a0 + (i - 1) * difa / 5, 0)-(a0 + (i - 1) * difa / 5, -kmax / 40)
NEXT i
LOCATE 1, 22
PRINT C$
LOCATE 22, 65
ya = 50
xa = 50
PRINT "t (hours)"
END SUB

```

```

SUB CURVA (xx(1), yy(1), x, y, adj) STATIC
IF x <= xx(1) THEN y = yy(1)
IF x >= xx(UBOUND(xx)) THEN y = yy(UBOUND(xx))
IF x > xx(1) AND xx < xx(UBOUND(xx)) THEN
  FOR i = 1 TO UBOUND(xx)
    IF x <= xx(i) THEN
      y = (yy(i) - yy(i - 1)) / (xx(i) - xx(i - 1)) * (x - xx(i - 1)) + yy(i - 1)
      y = y * adj
    EXIT FOR
  END IF
  NEXT i
END IF
END SUB ' ' SAVE FILE PROGRAM ' '
'$INCLUDE: 'COMUNES.BAS'
OPEN "DATA.XOL" FOR INPUT AS #1
  FOR iq = 1 TO nqi
    INPUT #1, qi(iq), tq(iq + 1)
  NEXT iq
CLOSE #1
CLS
ON ERROR GOTO ERRORES
Shell$ = " dir/w " + DRIVE$ + ":\" + DIR$ + ".BTH"
SHELL shell$
LOCATE 18, 10
  IF file1$ <> "" THEN
    PRINT " THE CURRENT FILE LOADED IS : "; (file1$)
  ELSE
    PRINT " NO FILE IS CURRENTLY LOADED"
  END IF
LOCATE 22, 10
PRINT " PRESS ENTER TO RETURN TO MENU"
SAVE1:
LOCATE 20, 10
INPUT " NAME OF THE FILE TO SAVE (EXTENSION .BTH ASSUMED )", file$
  IF file$ <> "" THEN
    file1$ = file$ + ".BTH"
    file2$ = DRIVE$ + ":\" + DIR$ + file$ + ".BTH"
    OPEN file2$ FOR OUTPUT AS #8
      WRITE #8, nqi, dt, t0
      WRITE #8, nqi
      FOR iq = 1 TO nqi
        WRITE #8, qi(iq), tq(iq + 1)
      NEXT iq
      WRITE #8, srec, necb, nmb, srb
      WRITE #8, ptrucks, phov
      WRITE #8, cpFuel, tpFuel
      WRITE #8, tspstop, cspstop
      WRITE #8, accrate, fatalcost, nofatcost, propdcost
      WRITE #8, cpOil, tpOil
      WRITE #8, csetTire, tsetTire
      WRITE #8, cRepvalue, tRepvalue
      WRITE #8, carvalue, truckvalue
      WRITE #8, cvaluetime, tvaluetime
      WRITE #8, opcost, fecar, feetruck
      WRITE #8, tacc, cacc, tdec, cdec
    CLOSE #8

```

```
END IF
CHAIN "OPTIONS"
END
```

```
ERRORES:
```

```
  CONST FILENOTFOUND = 53
  CONST BADFILENAME = 64
  SELECT CASE ERR
CASE FILENOTFOUND
  LOCATE 21, 50, 1
  BEEP
  PRINT " FILE NOT FOUND, TRY AGAIN "
  LOCATE 20, 10, 1
  PRINT "
  RESUME SAVE1
CASE BADFILENAME, 52
  LOCATE 21, 50, 1
  BEEP
  PRINT " BAD FILE NAME, TRY AGAIN "
  LOCATE 20, 10, 1
  PRINT "
  RESUME SAVE1
CASE ELSE
  ON ERROR GOTO 0
END SELECT
```

**The vita has been removed from  
the scanned document**