Reference

REPORT NO. DOT-TSC-FHWA-72-1

MANUAL FOR HIGHWAY NOISE PREDICTION (APPENDIX B)

J. E. WESLER
TRANSPORTATION SYSTEMS CENTER
55 BROADWAY
CAMBRIDGE, MA. 02142

DEPARTMENT OF TRANSPORTATION • UNITED STATES OF AMERICA

MARCH 1972
TECHNICAL REPORT

Prepared for:

DEPARTMENT OF TRANSPORTATION
FEDERAL HIGHWAY ADMINISTRATION
WASHINGTON, D.C. 20590

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
DOT-TSC-FHWA-72-1		

4. Title and Subtitle	5. Report Date
MANUAL FOR HIGHWAY NOISE PREDICTION APPENDIX B	March 1972
	6. Performing Organization Code
7. Author(s) J.E. Wesler	8. Performing Organization Report No.
9. Performing Organization Name and Address Department of Transportation Transportation Systems Center 55 Broadway Cambridge, MA 02142	10. Work Unit No. R-2519 11. Contract or Grant No. OS-207
12. Sponsoring Agency Name and Address Department of Transportation Federal Highway Administration Washington, D.C. 20590	13. Type of Report and Period Covered Technical Report 14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

The basic manual, published as the first volume of this report, is intended for use as a tool in predicting noise levels which will be generated by freely-flowing vehicle traffic along a highway of known characteristics. The first volume explains the basis for the computerized prediction model, used for highway noise level prediction, and contains the user's manual for the computer program. This volume contains the programmer's manual for the computer program, and the program listing in FORTRAN IV.

17. Key Words	18. Distribution Statement
·Traffic noise prediction ·Highway noise simulation, ·Traffic noise levels	Availability is Unlimited. Document may be Released To the National Technical Information Service, Springfield, Virginia 22151, for Sale to the Public.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	80	

TABLE OF CONTENTS

TABLE OF CONTENTS (CONT.)

LIST OF ILLUSTRATIONS

PREFACE

Appendix B to the basic report, Manual for Highway Noise Prediction, contains the programmer's manual and computer listing for the highway noise prediction model. This manual and listing are bound separately from the basic report, for convenience, and to permit a smaller overall size for the basic user's manual which forms the larger portion of the basic report.

APPENDIX B DETAILS AND LISTING OF COMPUTER PROGRAM

INTRODUCTION

The computer program, known as the Traffic Noise Prediction Model MOD 2, is designed to run on the IBM 7094 computer, which is currently installed at TSC. The program is written in FORTRAN IV language, and is designed to run in the batch mode. Because of the modular organization of the program, only a modest level of effort should be required to modify it for an interactive mode of operation. This modification is currently underway at TSC. As formatted in this manual, however, inputs are provided through punched cards, and outputs are provided through a line printer. This mode of operation is felt to provide maximum flexibility for other groups who may adapt this program for their own use.

The organizational scheme of the program is shown in the following flow diagram. The main blocks of code are contained in the main program, and in the subroutines INPUT and GEOMRY. The subroutine INPUT reads input data provided on cards, stores this information in core memory, and outputs it on the user's line printer. The subroutine GEOMRY provides the bulk of the calculations performed by the computer program, using a number of additional subroutines and functions in the process. Additional subroutines called by the main program are FILES, which assigns device numbers to the various I/O devices used by the program, and INTER, which computes reference intensities for the various vehicle groups specified by the user.

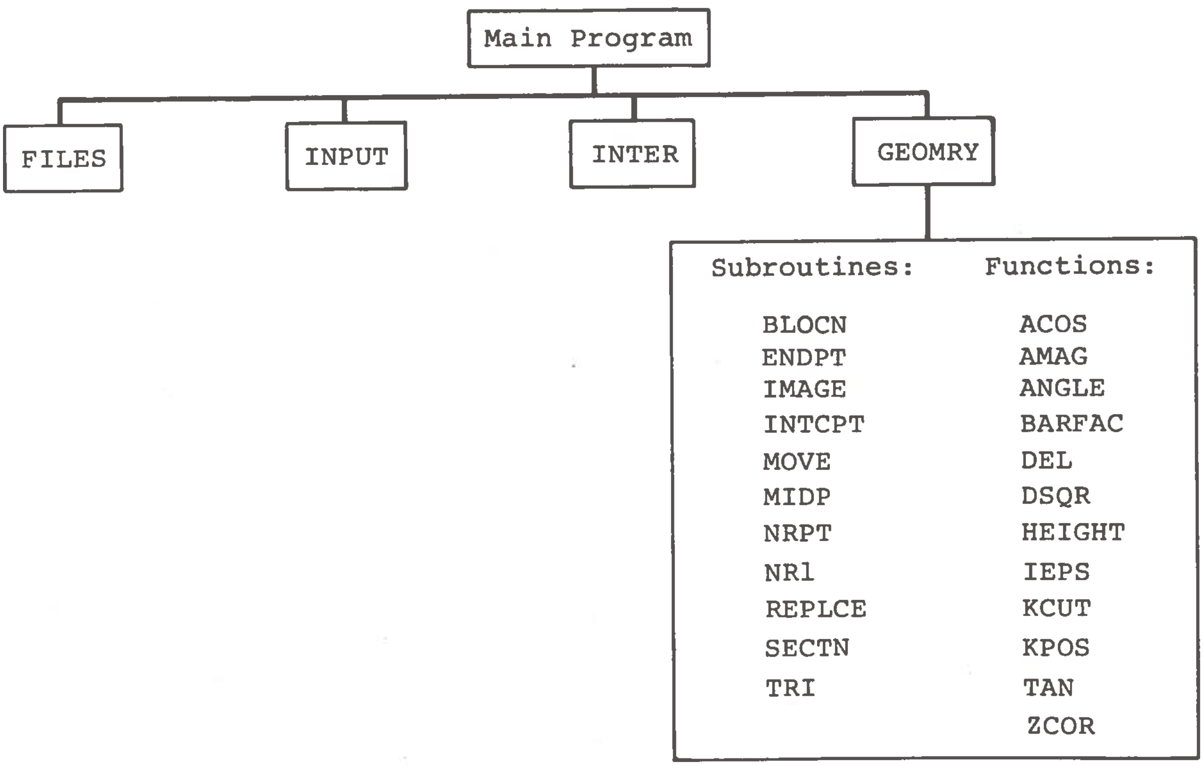
The main program and the subroutines INPUT and GEOMRY are discussed individually in some detail in the remainder of this section, and the remaining subroutines and functions are described very briefly. Listings for the entire computer program and details on program usage are also provided.

The following definitions are used throughout the remainder of this manual. A "road" or "roadway" is defined as a set of connected straight "road sections" for which the traffic conditions are uniform. That is, the number of vehicle groups and the concentration and mean vehicle speed associated with each group are the same for all sections of a given road. Barriers are treated similarly. Thus, a "barrier" is defined as a set of connected straight "barrier sections" for which the absorptive/reflective properties are the same.

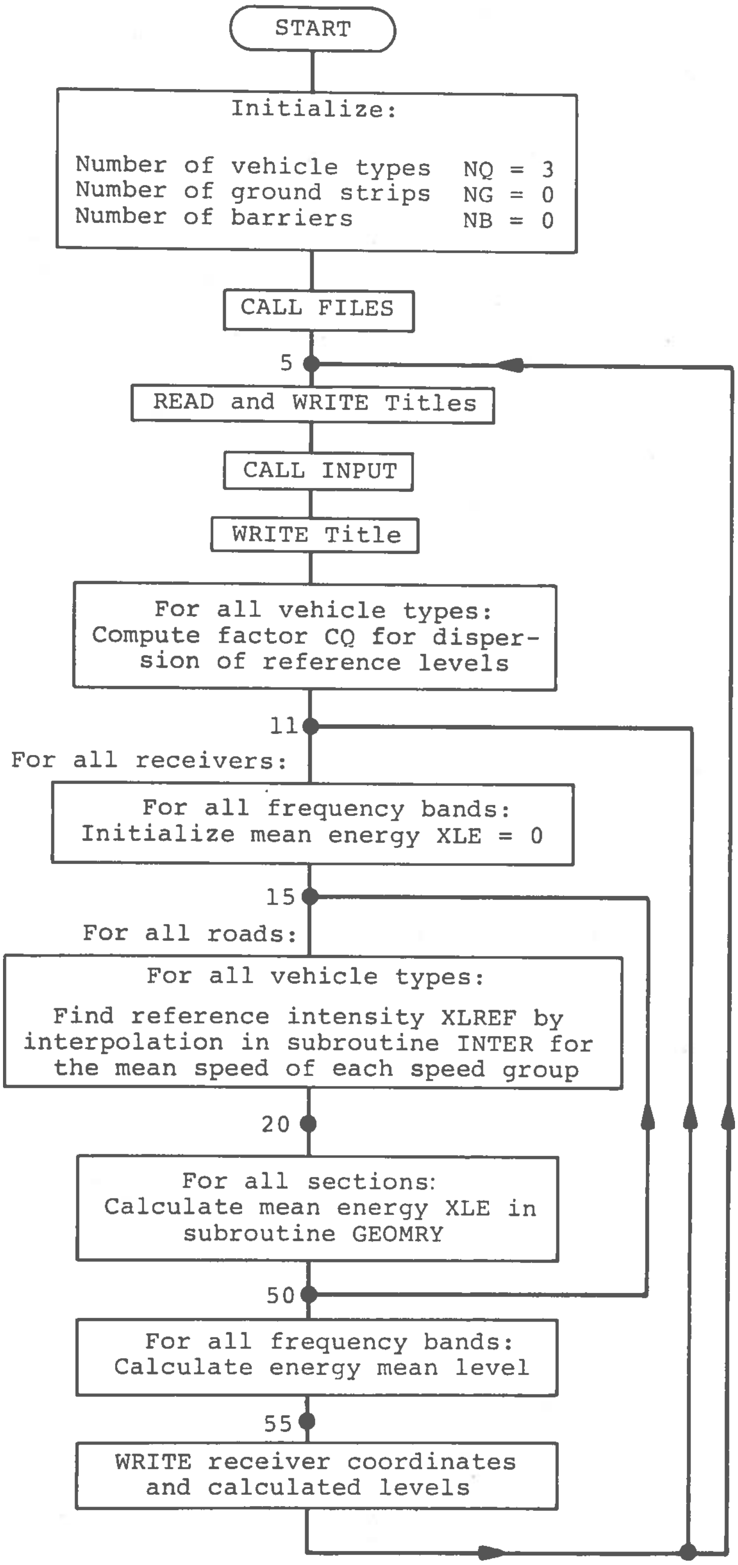
MAIN PROGRAM

The primary function of the main program is to control the flow of operations. Various subroutines are called upon to perform the bulk of the input-output operations and the analysis. In addition, certain variables are initialized in the main program and the final computation of energy mean levels is made.

ORGANIZATIONAL SCHEME OF COMPUTER PROGRAM



MAIN PROGRAM



A flow diagram of the main program is given below. This diagram is self-explanatory. Whenever possible, statement numbers are shown so that reference may be made to the corresponding block of code in the program listing.

LIST OF VARIABLES FOR MAIN PROGRAM

Variables appearing in the main program are listed below. Array variables are not generally identified as such; the dimensionality of the variables can be determined from the program listing. The variables XR1 and XR2 listed below are vector quantities which indicate the coordinates of road end points. The first element of the vector indicates the x-coordinate, the second element the y-coordinate, and the third element the z-coordinate. This format is used widely throughout the remainder of the entire computer program to denote coordinates.

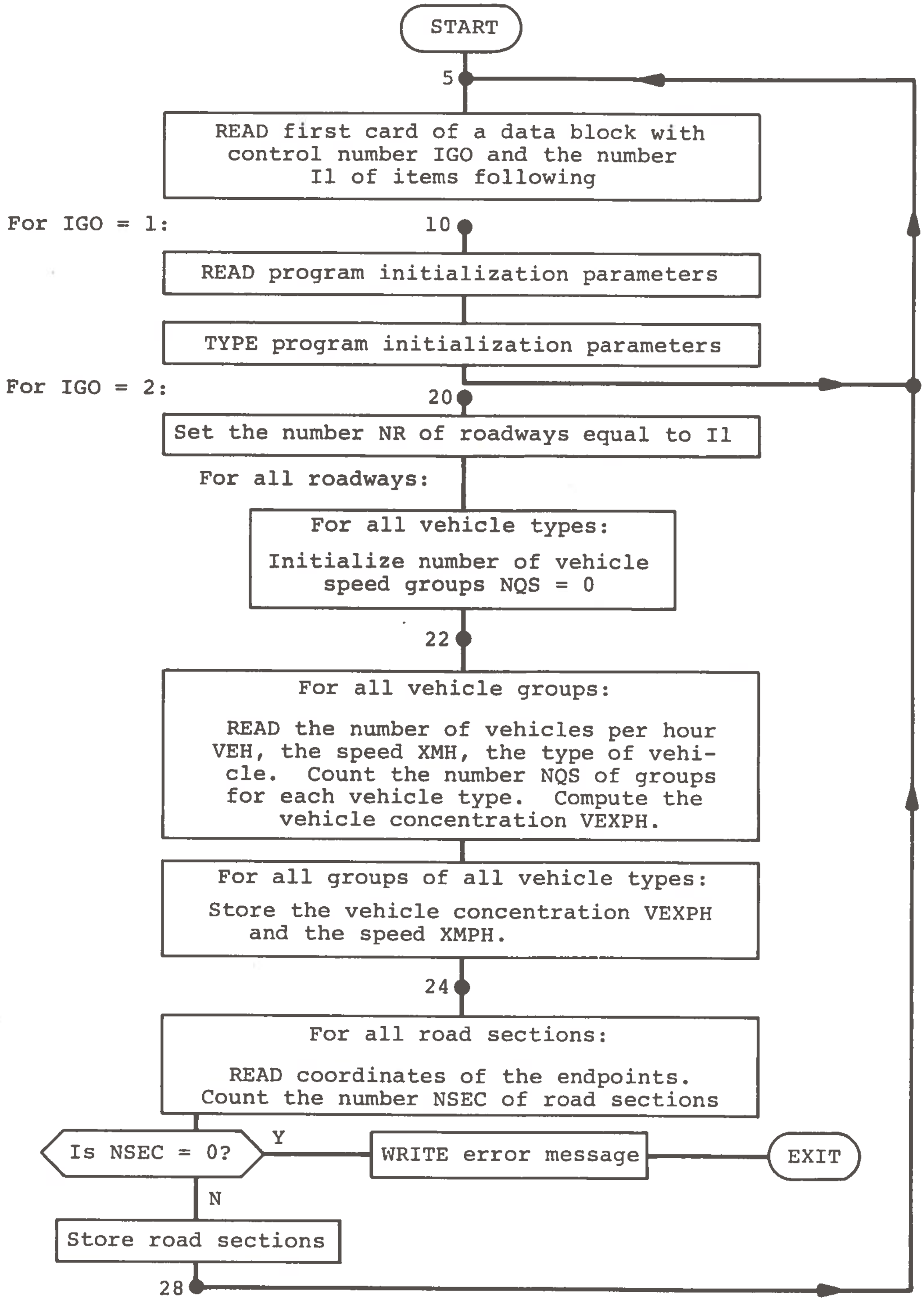
CQ	Factor accounting for standard deviation of reference level
I	Index
IBAR	Barrier number
IERR	Error index
IGRA	Ground strip number
II	Index
IQ	Index
ISEG	Barrier section number
J	Index
M	Roadway number
N	Road section number
NB	Number of barriers
NF	Number of frequency bands
NG	Number of absorptive ground strips
NLIM	Number of sections defining a roadway
NQ	Number of vehicle types
NQC	Number of groups within one vehicle type
NQS	Vector notation for number of vehicle groups

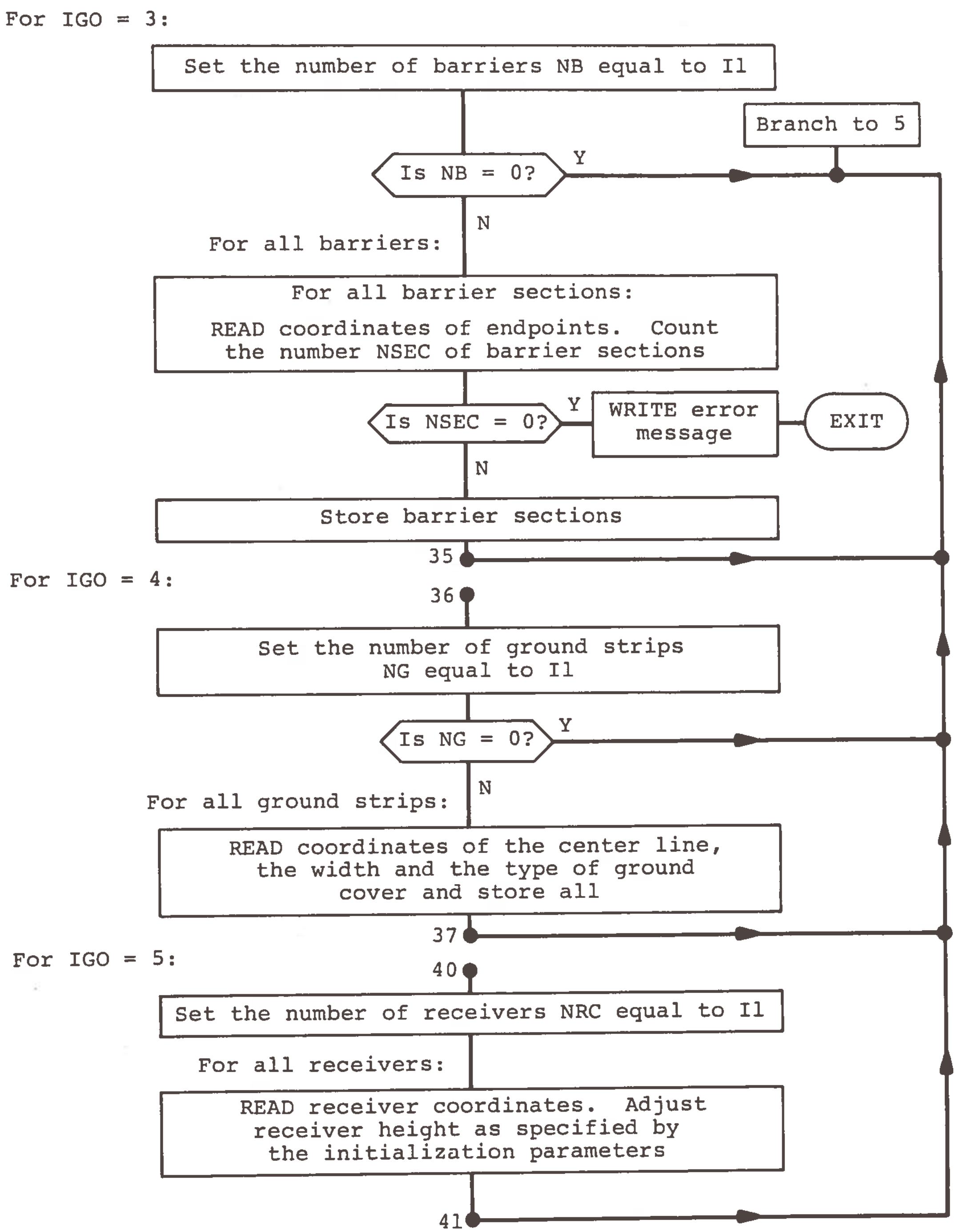
NR	Number of roadways
NRC	Number of receivers
NRSEC	Number of sections for one roadway
NRSM1	Number of sections for one roadway
RDIN	Vector notation for initialization parameters
RX	x-coordinate of roadway point
RY	y-coordinate of roadway point
RZ	z-coordinate of roadway point
SIG	Standard deviation of reference level
SIGL	Standard deviation of A-weighted sound intensity
VEXPH	Vehicles per foot
XKA	Cumulant of the A-weighted sound intensity
XLA	A-weighted overall intensity and level
XKAP	Cumulant of the intensity normalized with the energy mean
XLE	A-weighted intensity and level in frequency bands
XL10	A-weighted SPL exceeded 10% of the time
XL50	A-weighted SPL exceeded 50% of the time
XL90	A-weighted SPL exceeded 90% of the time
XLNP	Noise Pollution Level
XMPH	Vehicle speed in mph
XR1	Road section initial point
XR2	Road section end point
XRC	x-coordinate of receiver
YRC	y-coordinate of receiver
ZRC	z-coordinate of receiver

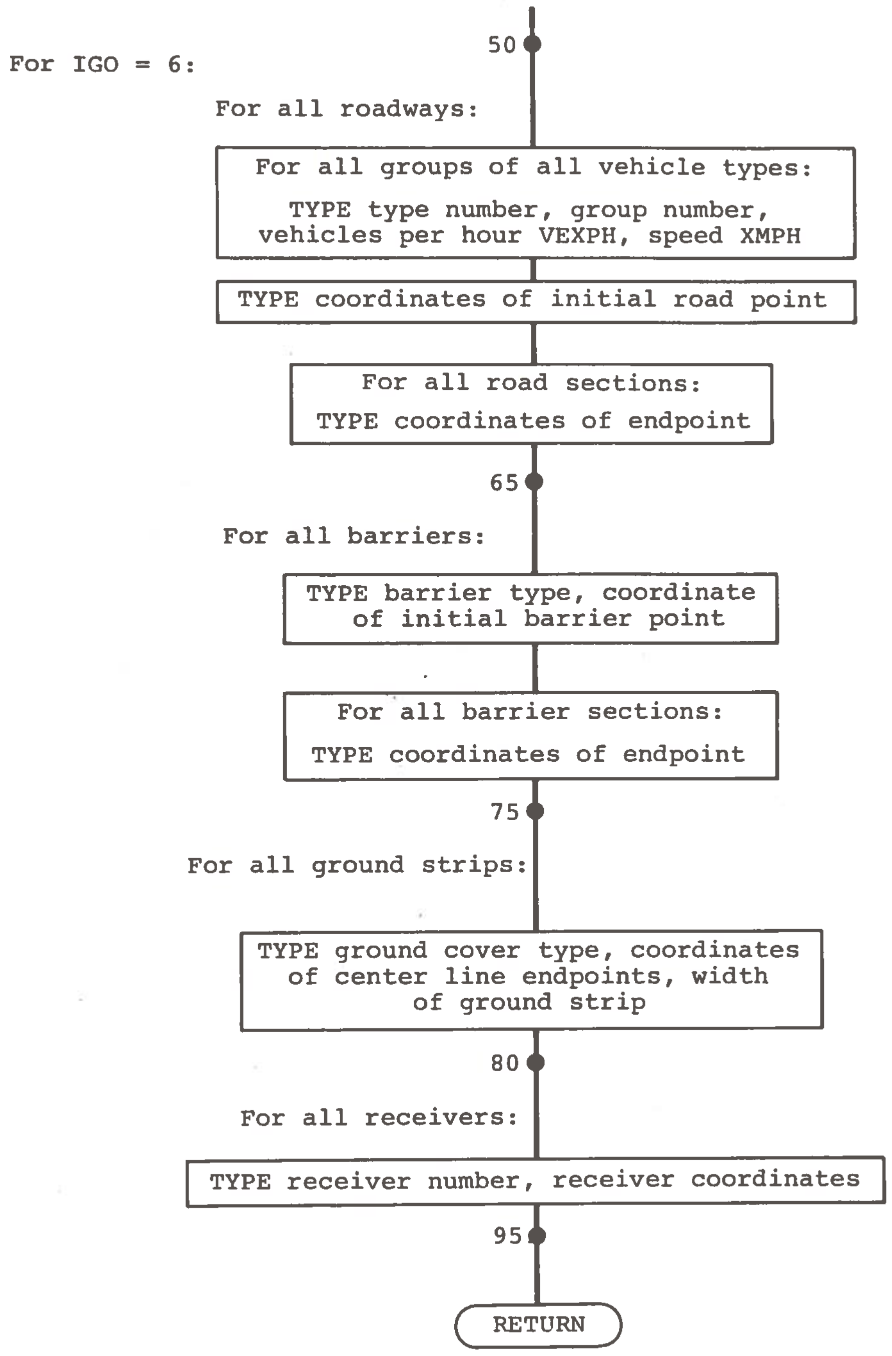
SUBROUTINE INPUT

This subroutine reads the input data provided by the user on punched cards, stores these data in core memory in a format suitable for use in the remainder of the computer program, and prints the

Subroutine Input




For IGO = 3:
Set the number of barriers NB equal to I1
Branch to 5
Is NB = 0?
Y
N
For all barriers:
For all barrier sections:
READ coordinates of endpoints. Count the number NSEC of barrier sections
Is NSEC = 0?
Y
WRITE error message
EXIT
N
Store barrier sections
35
For IGO = 4:
36
Set the number of ground strips NG equal to I1
Is NG = 0?
Y
N
For all ground strips:
READ coordinates of the center line, the width and the type of ground cover and store all
37
For IGO = 5:
40
Set the number of receivers NRC equal to I1
For all receivers:
READ receiver coordinates. Adjust receiver height as specified by the initialization parameters
41


For IGO = 6:
50
For all roadways:
For all groups of all vehicle types:
TYPE type number, group number, vehicles per hour VEXPH, speed XMPH
TYPE coordinates of initial road point
For all road sections:
TYPE coordinates of endpoint
65
For all barriers:
TYPE barrier type, coordinate of initial barrier point
For all barrier sections:
TYPE coordinates of endpoint
75
For all ground strips:
TYPE ground cover type, coordinates of center line endpoints, width of ground strip
80
For all receivers:
TYPE receiver number, receiver coordinates
95
RETURN

input and output data so that a permanent record is created.

The computer program is organized in such a way that the user need not redefine the entire situation each time a new card is analyzed. Instead, it is necessary to specify only those aspects of the problem that are different from the previous case. Accordingly, each block of input data is preceded by a control card that indicates the type of data (e.g., barrier specifications) and the number of items (e.g., number of barriers) to follow. After a special control character has signified the end of data input, the entire set of input variables is typed out and control returns to the main program.

A flow diagram of this subroutine is given below, followed by a list of variables.

LIST OF VARIABLES FOR SUBROUTINE INPUT

ALPHA	Alphanumeric information
BGS	Width of absorptive ground strip
BX	x-coordinate of barrier point
BY	y-coordinate of barrier point
BZ	z-coordinate of barrier point
I	Index
IBLAST	Barrier type
II	Number of items in data block
I2	Dummy variable
IA	Alphanumeric "A"
IDN	Index for program initialization parameter
IDUM	Index for type of absorptive ground cover
IG	Alphanumeric "G"
IGO	Index for data blocks
ILAST	Alphanumeric indicator for last section
IR	Alphanumeric "R"
IT	Alphanumeric "T"
ITY	Vehicle type

J	Index
K	Index
LAST	Alphanumeric "L"
NB	Number of barriers
NBSM1	Number of sections for one barrier
NG	Number of absorptive ground strips
NQ	Number of vehicle types
NQC	Number of groups within one vehicle type
NQS	Vector notation for number of vehicle groups per vehicle type
NR	Number of roadways
NRC	Number of receivers
NRSM1	Number of sections for one roadway
NSEC	Section number
NSM1	NSEC-1
NTEMP	Temporary variable
RDIN	Vector notation for initialization parameters
RX	x-coordinate of roadway point
RY	y-coordinate of roadway point
RZ	z-coordinate of roadway point
VALUE	Initialization parameter
VEH	Vehicles per hour
VEXPH	Vehicles per foot, vehicles per hour
VTEMP	Vehicle speed
X	Coordinate
XG1	X-coordinate for end point of ground strip center line
XMH	Speed in mph for one group of vehicles
XMPH	Vector notation for vehicle speed

XNIGHT	Height adjustment for receiver coordinate
XRC	X-coordinate of receiver
XT	X-coordinate for end point of road section
XXG1	X-coordinate of point on ground strip center line
Y	Coordinate
YG1	Y-coordinate for end point of ground strip center line
YRC	Y-coordinate of receiver
YT	Y-coordinate for end point of road section
YYG1	Y-coordinate of point on ground strip center line
Z	Coordinate
ZG1	Z-coordinate for end point of ground strip center line
ZRC	Z-coordinate of receiver
ZT	Z-coordinate for end point of road section
ZZG1	Z-coordinate of point on ground strip center line

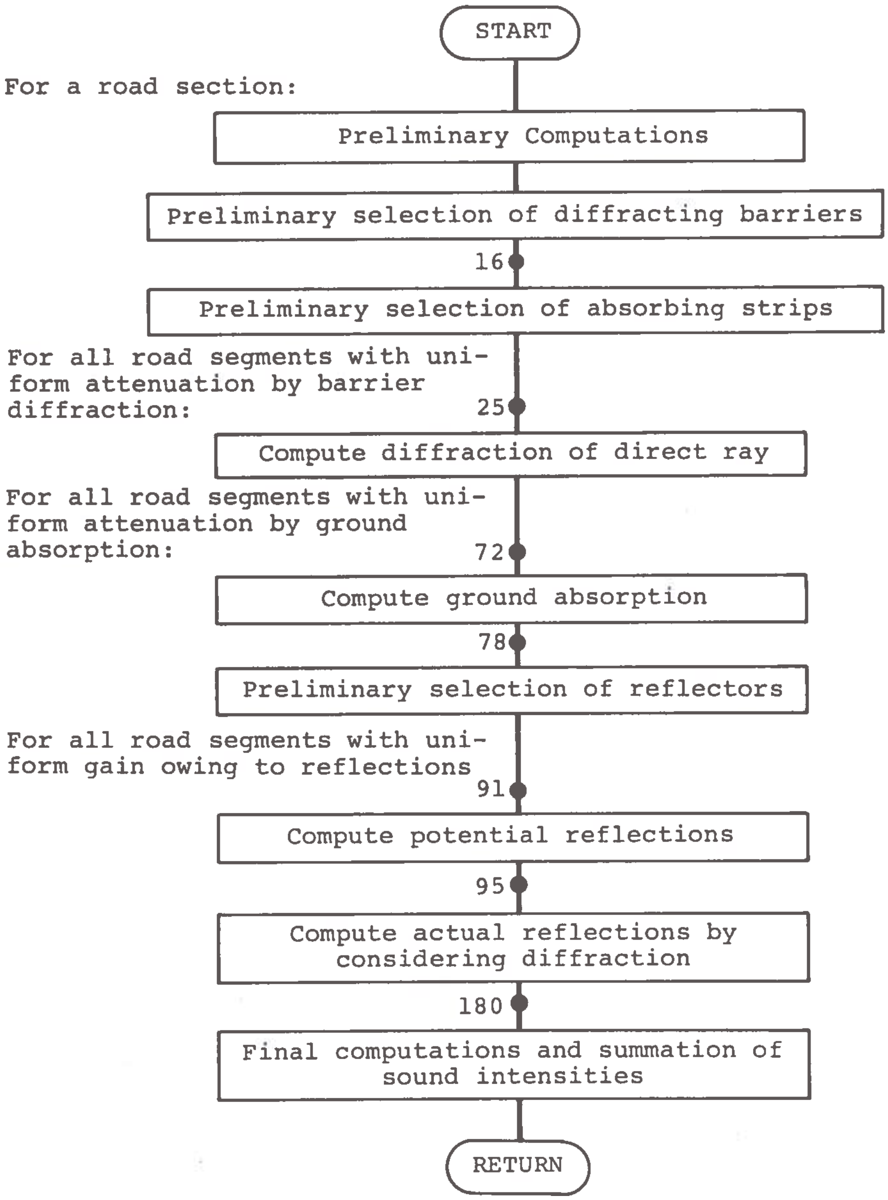
SUBROUTINE GEOMRY

The bulk of the computational effort of this subroutine is devoted to finding segments of a straight road section for which sound from all points along the segment reaching a particular receiver is nearly equally attenuated by barrier diffraction or ground absorption or is equally amplified by reflections.

Because of the complexity of this subroutine, a detailed description with a series of flow diagrams is presented. Before reviewing these diagrams, however, the reader should become familiarized with the basic philosophy of the computational scheme outlined below.

As shown in the following diagram giving an overview of subroutine GEOMRY, checks are performed on two different levels: first on all the barriers, then on all the absorptive ground strips, and finally on all the reflectors. On the lower level, preliminary checks are made mostly in the two dimensions of the x-y plane to determine whether a barrier, a ground strip, or a reflector can be at all effective for the road segment and the receiver under consideration. During these checks a smaller number of barriers, ground strips, and reflectors are stored and are investigated more closely at the higher level.

AN OVERVIEW OF SUBROUTINE GEOMRY



At this higher level, break points between road segments possibly affected and unaffected by barriers, ground strips, or reflectors are determined from checks first made in the two dimensions of the x-y plane. If the road segment between the initial point and the break point is unaffected, then no further checks in three dimensions are made on the actual effect of the remaining road segment, but the break point is considered as the new end point of the road segment. The remaining road segment is investigated after the calculations for the first road segment are finished.

If the road segment between the initial point and the break point is possibly affected by a barrier, a ground strip, or a reflector, the break point is taken as a preliminary end point until the subsequent analysis in three dimension shows whether there is an actual effect that makes it necessary to keep the break point as an end point or whether there is no effect and the break point can be dropped.

Since all the checks are performed first for diffraction, then for absorptive ground strips, and finally for reflections, the initial road section may be reduced several times by each group of elements. The procedure of subdividing a road section is diagrammed in Figure B-1. The final reduction in each group is stored in terms of end points XR2D for barriers, XR2G for absorptive ground strips, and XR2 for reflectors. By this means, duplicate calculations are avoided. After the noise from the road segment XR1Ø, XR2 is analyzed, the computation continues for possible reflections from the road segment XR2, XR2G is completely analyzed, the computation continues with consideration of ground absorption and reflections for the road segment XR2G, XR2D. The designation of end points during the course of this analysis is illustrated in Figure B-2. Finally, the computation continues with the entire problem of barrier diffraction, ground absorption, and reflection for the remaining road segment XR2D, XR2Ø until the noise from the entire road section XR1Ø, XR2Ø has been analyzed. At that point, program control is returned to the main program.

DETAILED FLOW DIAGRAMS

The flow diagrams presented at the end of this section show the operations and important branch points of the computations that are indicated in the overview diagram presented above. Again, statement numbers are shown so that reference may be made to the corresponding block of code in the program listing. The following comments are provided to supplement the information contained in the flow diagrams.

PRELIMINARY COMPUTATIONS

The main computation shown in the first flow chart concerns the distance between the near point XNPT on the source line and

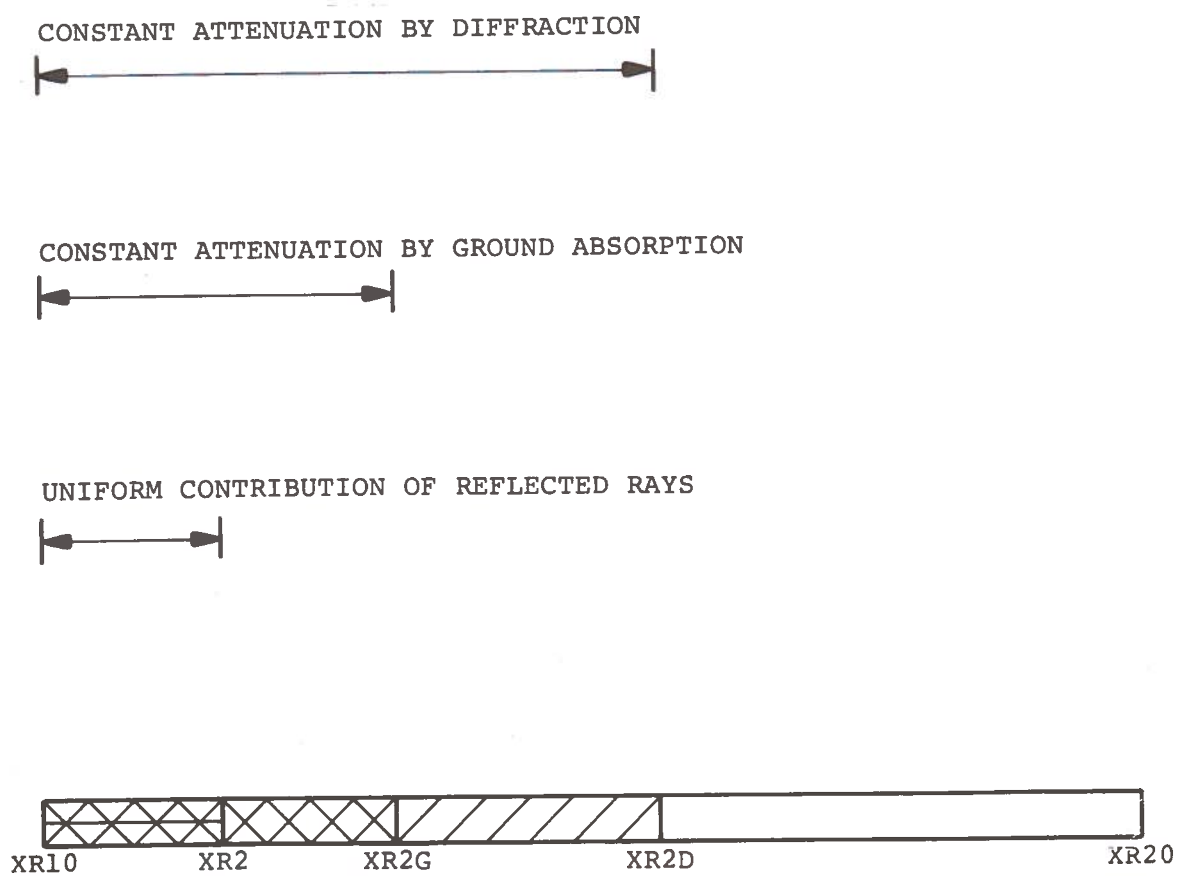


Figure B-1. Subdivision of Road Section

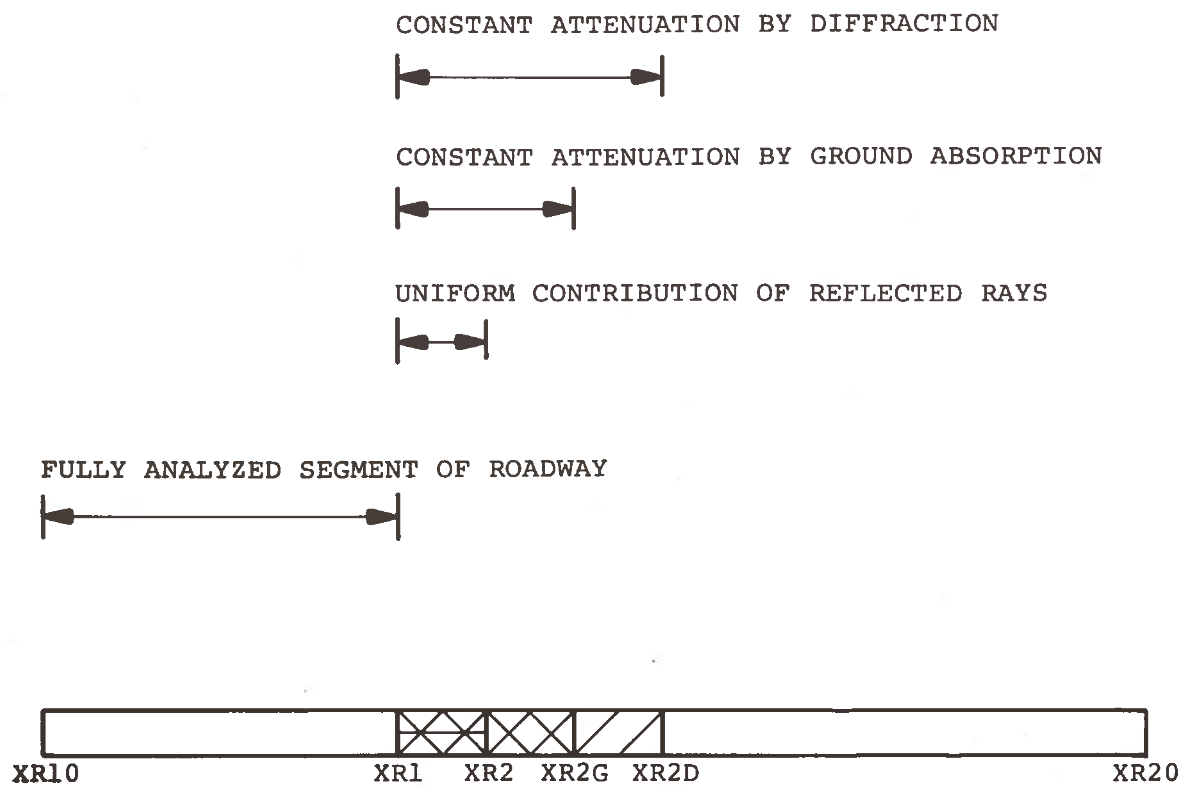


Figure B-2. Further Subdivision of Road Section the receiver XRC. This distance enters into the formula for the mean energy level and is a quantity that is characteristic of the entire road section.

PRELIMINARY SELECTION OF DIFFRACTING BARRIERS

The computations are made in the two dimensions of the x-y plane, and no checks are made for the height of barriers. A barrier whose top line crosses the roadway in a projection on the x-y plane is considered illegal and causes the program to abort. Thus, even a barrier crossing underneath an elevated roadway is illegal. A barrier stored is not necessarily high enough actually to diffract sound rays travelling from the road section to the receiver. The storage of the relative location(see SUBROUTINE BLOCN) saves considerable computation for very high barriers that shield the entire road section against the receiver.

PRELIMINARY SELECTION OF ABSORBING GROUND STRIPS

The computations are entirely analogous to the preliminary selection of diffracting barriers. No check is made for the distance of the ground strip center line from the road section. Therefore, a road section, or a part of it, which is inadvertently specified on a ground strip will not be detected in all cases by the program.

DIFFRACTION OF DIRECT RAY

The initialization of a path length DELPØ = -0.2 means that the attenuation by diffraction is set equal to zero in the frequency band centered at 500 Hz.

The test for whether or not a barrier possibly shields the entire road is made in the x-y plane and does not account for the actual height of the barrier.

The height difference for sound rays from cars above the barrier provides a rather conservative check for possible effects of diffraction. A positive height difference means the source is visible from the receiver location.

Only the most effective of a number of parallel barriers is considered; thus, the barrier having the strongest diffraction is retained. Note that this barrier is not necessarily the highest one. It is just as likely to be the barrier nearest to the road or to the receiver.

The calculation of a road segment with sufficiently similar diffraction is performed for cars only. It is then assumed that the same result holds for trucks having a different source height.

The check for maximum diffraction of sound in the frequency band centered at 500 Hz is made for trucks only, because the higher source height of trucks will result in smaller diffraction than for cars. If this test reveals a path length difference of more than 12.5 ft, further checks on other barriers can be skipped for the road segment considered. If a very high barrier shields the entire remaining road section, no further tests for diffractions of the direct ray are made. This procedure may result in some inaccuracies for very long road sections for which the diffraction of sound from points near to the receiver is very strong but for which the diffraction of sound from remote points might be considerably weaker. However, these inaccuracies are no more serious than those resulting from the assumption of the arbitrary value of 12.5 ft for the maximum path length difference, and are thus neglected.

GROUND ABSORPTION

Absorptive ground strips are disregarded for sound from a road segment whenever an attenuation different from zero is found due to barrier diffraction. The criterion for this decision may be modified in the future to be 5 or 10 dB, corresponding to path length differences of 0 to 0.1 ft at 500 Hz, if field experience shows that this is necessary. Furthermore, the decision to neglect ground strips, which is related to certain heights of rays above the center line of the ground strip, is not based on experimental evidence and may require modifications.

PRELIMINARY SELECTION OF REFLECTORS

By the introduction of an image receiver location, the reflection problem becomes similar to the diffraction problem. A reflector in the path from the road segment to an image receiver is effective whenever a barrier in the path from the road segment to a receiver strongly diffracts the sound.

In addition to the preliminary checks made above for diffracting barriers, the intensity of the direct (or diffracted) sound from the road segment considered is compared to the potential maximum contribution of each reflection. This check assures that only essential reflectors are considered. Since a reflector might not be high enough or reflections might be strongly attenuated by diffraction at additional barriers in the ray path, the reflectors found at this stage are considered "potential" reflectors.

Also involved in the part entitled "Preliminary Selection of Reflectors" is the preliminary selection of barriers which can possibly diffract the reflected sound. The check performed is based on the length and orientation of a barrier section and on the distance from the image receiver relative to the respective parameters of the road section and the receiver considered.

CALCULATION OF POTENTIAL REFLECTIONS

The computations in this section are analogous to those made for the diffraction problem except for a simplified decision concerning the height of the reflector. If the height difference above the reflector of a sound ray from the nearest point on the road segment leads to the acceptance or rejection of the reflector, no further check is made for other source points on the road segment.

Since diffraction of the reflected ray is not yet checked, a reflection is still called a potential one and the end point of the road segment is preliminary.

CONSIDERATION OF DIFFRACTION BEFORE REFLECTION

Checks for barriers in the area defined by projections of the four points XR1, XR2, XRB3, and XRB4 into the x-y plane are made by considering first the triangle containing image rays and then the triangle formed by the road segment and the image receiver.

If a barrier is found to be high enough for possible diffraction, the reflector is checked again to determine whether or not it is high enough to reflect the diffracted rays (which now come from an effective source that might be considerably higher than the roadway).

Calculation of the path length difference between diffracted and direct rays from only the near point of the road segment implies that the diffraction of sound rays from other source points is about the same.

Very strongly diffracted reflections are neglected. The decision is made on the basis of the diffraction of sound from trucks, since rays from cars are even more strongly diffracted.

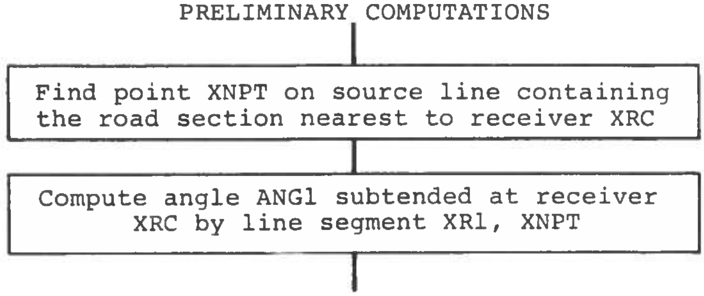
CONSIDERATION OF DIFFRACTION AFTER REFLECTION

Checks are made for diffracting barriers in the triangle defined by projections of the reflector segment XRB3, and the receiver XRC onto the x-y plane.

After a barrier has been found which is high enough for possible diffraction, the reflector is checked to determine whether or not it is high enough to reflect sound towards the top line of the diffracting barrier, which might be considerably higher than the receiver.

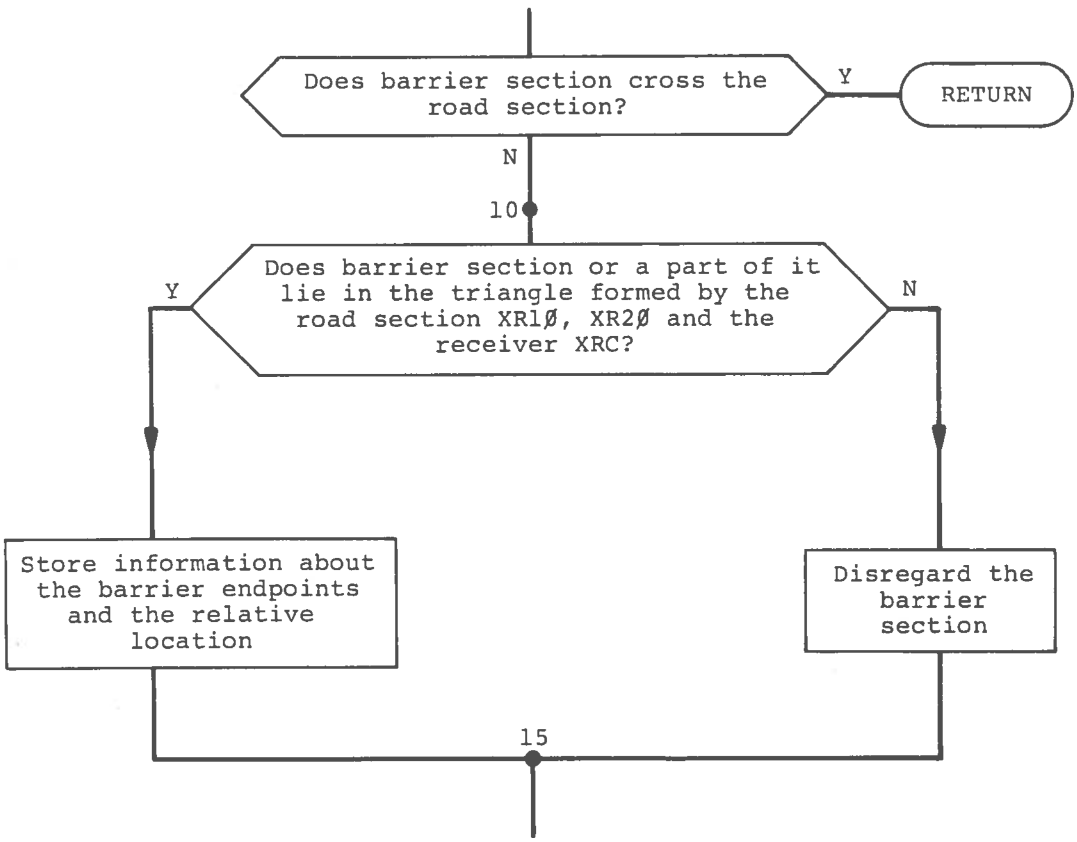
Calculation of the path length difference implies simplifying assumptions similar to those for the problem of diffraction before reflection.

PRELIMINARY COMPUTATIONS

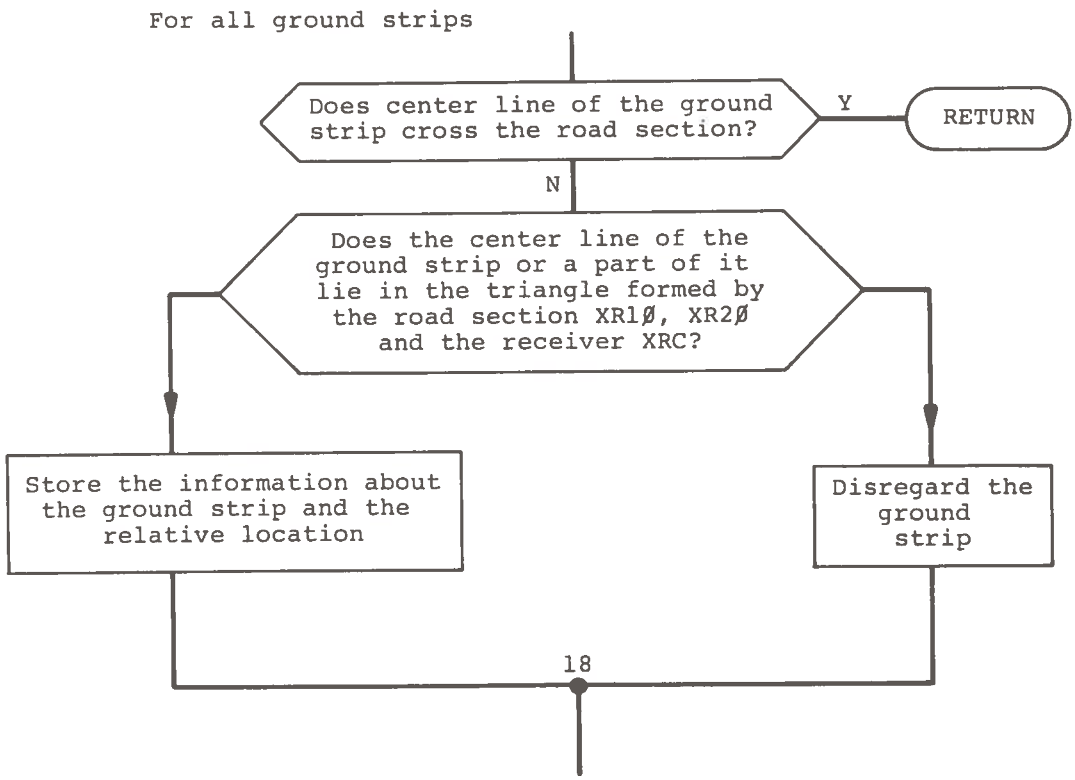


PRELIMINARY SELECTION OF DIFFRACTING BARRIERS

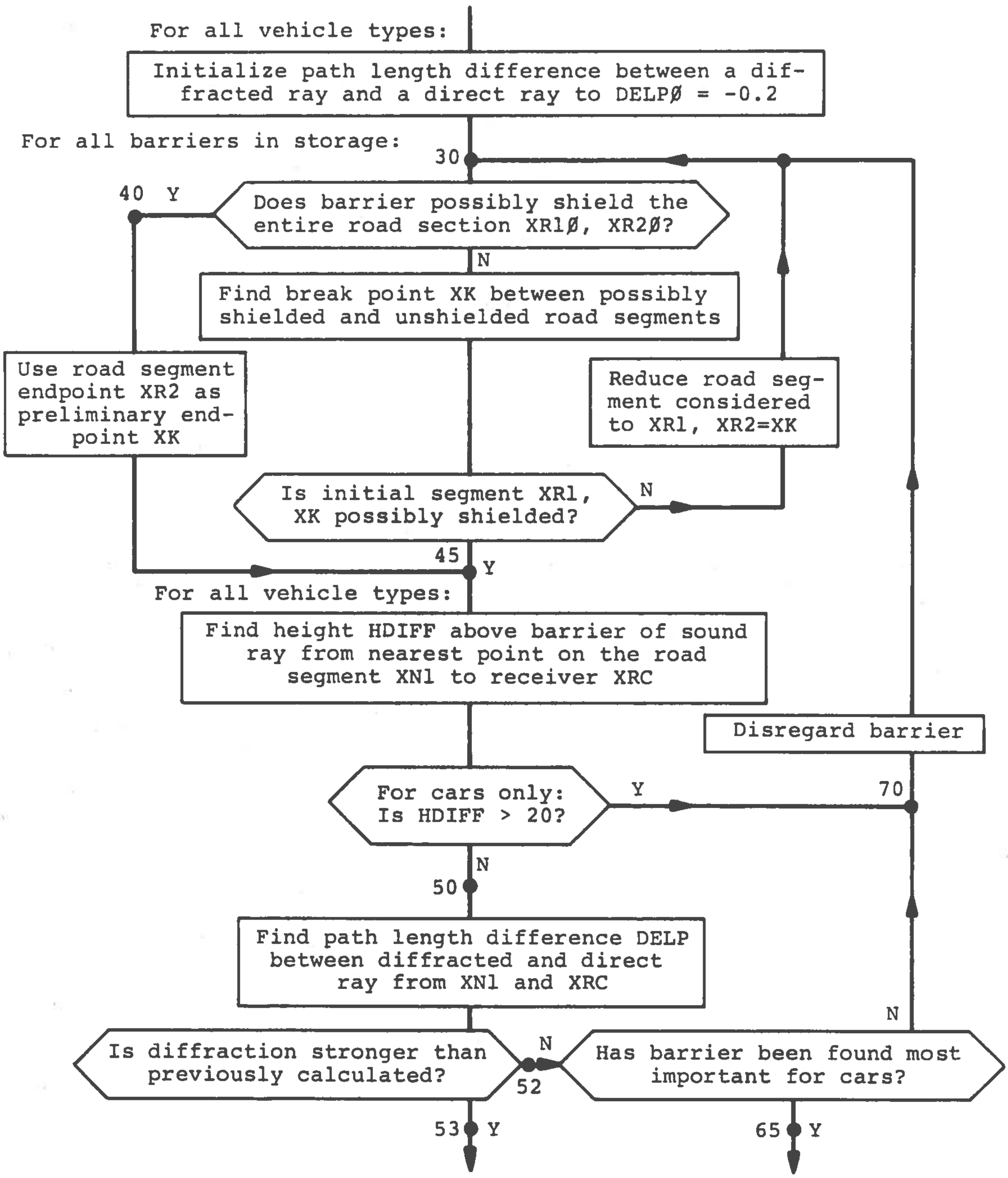
For all sections of all barriers:

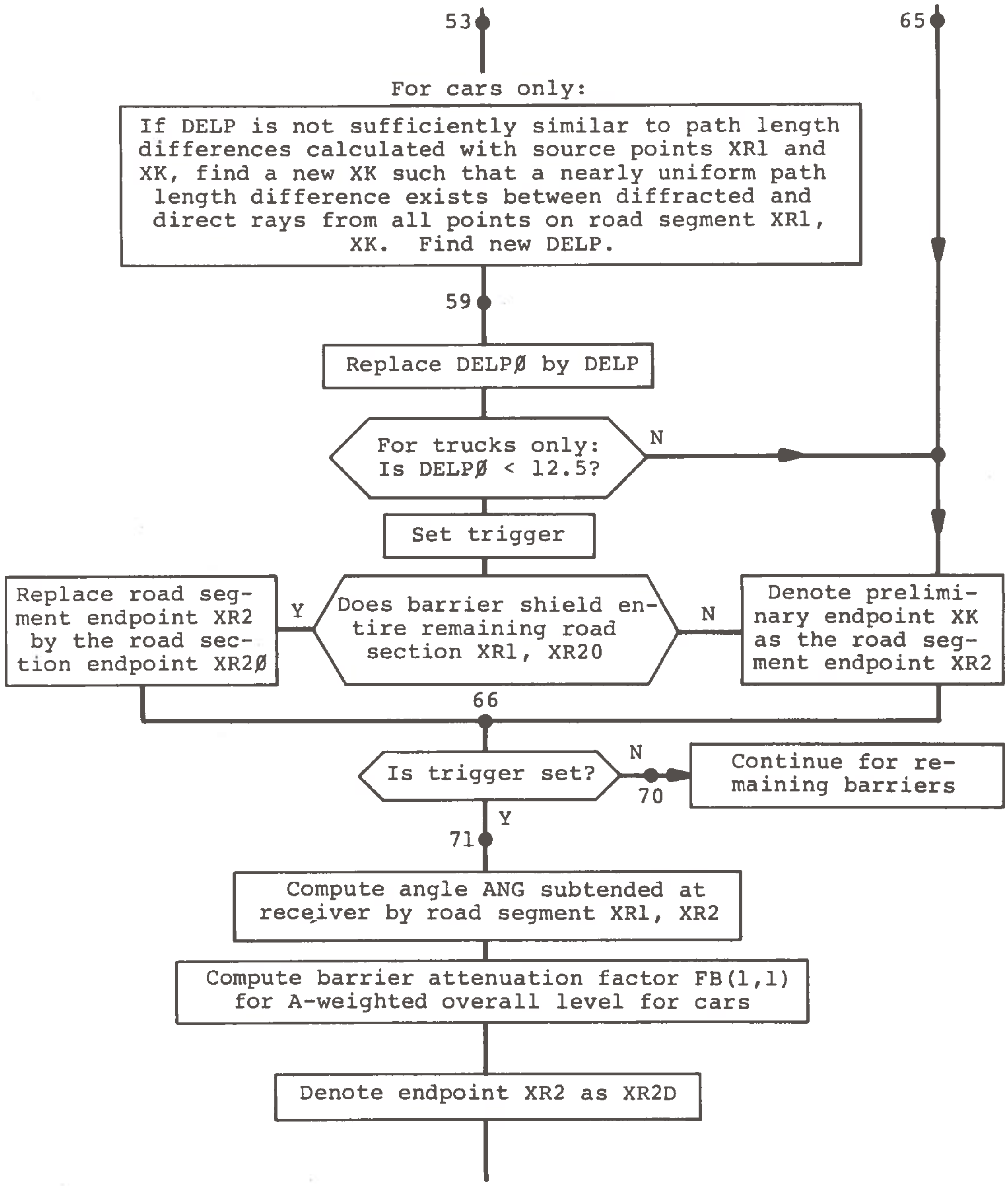


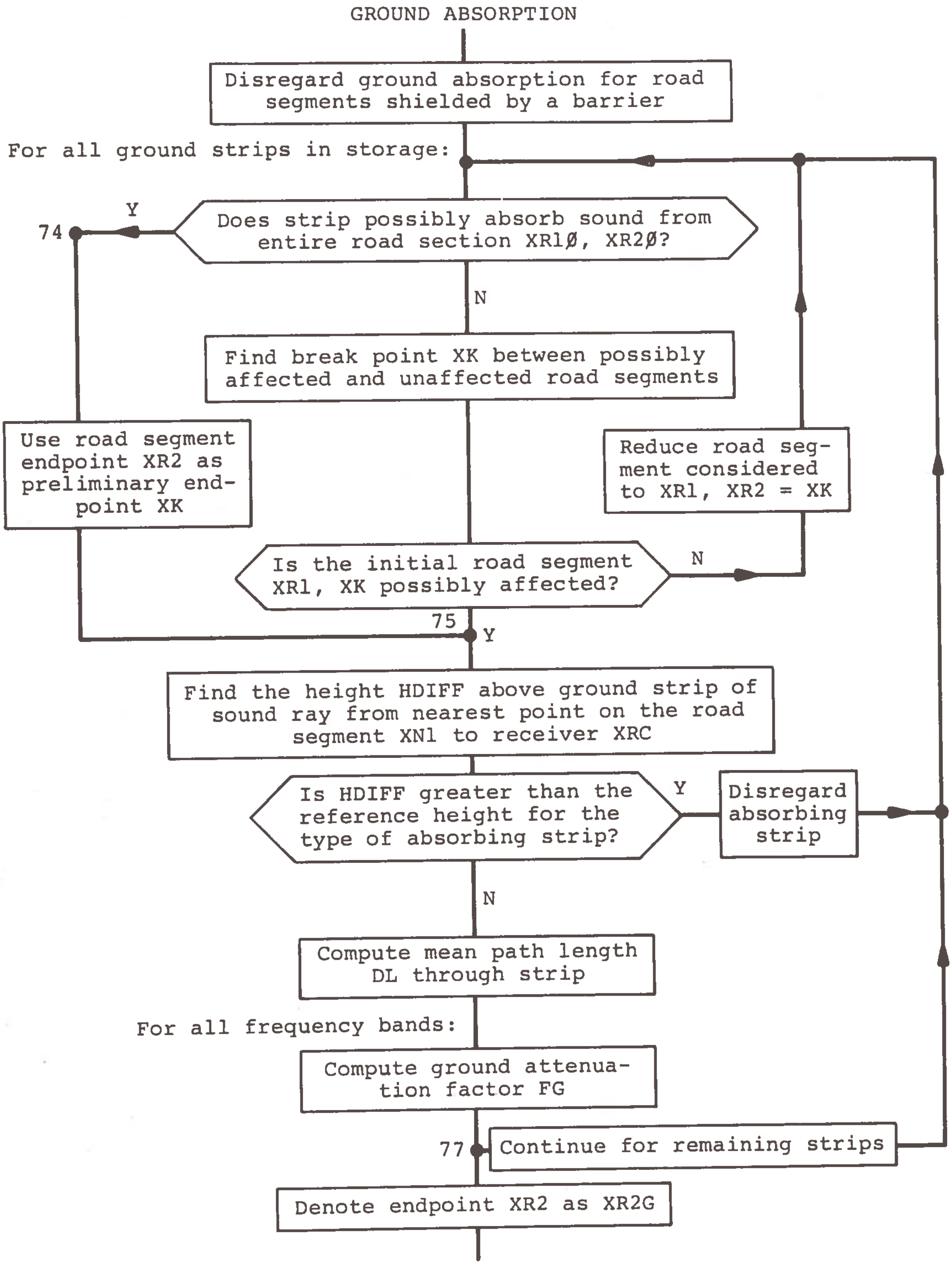
PRELIMINARY SELECTION OF ABSORBING GROUND STRIPS



DIFFRACTION OF DIRECT RAY

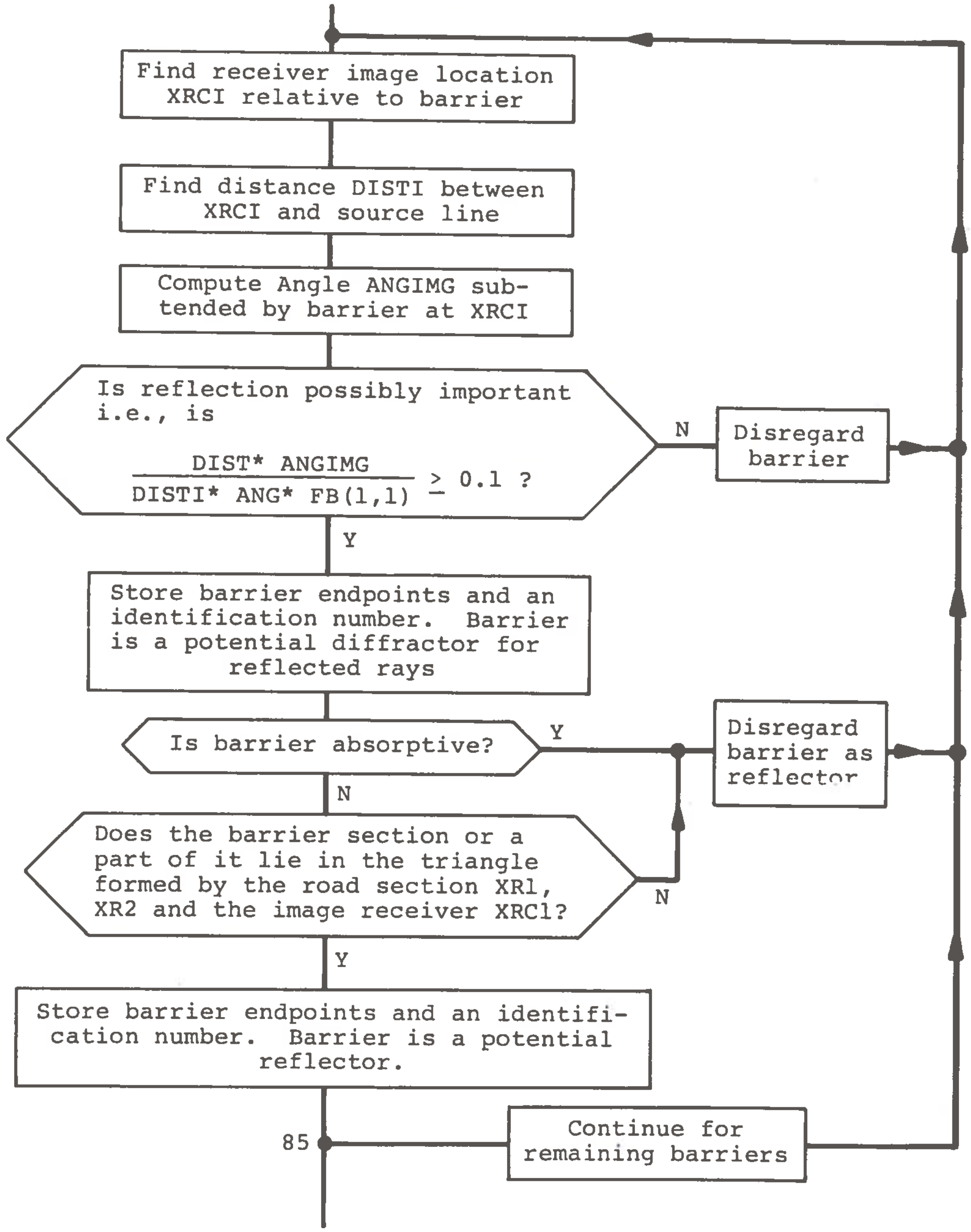



53
65
For cars only:
If DELP is not sufficiently similar to path length differences calculated with source points XR1 and XK, find a new XK such that a nearly uniform path length difference exists between diffracted and direct rays from all points on road segment XR1, XK. Find new DELP.
59
Replace DELPØ by DELP
For trucks only: Is DELPØ < 12.5?
N
Set trigger
Replace road segment endpoint XR2 by the road section endpoint XR2Ø
Y
Does barrier shield entire remaining road section XR1, XR20
N
Denote preliminary endpoint XK as the road segment endpoint XR2
66
Is trigger set?
N
70
Continue for remaining barriers
Y
71
Compute angle ANG subtended at receiver by road segment XR1, XR2
Compute barrier attenuation factor FB(1,1) for A-weighted overall level for cars
Denote endpoint XR2 as XR2D

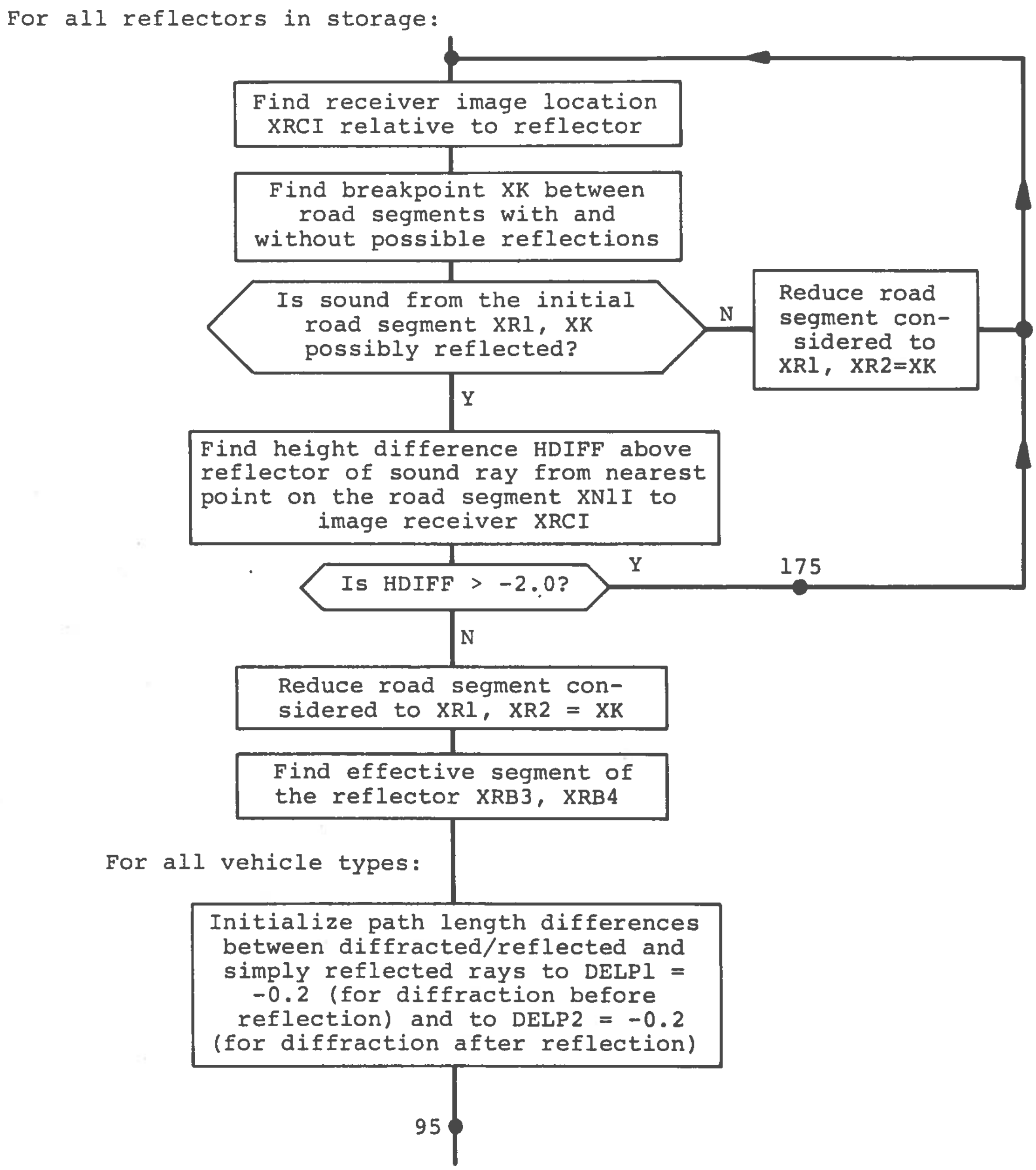

GROUND ABSORPTION
Disregard ground absorption for road segments shielded by a barrier
For all ground strips in storage:
Does strip possibly absorb sound from entire road section XR1Ø, XR2Ø?
Y
74
N
Find break point XK between possibly affected and unaffected road segments
Use road segment endpoint XR2 as preliminary end-point XK
Reduce road seg-ment considered to XR1, XR2 = XK
Is the initial road segment XR1, XK possibly affected?
N
75
Y
Find the height HDIFF above ground strip of sound ray from nearest point on the road segment XN1 to receiver XRC
Is HDIFF greater than the reference height for the type of absorbing strip?
Y
Disregard absorbing strip
N
Compute mean path length DL through strip
For all frequency bands:
Compute ground attenua-tion factor FG
77
Continue for remaining strips
Denote endpoint XR2 as XR2G

PRELIMINARY SELECTION OF REFLECTORS

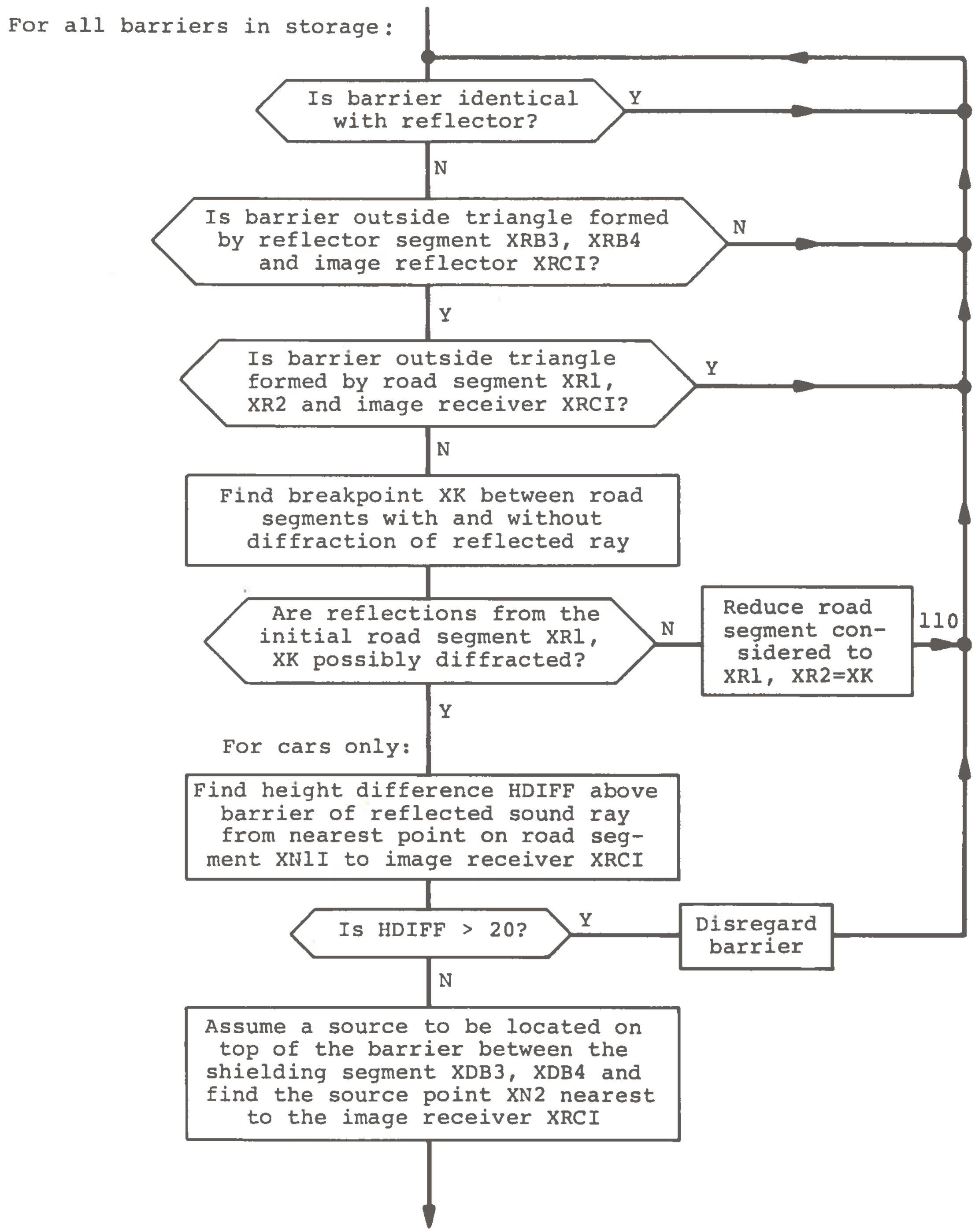
For all sections of all barriers in storage

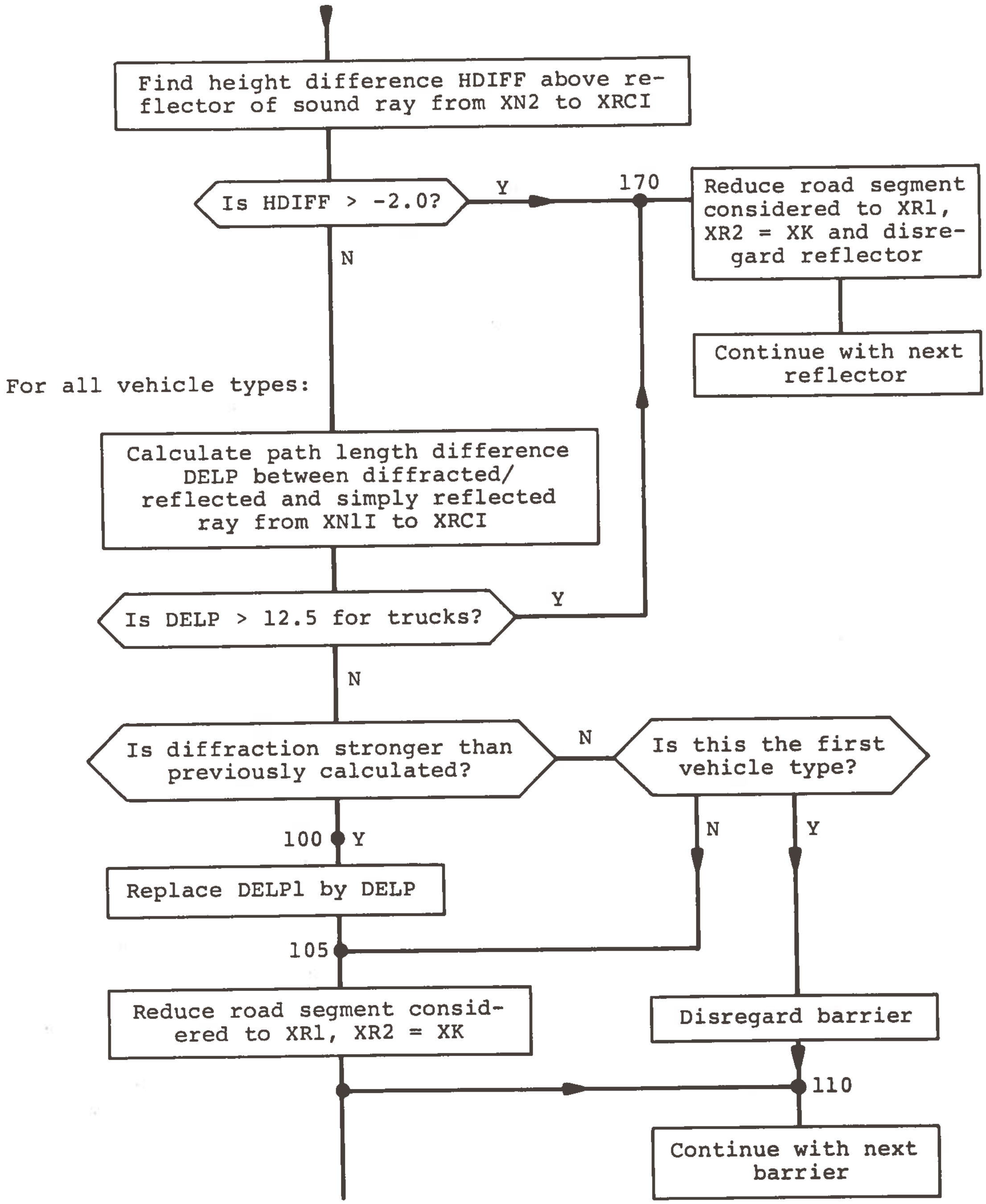


CALCULATION OF POTENTIAL REFLECTIONS

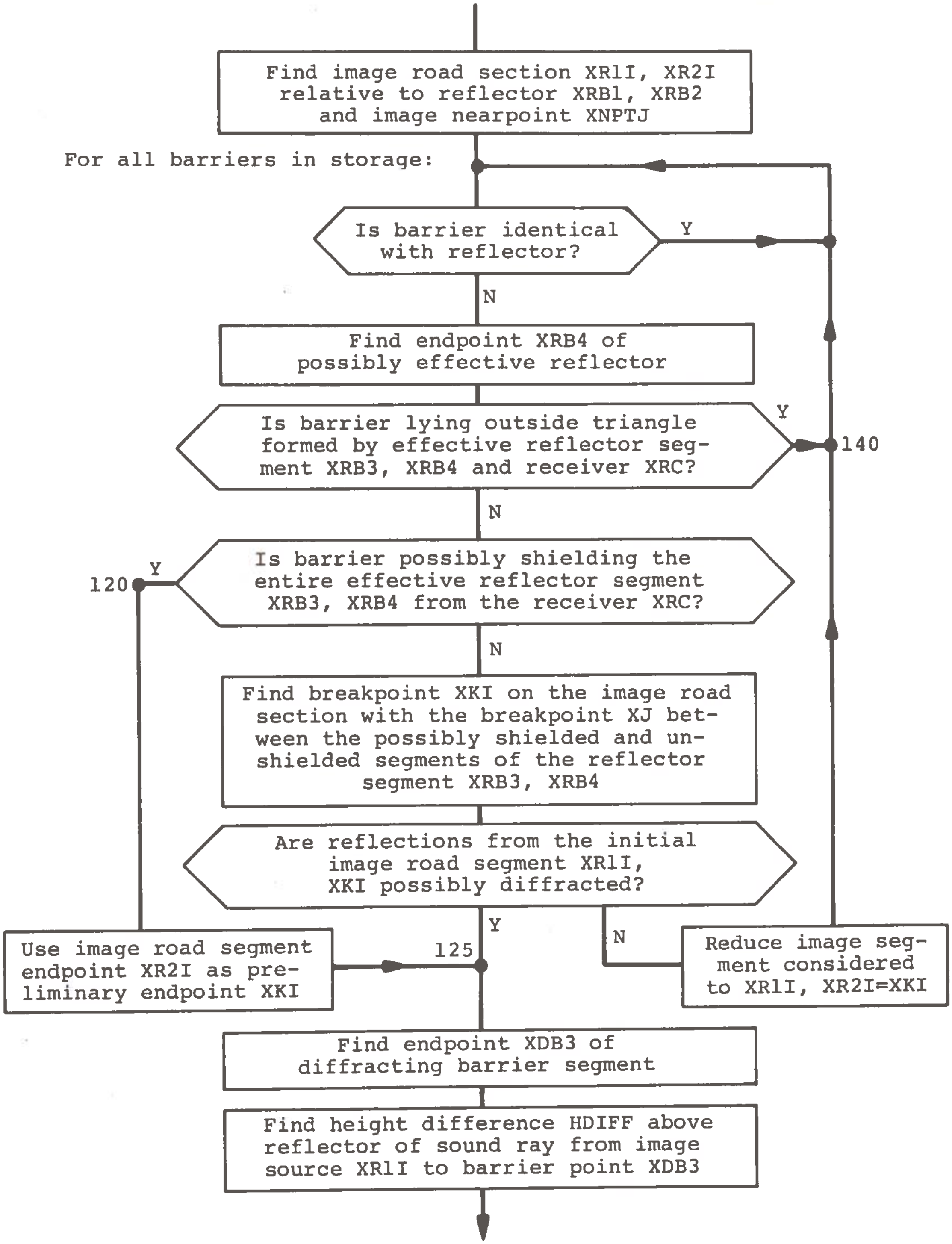


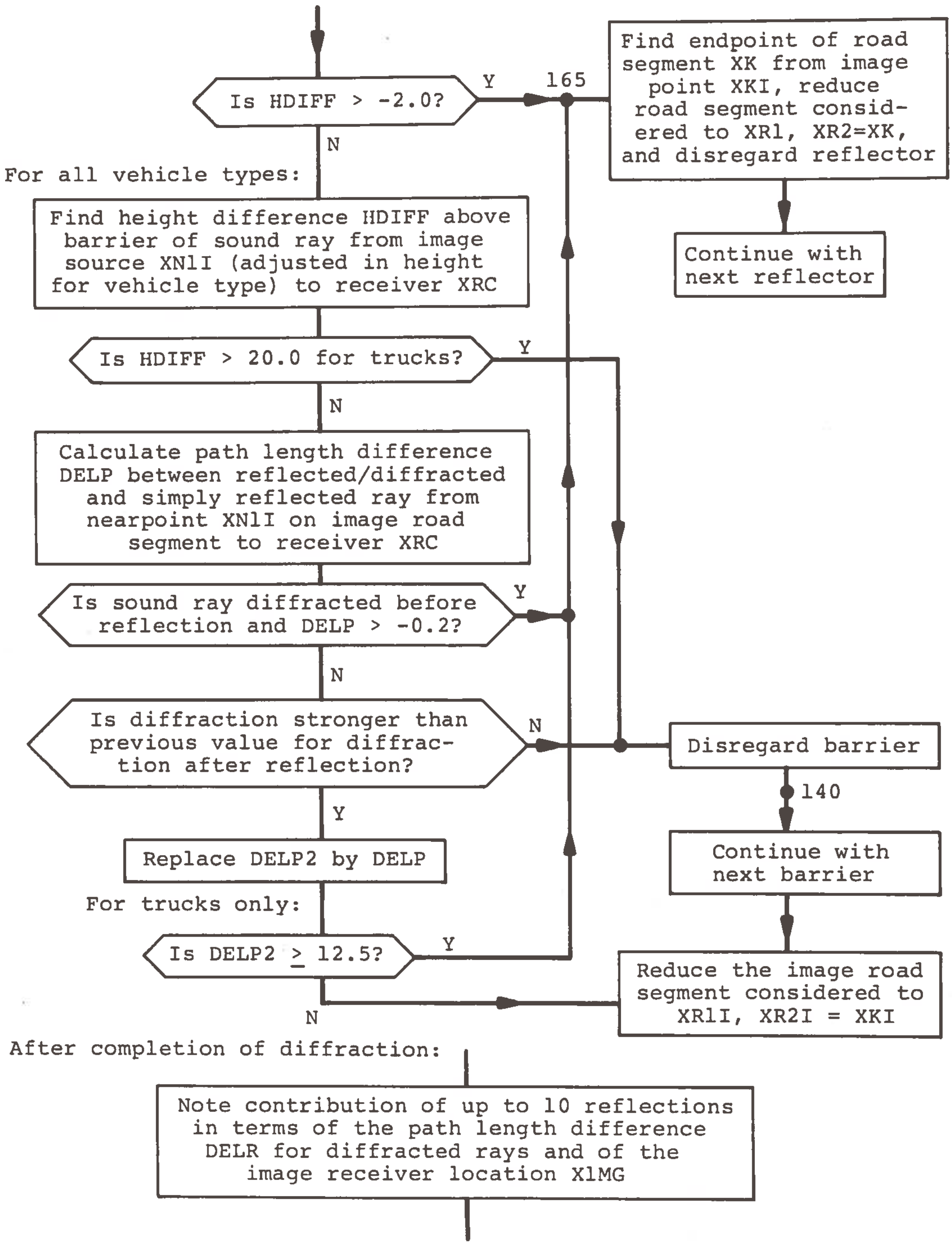
CONSIDERATION OF DIFFRACTION BEFORE REFLECTION




Find height difference HDIFF above reflector of sound ray from XN2 to XRCI
Is HDIFF > -2.0?
Y
170
Reduce road segment considered to XR1, XR2 = XK and disregard reflector
Continue with next reflector
N
For all vehicle types:
Calculate path length difference DELP between diffracted/ reflected and simply reflected ray from XN1I to XRCI
Is DELP > 12.5 for trucks?
Y
N
Is diffraction stronger than previously calculated?
N
Is this the first vehicle type?
100
Y
N
Y
Replace DELP1 by DELP
105
Reduce road segment considered to XR1, XR2 = XK
Disregard barrier
110
Continue with next barrier

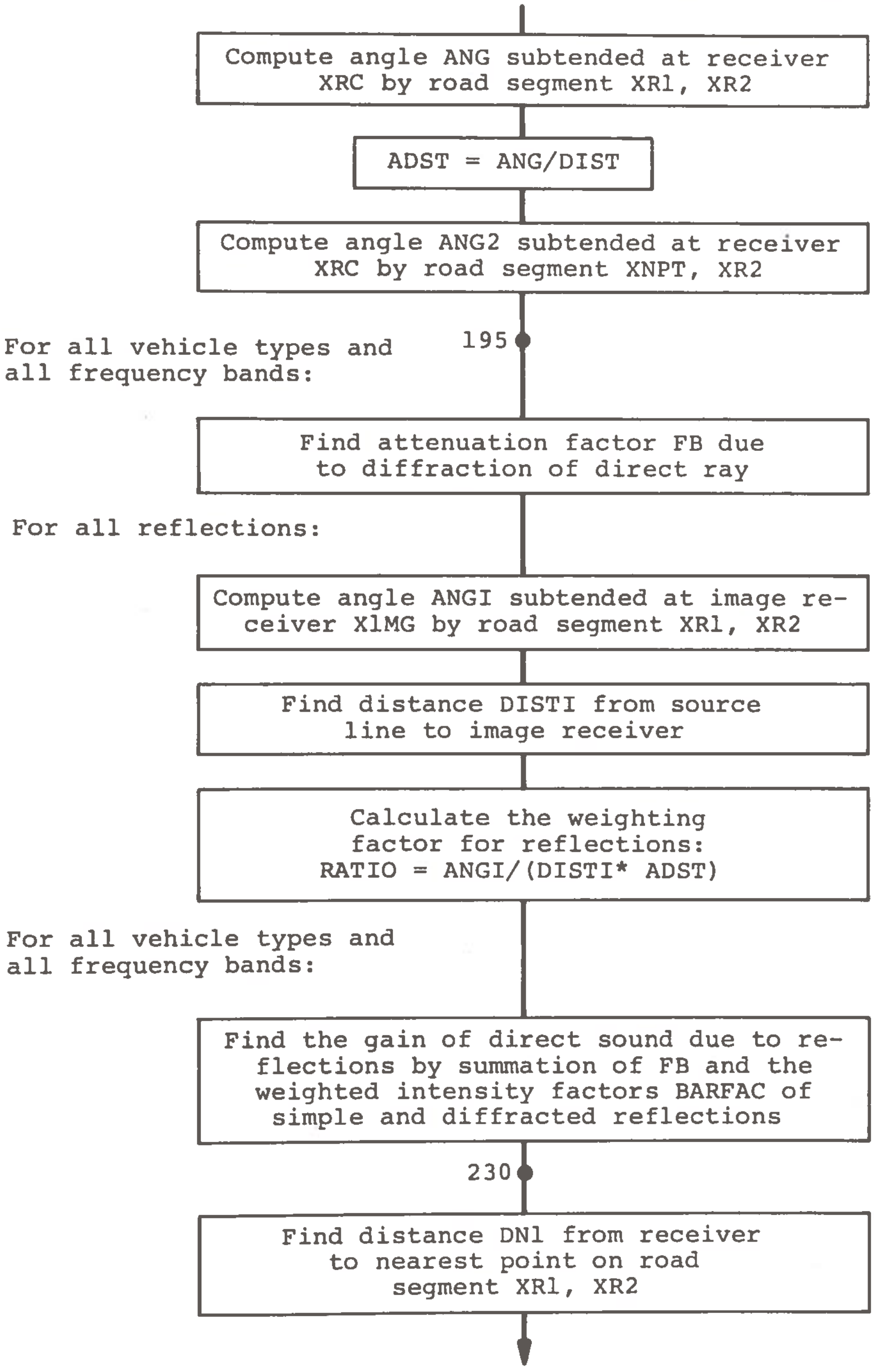
CONSIDERATION OF DIFFRACTION AFTER REFLECTION



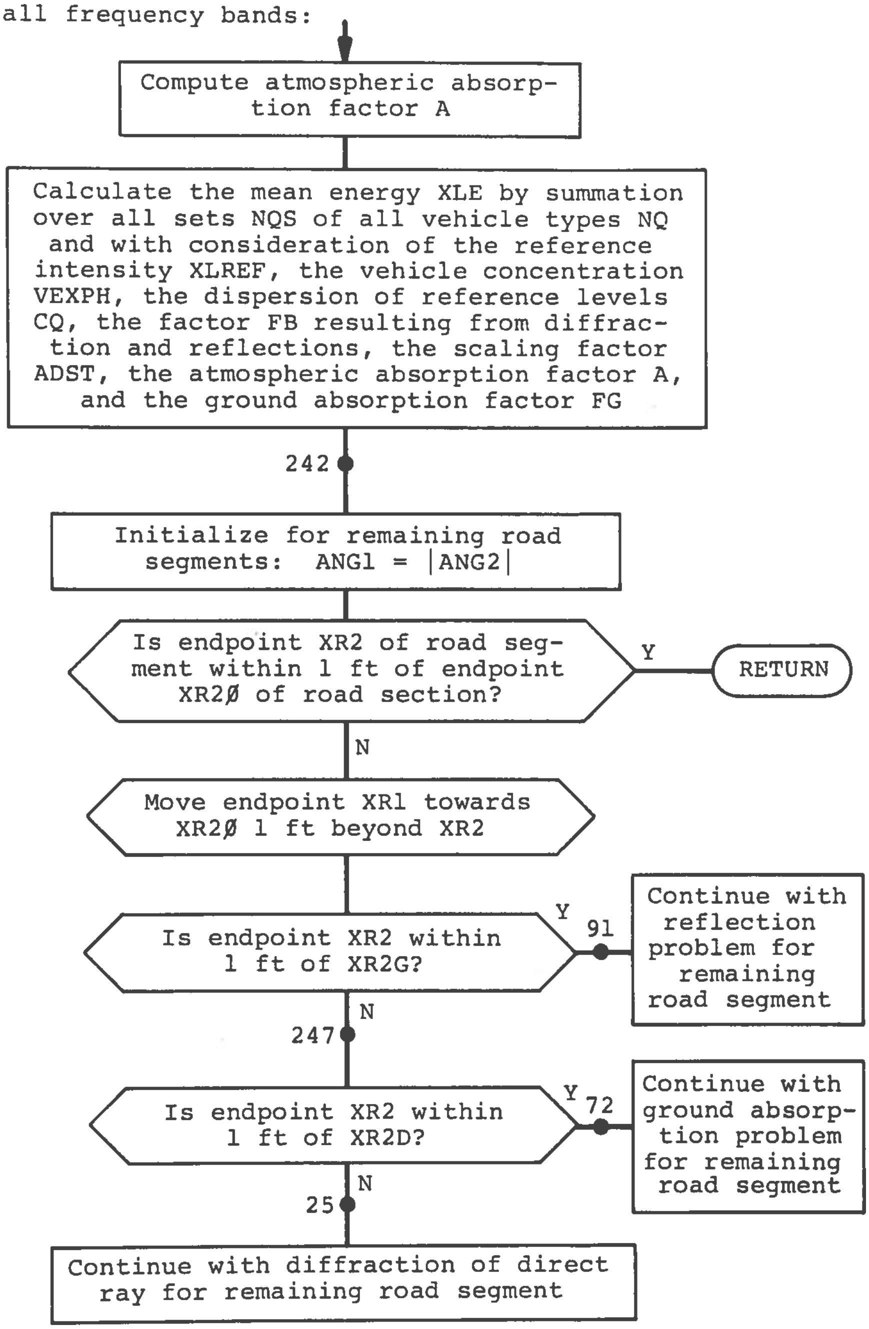


Is HDIFF > -2.0?
Y
165
Find endpoint of road segment XK from image point XKI, reduce road segment considered to XR1, XR2=XK, and disregard reflector
N
For all vehicle types:
Find height difference HDIFF above barrier of sound ray from image source XN1I (adjusted in height for vehicle type) to receiver XRC
Continue with next reflector
Is HDIFF > 20.0 for trucks?
Y
N
Calculate path length difference DELP between reflected/diffracted and simply reflected ray from nearpoint XN1I on image road segment to receiver XRC
Is sound ray diffracted before reflection and DELP > -0.2?
Y
N
Is diffraction stronger than previous value for diffraction after reflection?
N
Disregard barrier
140
Y
Replace DELP2 by DELP
Continue with next barrier
For trucks only:
Is DELP2 ≥ 12.5?
Y
Reduce the image road segment considered to XR1I, XR2I = XKI
N
After completion of diffraction:
Note contribution of up to 10 reflections in terms of the path length difference DELR for diffracted rays and of the image receiver location X1MG

FINAL COMPUTATIONS AND SUMMATION OF SOUND INTENSITIES



For all frequency bands:



Reflections are neglected in the case of diffraction before and after reflection and in the case of very strong diffraction after reflection.

FINAL COMPUTATIONS AND SUMMATION OF SOUND INTENSITIES

In order to facilitate future calculation of higher order statistical parameters of the noise, the contributions from all reflections are used to calculate a gain factor of the direct sound from a road segment. The subsequent computations are directly oriented at the formula of Equation A-8 for the mean sound intensity.

Also intended for future use is the separate computation of angles ANG1 and ANG2, of which only the absolute value of the difference is employed in Equation A-8.

After the calculation for a road segment has been completed, a point 1 ft beyond the end point of the segment is taken as the initial point of the next road segment. This procedure avoids computational complications without unduly sacrificing accuracy.

LIST OF VARIABLES FOR SUBROUTINE GEOMRY

A	Atmospheric attenuation factor, ground absorption parameter
ADST	ANG/DIST
ANG	Angle subtended at receiver by road segment
ANG1	Angle α_1 in Fig. A-1
ANG2	Angle α_2 in Fig. A-1
ANGI	Angle subtended at image receiver by road segment
ANGIMG	Angle subtended at image receiver by barrier section
B1	End point of barrier section
BGS	Width of absorptive ground strip
BGT	Width of absorptive ground strip
BX	x-coordinate of barrier point
BY	y-coordinate of barrier point
BZ	z-coordinate of barrier point
CA1	$\cos(\alpha_1)$

CA2	$\cos(\alpha_2)$
CPREV	Angular function C_{n-1}, defined in Equation A-10
CQ	Factor accounting for standard deviation of reference level
DELM	Mean path length difference
DELP	Path length difference
DELPØ	Maximum path length difference for diffraction of direct ray
DELP1	Maximum path length difference for diffraction before reflection
DELP2	Maximum path length difference for diffraction after reflection
DELPA	Path length difference for diffraction of direct ray
DELR	Path length difference for reflected ray
DELTA	Distance along the roadway
DIST	Distance from the receiver to the source line
DISTI	Distance from the image receiver to the source line
DISTJ	Distance from the image receiver to the diffracting barrier
DL	Mean path length over an absorptive ground strip
DN1	Distance from the receiver to the nearest point of the road segment
DN1I	Distance from the image receiver to the nearest point of the road segment
DN2	Distance from the image receiver to the nearest point of the diffracting barrier
DR1	Distance from the receiver to the initial point of the road segment
DRK	Distance from the receiver to the preliminary end point of the road segment
FB	Attenuation factor accounting for diffraction and reflections
FCTR	Weighting factor for reflections

FG	Ground attenuation factor
HDIFA	Height of ray from source point XR1 to receiver XRC above barrier
HDIFF	Height of ray above barrier, reflector, or ground strip
HGA	Data for effective height of ground cover
I	Index
IA	Alphanumeric "A"
IBAR	Barrier number
IBLAST	Barrier type
ICODE	Number for intermediate printout
IDUM	Index for kind of absorptive ground cover
IDXR	Number of reflections
IERR	Error index
IGRA	Ground strip number
IF	Index for frequency bands
II	Index
IK	Frequency band number
IKIN	Index for kind of absorptive ground cover
IP	Frequency band number
IQ	Index
IR	Alphanumeric "R"
ISEG	Barrier section number
IT3	2N - 1
ITRIG	Trigger
KAR	Alphanumeric indicator for type of barrier
KBAR	Total number of barrier sections; index for barriers
KBAR1	Reflector number in storage
KBAR2	Diffractor number in storage

KBCODE	Indicator for relative location of barrier
KCD	Indicator for relative location of barrier
KDIFF	Barrier number in storage
KF	Index for frequency bands
KGA	Number of ground strips stored
KGCODE	Indicator for relative location of ground strip
KIMG	Reflection number
KNUMB	Total number of relevant barrier sections
KREF	Reflector number stored
KRDNUM	Total number of barrier sections relevant to reflection
KRFDF	Barrier number stored
KRNUM	Total number of relevant reflector sections
KTRIG	Indicator for intersection of barrier or ground strip
LOC	Indicator for relative location of barrier or ground strip
MDIFF	Indicator for diffraction before reflection
MODD	Indicator for diffraction of direct ray
MR	Roadway number
N	Cumulant numbers
NB	Number of barriers
NBSEC	Number of barrier sections
NBSM1	Number of sections for one barrier
NDIFF	Number of barriers stored
NF	Number of frequency bands
NG	Number of absorptive ground strips
NIMG	Number of reflections
NLIM	Number of points defining one barrier
NQ	Number of vehicle types

NQQ	Number of groups within one vehicle type
NQS	Vector notation for number of vehicle groups
NR	Number of roadways
NREF	Number of reflectors stored
NRFDF	Number of barriers stored
PP	Frequency band number
R1	End point of potential reflector
RATIO	Weighting factor for reflected rays
RB1	End point of barrier in path of reflected ray
RDIN	Vector notation for initialization parameters
SA1	$\sin(\alpha_1)$
SA2	$\sin(\alpha_2)$
T1	Temporary variable
T2	Temporary variable
T3	Temporary variable
TA1	End point on center line of absorptive ground strip
VEXPH	Vehicles per foot
XB1	Initial point of barrier stored
XB2	End point of barrier stored
XDB1	Initial point of barrier stored
XDB2	End point of barrier stored
XDB3	Initial point of effective barrier segment
XDB4	End point of effective barrier segment
XG1	Initial point of center line of absorptive ground strip
XG2	End point of center line of absorptive ground strip
XG3	Initial point of effective ground strip segment
XG4	End point of effective ground strip segment
XIMG	Vector of image receivers for all reflections

XJ	Preliminary end point of effective reflector segment
XK	Preliminary end point of road segment
XKA	Cumulant of the A-weighted sound intensity
XKI	Preliminary end point of image road segment
XLA	A-weighted intensity in frequency bands
XLE	Mean intensity
XLREF	Vector notation for reference intensities
XN1	Point on road segment nearest to receiver
XN1I	Point on road segment nearest to image receiver; point on image road segment nearest receiver
XN2	Point on barrier segment nearest to image receiver
XNPT	Point on source line nearest to receiver
XNPTI	Point on source line nearest to image receiver
XNPTJ	Point on image source line nearest receiver
XR	X-coordinate of receiver
XR1	Initial point of road segment
XR1Ø	Initial point of road section
XR1I	Initial point of image road segment
XR2	End point of road segment
XR2Ø	End point of road section
XR2D	End point of road segment with constant attenuation by diffraction
XR2G	End point of road segment with constant attenuation by ground absorption
XR2I	End point of image road segment
XRB1	Initial point of reflector stored
XRB2	End point of reflector stored
XRB3	Initial point of effective reflector segment
XRB4	End point of effective reflector segment

XRC	Receiver point
XRCI	Image receiver point
XXG1	X-coordinate of point on ground strip center line
YR	Y-coordinate of receiver
YYG1	Y-coordinate of point on ground strip center line
ZN1Ø	Z-coordinate of XN1 or XN1I
ZR	Z-coordinate of receiver
ZS	Height adjustment for vehicles
ZZG1	Z-coordinate of point on ground strip center line

DESCRIPTION OF OTHER SUBROUTINES AND FUNCTIONS

SUBROUTINE FILES

FILES defines file device numbers of the three files with roadway, barrier, and absorptive ground strip data created by the subroutine INPUT and for the four subfiles with barrier and ground strip data created by the subroutine GEOMRY.

SUBROUTINE INTER (NR,IQ)

INTER interpolates between stored A-weighted overall and octave band reference levels to find reference intensities in frequency band numbers IQ as a function of the average speed of vehicles in a group.

FUNCTION AMAG (X1V, X2V)

AMAG is a function subroutine which calculates the distance between the points X1V and X2V.

FUNCTION ANGLE (X1V, X2V, X3V)

ANGLE is a function subroutine which calculates the angle at point X3V that is subtended by the line between X1V and X2V. The angle is positive with values between 0 and π.

FUNCTION BARFAC (KF, DELP)

BARFAC is a function subroutine which calculates the attenuation factor due to barrier diffraction as a function of the frequency band number KF and of the path length difference DELP of the diffracted ray path relative to the direct ray path.

SUBROUTINE BLOCN (X1V, X2V, X3V, X4V, X5V, X6V, LOC)

BLOCN calculates the location LOC of a barrier or a ground strip, defined by the end points X4V, and X5V, relative to the triangle formed by two points X1V and X2V on a road segment and the receiver point X3V (Figure B-3). The point X6V is identical with X2V if both points X4V and X5V are outside the triangle. If one of the points X4V or X5V is inside the triangle, X6V is the intercept of a line through X3V and X4V with the road segment X1V, X2V. If both points X4V and X5V are inside the triangle, then the point X6V is the intercept of the road segment X1V, X2V with the line through X3V and either X4V or X5V depending on which intercept is nearer to the point X1V.

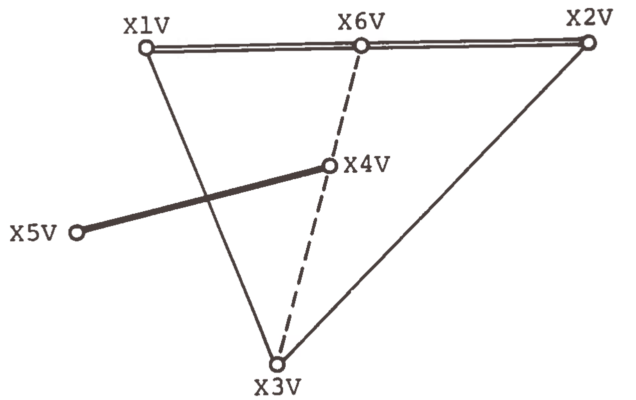


Figure B-3 Notation for Subroutine BLOCN

LOC = 0 denotes that the line X4V, X5V is outside the triangle X1V, X2V, X3V.

LOC = 1 denotes that the line X4V, X5V intersects the line X1V, X3V.

LOC = 2 denotes that the line X4V, X5V intersects the line X2V, X3V.

LOC = 3 denotes that the line X4V, X5V intersects both lines X1V, X3V and X2V, X3V.

LOC = 4 denotes that the line X4V, X5V is completely inside the triangle X1V, X2V, X3V.

FUNCTION DEL (X1V, X2V, X3V, X4V, HDIFF, DN1)

DEL is a function subroutine which calculates the path length difference between the ray emanating from the source point X1V and diffracted over the barrier X3V, X4V towards the receiver X2V and the direct ray from X1V to X2V. HDIFF is the height of the direct ray above the barrier, and DN1 is the path length traveled by the direct ray.

SUBROUTINE ENDPT (X1V, X2V, X3V, X4V, X5V, X6V, KTRIG, IERR)

ENDPT calculates the new end point of a road segment from which sound is uniformly attenuated during propagation towards the receiver X3V. The original road segment has the initial point X1V and the end point X2V. A barrier, reflector, or ground strip has the end points X4V, X5V. X6V is the break point on the road between segments that are affected and unaffected by the barrier, reflector, or ground strip. If the entire road segment X1V, X2V is unaffected, KTRIG = 0 and X6V is identical with X2V. If the entire road segment X1V, X2V is affected, KTRIG = 1 and X6V is identical with X2V. If a break point X6V is at a distance of less than 0.51 ft from the initial point X1V, this break point is disregarded. If the initial road segment X1V, X6V is affected by a barrier, reflector, or ground strip, KTRIG = 1, the break point X6V is moved toward X1V by 0.5 ft, and the new end point is denoted as X6V. If the initial road segment is unaffected, KTRIG = 0, the break point X6V is moved towards X1V by 0.5 ft, and the new end point is denoted as X2V.

The error indicator IERR is set in the SUBROUTINE MOVE.

FUNCTION HEIGHT (X1V, X2V, X3V, X4V)

HEIGHT is a function subroutine which calculates the height of a ray from point X1V to point X2V above the line with the end points X3V, X4V.

FUNCTION IEPS (X1V, X2V, X3V, X4V, DEL1)

IEPS is a function subroutine which calculates whether the path length difference DEL1 is sufficiently similar to the path length difference between the ray emanating from X1V and diffracted over the barrier X3V, X4V towards the receiver X2V and the direct ray from X1V to X2V. If the two path length differences are sufficiently similar, IEPS = 1; otherwise, IEPS = 0.

SUBROUTINE IMAGE (X1V, X2V, X3V, X4V)

IMAGE calculates the location of an image receiver X4V relative to a receiver X2V and a reflector with the end points X1V, X2V.

SUBROUTINE INTCPT (X1V, X2V, X3V, X4V, X5V)

INTCPT calculates in the x-y plane the intercept X5V of two lines specified by the two pairs of points X1V, X2V and X3V, X4V.

FUNCTION KCUT (X1V, X2V, X3V, X4V)

KCUT is a function subroutine which checks whether two line segments with the end points X1V, X2V and X3V, X4V, respectively, cross in the x-y plane. If the two line segments cross, KCUT = 1; otherwise, KCUT = 0.

FUNCTION KPOS (X1V, X2V, X3V)

KPOS is a function subroutine which checks whether the point X3V lies between two points X1V and X2V on a line. If it does, KPOS = 1; otherwise, KPOS = 0.

SUBROUTINE MIDP (X1V, X2V, X3V)

MIDP calculates the center point X3V between two points X1V and X2V.

SUBROUTINE NRPT (X1V, X2V, X3V, X4V, DIST)

NRPT calculates the point X4V, on a line specified by the points X1V, X2V, that is nearest to the point X3V. It also calculates the distance DIST between the points X3V and X4V. If the distance is zero, DIST = 1.

SUBROUTINE NR1 (X1V, X2V, X3V, X4V, DIST, X5V, DN1)

NR1 calculates the point X5V, on a line segment between the end points X1V, X2V, that is nearest to the point X3V. It also calculates the distance DN1 between the points X3V and X5V. The subroutine employs as input parameters the distance DIST between the receiver and the nearest point X4V on the entire line.

SUBROUTINE REPLCE (X1V, X2V)

REPLCE sets the vector X2V equal to the vector X1V.

SUBROUTINE SECTN (X1V, X2V, X3V, X4V, X5V, X6V, X7V)

SECTN calculates the segment X6V, X7V of the line between X4V and X5V that is bounded by the two lines from X3V to X1V and from X3V to X2V.

SUBROUTINE TRI (X1V, X2V, X3V, X4V, X5V, KTRI)

TRI determines whether or not the point X4V lies in the triangle formed by the points X1V, X2V, X3V. If X4V is in the triangle, KTRI = 1; otherwise, KTRI = o. TRI also calculates the intercept X5V of two lines specified by the points X1V, X2V and X3V, X4V, respectively.

FUNCTION ZCOR (X1V, X2V, X3V)

ZCOR is a function subroutine which calculates the z-coordinate of the point X3V so that the point X3V lies on the line specified by the points X1V and X2V.

SUBROUTINE MOVE (X1V, X2V, X3V, DELTA, IERR)

MOVE calculates the point X2V, on the line specified by the points X1V and X2V, that is at a distance of DELTA feet from the point X1V in the direction away from X3V. MOVE gives the error message IERR = 4 if the two points X1V and X3V are identical.

FUNCTION TAN (X)

TAN is a function subroutine which calculates the tangent of X.

FUNCTION ACOS (X)

ACOS is a function subroutine which calculates the arc cos of X between 0 and π.

CONSTRAINTS ON INPUT DATA

The storage of input data in core memory is laid out for a maximum number of

10 roadways,

10 barrier, and

10 ground strips.

Further constraints given in the "User's Manual" are maximum numbers of

5 speed groups each for trucks and passenger cars,

10 sections for each roadway,

10 sections for each barrier, and

15 receivers.

These constraints may, of course, be changed through appropriate changes to the dimension statements in the FORTRAN IV program, provided adequate core storage is available.

```
C     TRAFFIC NOISE PREDICTION MODEL
C     MAIN PROGRAM     12/14/71
      DIMENSION SIG(4),XKAP(4)
      DIMENSION XR1(3),XR2(3)
      COMMON/INOU/INPT,IOUT
      COMMON/BLK2/NQ
      COMMON/INPT1/RDIN(15)
      COMMON/INPT2/NR,NB,NG
      COMMON/INPT3/XRC(15),YRC(15),ZRC(15),NRC
      COMMON/DRIV2/NQS(10,4),NF
      COMMON/DRIV3/CQ(4),XLE(9)
      COMMON/STORE4/XMPH(10,5,4),VFXPH(10,5,4)
      COMMON /STORE5/NQC(5)
      COMMON/DRIV4/XKA(5)
      COMMON/STORE3/RX(10,11),RY(10,11),RZ(10,11),NRSM1(10)
      COMMON/GEO1/IBAR,ISEG,IGRA
      DATA LZ/2HZZ/
      INTEGER TITLE(15)
      EQUIVALENCE (RDIN(3),SIG(1))
    1 NQ=3
      NG=0
      NB=0
      CALL FILES                                                        5
    5 READ(INPT,1005) (TITLE(I),I=1,15)                                 6
      IF(TITLE(1).EQ.LZ)GO TO 105
      WRITE(IOUT,2009)                                                 15
      WRITE(IOUT,1005) (TITLE(I),I=1,15)                               15
      CALL INPUT                                                       23
      NF=RDIN(2)
      WRITE(IOUT,1002)(TITLE(I),I=1,15)                                25
      WRITE(IOUT,1003)                                                 32
      DO 11 I=1,NQ
      CQ(I)=EXP(.5*(SIG(I)*.23026)**2)                                 37
```

```
11 CONTINUE
   DO 60 I=1,NRC
   DO 15 J=1,NF
   XLF(J)=0.
15 CONTINUE
   DO 16 J=1,4
   XKA(J)=0.
16 CONTINUE
   DO 50 M=1,NR
   DO 20 IQ=1,NQ
   NOC1=NOS(M,IQ)
   IF(NOC1.EQ.0) GO TO 20
   CALL INTER(M,IQ)                                                  54
20 CONTINUE
   XR1(1)=RX(M,1)
    XR1(2)=RY(M,1)
   XR1(2)=RY(M,1)
   XR1(3)=RZ(M,1)
   NLIM=NRSM1(M)
   DO 40 N=2,NLIM
   XR2(1)=RX(M,N)
   XR2(2)=RY(M,N)

                                                              04/18/71
      MAIN        -  EFN   SOURCE STATEMENT  -  IFN(S)  -

   XR2(3)=RZ(M,N)
   CALL GEOMRY(XRC(I),YRC(I),ZRC(I),XR1,XR2,IERR,M)                  78
   IF(IERR.EQ.1)GO TO 65
   IF(IERR.EQ.2)GO TO 70
   IF(IERR.EQ.3)GO TO 66
   IF(IERR.EQ.4)GO TO 60
   CALL REPLCE(XR2,XR1)                                              95
40 CONTINUE
50 CONTINUE
   DO 55 J=1,NF
   XLF(J)=100.+10.*ALOG10(XLF(J))                                    103
55 CONTINUE
   IF(I.EQ.1) GO TO 155
   WRITE(IOUT,1003)                                                  103
```

```
 155 WRITE(IOUT,1004) I,XRC(I),YRC(I),ZRC(I)                          110
 156 IF(NF.EQ.1) GO TO 56
     WRITE(IOUT,2010)                                                 116
     WRITE(IOUT,2001)                                                 117
     WRITE(IOUT,2002) (XLF(II),II=2,NF)                               118
  56 XLA=100.+10.0*ALOG10(XKA(1))                                     124
     DO 59 J=2,4
     XKAP(J)=XKA(J)/XKA(1)**1                                         127
  59 CONTINUE
     SIGL=4.35*SQRT(ALOG(1.+XKAP(2)))                                 132   133
     XLNP=XLA+2.56*SIGL
     XL50=XLA-SIGL**2/8.7
     XL10=XL50+1.28*SIGL
     XL90=XL50-1.28*SIGL
     WRITE(IOUT,2003)                                                 134
     WRITE(IOUT,2004)XLA,XLNP,XL90,XL50,XL10                          135
  60 CONTINUE
     GO TO 100
  65 WRITE(IOUT,1006)I,M,N,IBAR,ISEG                                  140
     GO TO 100
  66 WRITE(IOUT,1009)I,M,N                                            142
     GO TO 100
  70 WRITE(IOUT,1008)I,M,N,IGRA                                       144
 100 GO TO 5
 105 GO TO 1
1002 FORMAT(//15A2)
1003 FORMAT(11HORECEIVER  10X,3HXRC9X,3HYRC9X,3HZRC)
1004 FORMAT(4X,I3,5X,3F12.1)
1005 FORMAT(15A2)
1006 FORMAT(44HOILLEGAL BARRIER INTERSECTS ROADWAY FOR SEC I2,2X,2HR I2
    1,3HRS I2,2X,2HR I2,2X,3HRS I2)
1008 FORMAT(49HOILLEGAL GROUND STRIP INTERSECTS ROADWAY FOR SEC I2,2X,2
    1HR I2,3HRS I2,2X,4HAGS I2)
1009 FORMAT(26HOTOO MANY REFLECTIONS,RCV ,I2,4H  R ,I2,4H  S ,I2)
2001 FORMAT( 14X,2H635X,3H1254X,3H2504X,3H5003X,4H10003X,4H20003X,
    X 4H40003X,4H8000)
2002 FORMAT(10X,8F7.1)
2003 FORMAT(/10X,5HLF(A),5X,3HLNP,5X,3HL90,5X,3HL50,5X,3HL10)
2004 FORMAT(8X,5F8.1//)
2009 FORMAT(1H1,15X,24HTRAFFIC NOISE PREDICTION//)
```

```
2010 FORMAT(/25X,22HOCTAVE BAND LEVELS (A))
     END
```

04/18/71

FILE - EFN SOURCE STATEMENT - IFN(S) -

```
      SUBROUTINE FILES
      COMMON/INOU/INPT,IOUT
      INPT=5
      IOUT=6
      RETURN
      END
```

04/18/71

INP - EFN SOURCE STATEMENT - IFN(S) -

```
      SUBROUTINE INPUT
      COMMON/DRIV2/NOS(10,4),NF
      INTEGER ALPHA(25),ALP1(9,8),BLNK
      DIMENSION VTEMP(5)
      COMMON/STORE1/BX(10,5),BY(10,5),BZ(10,5),IBLAST(10),NBSM1(10)
      COMMON/STORE2/XXG1(10,2),YYG1(10,2),ZZG1(10,2),BGS(10),IDIM(10)
      COMMON/STORE3/RX(10,11),RY(10,11),RZ(10,11),NRSM1(10)
      COMMON/STORE4/XMPH(10,5,4),VEXPH(10,5,4)
      COMMON/STORE5/NOC(5)
      COMMON/INOU/INPT,IOUT
      COMMON/BLK2/NO
      COMMON/INPT1/RDIN(15)
      COMMON/INPT2/NR,NB,NG
      COMMON/INPT3/XRC(15),YRC(15),ZRC(15),NRC
      COMMON/XIN/ XINS(9),NINE
      DIMENSION RDN(15)
      EQUIVALENCE (RDIN(1),XNIGHT)
      DATA   IA/2HA /
      DATA   IG/2HG /
      DATA   IR/2HR /
      DATA   IT/2HT /
```

```
      DATA LAST/2HL /
      DATA BLNK/6H      /
      DATA (ALP1(I,1),I=1,9)/6HRECEIV,6HER HEI,6HGHT AD,6HJUSTME,
     1     6HNT  ,6H    ,6H    ,6H    ,6H    /
      DATA (ALP1(I,2),I=1,9)/6HNUMBER,6H OF FR,6HEQUENC,6HY BAND,
     2     6HS   ,6H    ,6H    ,6H    ,6H    /
      DATA (ALP1(I,3),I=1,9)/6HSTANDA,6HRD DEV,6HIATION,6H FOR C,
     3     6HARS  ,6H    ,6H(TYPE ,6H1 VEHI,6HCLES) /
      DATA (ALP1(I,4),I=1,9)/6HSOURCE,6H HEIGH,6HT FOR ,6HCARS  ,
     4      6H    ,6H    ,6H    ,6H    ,6H    /
      DATA (ALP1(I,5),I=1,9)/6HSTANDA,6HRD DEV,6HIATION,6H FOR T,
     5      6HRUCKS ,6H    ,6H(TYPE ,6H2 VEHI,6HCLES) /
      DATA (ALP1(I,6),I=1,9)/6HSOURCE,6H HEIGH,6HT FOR ,6HTRUCKS,
     6      6H    ,6H    ,6H    ,6H    ,6H    /
      DATA (ALP1(I,7),I=1,9)/6HSTANDA,6HRD DEV,6HIATION,6H FOR N,
     7      6HEW VEH,6HICLES ,6H(TYPE ,6H3 VEHI,6HCLES) /
      DATA (ALP1(I,8),I=1,9)/6HSOURCE,6H HEIGH,6HT FOR ,6HNEW VE,
     8      6HHICLES,6H    ,6H    ,6H    ,6H    /
      NINF=0
      DO 1 J=1,5
    1 NOC(J)=0
    5 READ(INPT,1000)IGO,I1,I2                                    10
      GO TO(10,20,30,36,40,50),IGO
C GARBAGE DATA....PROGRAM INITIALIZATION PARAMETERS
   10 WRITE(IOUT,2000)                                            15
   11 READ(INPT,1001) VALUE,IDN,ILAST,(ALPHA(I),I=1,25)           16
      RDN(IDN)=VALUE
      IF(ALPHA(1).NE.BLNK) GO TO 13
      WRITE(IOUT,2016) VALUE,IDN,(ALP1(I,IDN),I=1,9)              28
      GO TO 14
   13 WRITE(IOUT,2001) VALUE,IDN,(ALPHA(I),I=1,25)                36
   14 IF(ILAST.NE.LAST) GO TO 11
      DO 12 J1=1,3
      J2=J1+2
```

04/18/71

INP - EFN SOURCE STATEMENT - IFN(S) -

```
      J3=2*J1+1
      J4=J1+6
```

```
      J5=2*J1+2
      RDIN(J2)=RDN(J3)
      RDIN(J4)=RDN(J5)
   12 CONTINUE
      RDIN(1)=RDN(1)
      RDIN(2)=RDN(2)
      IF(IDN.EQ.6) GO TO 5
      NINE=1
      READ(INPT,1006) (XINS(J1),J1=1,9)                                  63
      WRITE(IOUT,2015) (XINS(J1),J1=1,9)                                 70
      GO TO 5
C
C
C VEHICLE DATA
C
   20 NR=I1
      DO 28 J=1,NR
      NSEC=1
      DO 21 K=1,NQ
   21 NQS(J,K)=0
   22 READ(INPT,1002) VEH,XMH,ITY,ILAST                                  89
      NQS(J,ITY)=NQS(J,ITY)+1
      NQC1=NQS(J,ITY)
      VEXPH(J,NQC1,ITY)=VEH/XMH/5280.
      XMPH(J,NQC1,ITY)=XMH
      IF(ILAST.NE.LAST)GO TO 22
C ROADWAY DATA SECTIONS
   24 READ(INPT,1003)RX(J,NSEC),RY(J,NSEC),RZ(J,NSEC),ILAST             101
      IF(ILAST.EQ.LAST)GO TO 25
      NSEC=NSEC+1
      GO TO 24
   25 IF(NSEC-1.NE.0)GO TO 26
      WRITE(IOUT,2010)                                                  112
      CALL EXIT                                                         113
   26 NRSM1(J)=NSEC-1
   28 CONTINUE
      GO TO 5
C BARRIER DATA SECTIONS
   30 NB=I1
      IF(NB.EQ.0)GO TO 5
```

```
      DO 35 J=1,NB
      NSEC=1
   31 READ(INPT,1003)BX(J,NSEC),BY(J,NSEC),BZ(J,NSEC),IBLAST(J)          129
      IF(IBLAST(J).EQ.IA.OR.IBLAST(J).EQ.IB)GO TO 32
      NSEC=NSEC+1
      GO TO 31
   32 IF(NSEC-1.NE.0)GO TO 33
      WRITE(IOUT,2011)                                                    141
      CALL EXIT                                                           142
   33 NBSM1(J)=NSEC-1
   35 CONTINUE
      GO TO 5
C ABSORBING GROUND STRIPS
   36 NG=I1
```

04/18/71

```
         INP         -  EFN    SOURCE STATEMENT   -  IFN(S)  -

      IF(NG.EQ.0)GO TO 5
      DO 37 I=1,NG
      READ(INPT,1004)XXG1(I,1),YYG1(I,1),ZZG1(I,1),BGS(I)                 156
      READ(INPT,1003)XXG1(I,2),YYG1(I,2),ZZG1(I,2),IDUM(I)                161
      IF(IDUM(I).EQ.IG)IDUM(I)=1
      IF(IDUM(I).EQ.IT)IDUM(I)=2
   37 CONTINUE
      GO TO 5
C RECEIVER DATA
   40 NRC=I1
      DO 41 I=1,NRC
      READ(INPT,1003)XRC(I),YRC(I),ZRC(I),IRDUM                           182
      ZRC(I)=ZRC(I)+XNIGHT
   41 CONTINUE
      GO TO 5
C
   50 DO 65 J=1,NR
      WRITE(IOUT,2002)J                                                   196
      DO 55 K=1,NQ
      NOC1=NOS(J,K)
```

```
      IF(NOC1.EQ.0) GO TO 55
      DO 54 I=1,NOC1
   54 VTEMP(I)=VEXPH(J,I,K)*XMPH(J,I,K)*5280.
      WRITE(IOUT,2004)K,(I,VTEMP(I),XMPH(J,I,K),I=1,NOC1)
   55 CONTINUE
      WRITE(IOUT,2013)
      WRITE(IOUT,2005)RX(J,1),RY(J,1),RZ(J,1)
      NSEC=NRSM1(J)+1
      DO 60 I=2,NSEC
      WRITE(IOUT,2006)I,RX(J,I),RY(J,I),RZ(J,I)
   60 CONTINUE
   65 CONTINUE
      IF(NB.EQ.0)GO TO 80
      DO 75 J=1,NB
      WRITE(IOUT,2007)J,IBLAST(J)
      WRITE(IOUT,2005)BX(J,1),BY(J,1),BZ(J,1)
      NSEC=NBSM1(J)+1
      DO 70 I=2,NSEC
      WRITE(IOUT,2006)I,BX(J,I),BY(J,I),BZ(J,I)
   70 CONTINUE
   75 CONTINUE
   80 IF(NG.EQ.0) GO TO 90
      DO 85 I=1,NG
      IF(IDUM(I).EQ.1)IDM=IG
      IF(IDUM(I).EQ.2)IDM=IT
      WRITE(IOUT,2012)I,IDM,XXG1(I,1),YYG1(I,1),ZZG1(I,1),
     1 BGS(I),XXG1(I,2),YYG1(I,2),ZZG1(I,2)
   85 CONTINUE
   90 WRITE(IOUT,2008)
      DO 95 I=1,NRC
      WRITE(IOUT,2006)I,XRC(I),YRC(I),ZRC(I)
   95 CONTINUE
      RETURN
 1000 FORMAT(3I5)
 1001 FORMAT(E10.0,I5,4X,A1,10X,25A2)
 1002 FORMAT(2F10.0,I5,5X,A1)
```

```
1003 FORMAT(3E10.0,A1)
1004 FORMAT(4F10.0)
1006 FORMAT(9F5.0)
2000 FORMAT(34HOPROGRAM INITIALIZATION PARAMETERS)
2001 FORMAT(1X,E12.5,I10,5X,25A2)
2002 FORMAT(10HOROADWAY   ,I3)
2004 FORMAT(10HONUMBER OF13X,5HVEH/48X,3HMPH/5H TYPE,I2,4H VEH/(3X,I2
    1,15X,2F13.4))
2005 FORMAT(7HONUMBER,5X,1HX12X,1HY12X,1HZ/4X,1H1,2X,3F13.4)
2006 FORMAT(3X,I2,2X,3E13.4)
2007 FORMAT(10HOBARRIER  I3,2X,1H(A1,1H)10X,19HBARRIER COORD IN FT)
2008 FORMAT(9HORECEIVER14X,20HRECEIVER COORD IN FT/7H NUMBER5X,1HX12X,1
    1HY12X,1HZ)
2010 FORMAT(27HOINSUFFICIENT ROAD SECTIONS)
2011 FORMAT(30HOINSUFFICIENT BARRIER SECTIONS)
2012 FORMAT(18HOABSORBING STRIP  I3,2X,1H(A1,1H)//5H PT  7X,1HX12X,1HY1
    A2X,1HZ12X,5HWIDTH/4X,1H12X,4F13.4/4X,1H22X,3F13.4)
2013 FORMAT(22X,18HSOURCE COORD IN FT)
2015 FORMAT(5X,23HOPTIONAL NOISE SPECTRUM/5X,9F7.1)
2016 FORMAT(1X,F12.5,I10,5X,9A6)
     END

                                                              04/18/71
          INTE        -   FFN   SOURCE STATEMENT   -   IFN(S)   -

C   SUBROUTINES FOR TRAFFIC NOISE     12/13/71
      SUBROUTINE INTER(NR,IQ)
      DIMENSION TL1(2,9,4),TL(72)
      EQUIVALENCE(TL(1),TL1(1,1,1))
      COMMON/XIN/ XINS(9),NINF
      COMMON/BLK2/NQ
      COMMON/STORE4/XMPH(10,5,4),VEXPH(10,5,4)
      COMMON/DRIV2/NQS(10,4),NF
      COMMON/INTER1/XLREF(10,5,9,4)
      DATA TL(1), TL(2), TL(3), TL(4), TL(5), TL(6), TL(7), TL(8),
     1     TL(9),TL(10),TL(11),TL(12),TL(13),TL(14),TL(15),TL(16),
     2    TL(17),TL(18),TL(19),TL(20),TL(21),TL(22),TL(23),TL(24),
     3    TL(25),TL(26),TL(27),TL(28),TL(29),TL(30),TL(31),TL(32),
```

```
4      TL(33),TL(34),TL(35),TL(36)/
5   61.,75.,38.,48.,45.,57.,47.,62.,55.,66.,58.,70.,54.,72.,49.,63.,
6   42.,57.,
7   2*87.,2*60.,2*73.,2*78.,2*83.,2*82.,2*79.,2*74.,2*66./
      IF(NINF.LE.0) GO TO 1
      NINF=-1
      DO 13  J1=1,9
      J2=35+2*J1
      J3=36+2*J1
      J4=53+2*J1
      J5=54+2*J1
      TL(J2)=XINS(J1)
      TL(J3)=XINS(J1)
      TL(J4)=0.0
      TL(J5)=0.0
   13 CONTINUE
    1 NOO=NOS(NR,IO)
      CONS=ALOG10(70./30.)                                            25
      DO 10 IF=1,NF
      TEMP=TL1(2,IF,IO)-TL1(1,IF,IO)
      DO 10  I=1,NOO
      XLREF1=TL1(1,IF,IO)+TEMP*ALOG10(XMPH(NR,I,IO)/30.)/CONS         34
      XLREF(NR,I,IF,IO)=10.**((XLREF1-66.)/10.)                        34
   10 CONTINUE
      RETURN
      END
```

04/18/71

AMG   - EFN   SOURCE STATEMENT - IFN(S) -

```
      FUNCTION AMAG(X1V,X2V)
C FIND MAGNITUDE OF VECTOR
      DIMENSION X1V(3),X2V(3)
      AMAG=SQRT(DSQR(X1V,X2V))                                   2      2
      RETURN
      END
```

```
                                                                    04/18/71
      ANGL          -  EFN    SOURCE STATEMENT   -  IFN(S)  -

       FUNCTION ANGLE(X1V,X2V,X3V)
       DIMENSION X1V(3),X2V(3),X3V(3)
       D13=DSQR(X1V,X3V)                                                  2
       D23=DSQR(X2V,X3V)                                                  3
       ANGLE=1.5708
       IF(D13*D23.EQ.0.)RETURN
       D12=DSQR(X1V,X2V)                                                  5
       ANGLE=ACOS((D23+D13-D12)/(SQRT(D13*D23)*2.))                       7     8
       RETURN
       END

                                                                    04/18/71
      BARF          -  EFN    SOURCE STATEMENT   -  IFN(S)  -

       FUNCTION BARFAC(KF,DELP)
C FIND BARRIER FACTOR
       IF(DELP.EQ.-0.2)GO TO 3
       IF(DELP.GE.12.5)GO TO 4
       IP=KF
       IF(KF.EQ.1)IP=5
       A=DELP*(2.**IP)/5.7                                                11
       IF(A.GE.78.5)GO TO 4
       IF(A.GT.0..AND.A.LT.78.5)GO TO 5
       IF(A.EQ.0.)GO TO 6
       IF(A.GT.-1.25.AND.A.LT.0.)GO TO 7
     3 BARFAC=1.0
       RETURN
     4 BARFAC=4.0E-3
       RETURN
     5 BARFAC=(TANH(SQRT(A))**2)/A/3.16                                   27    28
       RETURN
     6 BARFAC=.316
       RETURN
```

```
      7 A1=ABS(A)
        BARFAC=(TAN(SQRT(A1)))**2/A1/3.16                                    31      32
        RETURN
        END

                                                                      04/18/71
          BLOC        -  EFN    SOURCE STATEMENT  -  IFN(S)  -

        SUBROUTINE BLOCN(X1V,X2V,X3V,X4V,X5V,X6V,LOC)
C FIND RELATIVE LOCATION OF BARRIER
        DIMENSION X1V(2),X2V(2),X3V(2),X4V(2),X5V(2),X6V(2),XAV(2),XBV(2)
        LOC=KCUT(X1V,X3V,X4V,X5V)                                             2
        LOC=LOC+KCUT(X2V,X3V,X4V,X5V)*2                                       4
        IF(LOC.EQ.3)RETURN
        CALL TRI(X1V,X2V,X3V,X4V,XAV,KTRI)                                    8
        IF(LOC.EQ.0)GO TO 4
        IF(KTRI.EQ.1)GO TO 6
      2 CALL INTCPT(X1V,X2V,X3V,X5V,XBV)                                      17
        IF(LOC.EQ.4)GO TO 5
      3 X6V(1)=XBV(1)
        X6V(2)=XBV(2)
        RETURN
      4 IF(KTRI.EQ.0)RETURN
        LOC=4
        GO TO 2
      5 IF(KPOS(X1V,XAV,XBV).EQ.1)GO TO 3                                     29
      6 X6V(1)=XAV(1)
        X5V(2)=XAV(2)
        RETURN
        END

                                                                      04/18/71
          DEL1        -  EFN    SOURCE STATEMENT  -  IFN(S)  -

        FUNCTION DEL(X1V,X2V,X3V,X4V,HDIFF,DN1)
```

```
C FIND PATH LENGTH DIFFERENCE
      DIMENSION X1V(3),X2V(3),X3V(3),X4V(3),XAV(3),XBV(3)
      CALL NRPT(X3V,X4V,X1V,XAV,DISTA)                                2
      CALL NRPT(X3V,X4V,X2V,XBV,DISTB)                                4
      DISTC=DSQR(XBV,XAV)                                             5
      DEL=SQRT((DISTA+DISTB)**2+DISTC)-DN1                            7
      IF(HDIFF.GT.0.)DEL=-DEL
      RETURN
      END
```

04/18/71

DSQR1 - EFN SOURCE STATEMENT - IFN(S) -

```
      FUNCTION DSQR(X1V,X2V)
      DIMENSION  X1V(3),X2V(3)
      DSQR=0.0
      DO 10 I=1,3
   10 DSQR=(X1V(I)-X2V(I))**2+DSQR
      RETURN
      END
```

04/18/71

ENPT - EFN SOURCE STATEMENT - IFN(S) -

```
      SUBROUTINE ENDPT(X1V,X2V,X3V,X4V,X5V,X6V,KTRIG,IERR)
C FIND NEW ENDPOINT
      DIMENSION X1V(3),X2V(3),X3V(3),X4V(3),X5V(3),X6V(3),XDUM(3)
      IERR=0
      KTRIG=0
      ITRIG=1
      CALL REPLCE(X1V,XDUM)                                           5
    1 CALL BLOCN(XDUM,X2V,X3V,X4V,X5V,X6V,LOC)                        7
      IF(LOC.EQ.0) RETURN
      IF(LOC.NE.3) GO TO 2
      CALL REPLCE(X2V,X6V)                                           14
      GO TO 4
    2 X6V(3)=ZCOR(X1V,X2V,X6V)                                       17
```

```
      IF(ITRIG.EQ.0) GO TO 5
      IF(AMAG(X1V,X6V).GT.0.51) GO TO 5                             21
      ITRIG=0
      DELTA=-0.51
      CALL MOVE(XDUM,XDUM,X2V,DELTA,IERR)                           26
      GO TO 1
    5 DELTA=-0.5
      IF(LOC.EQ.1) GO TO 3
      CALL MOVE(X6V,X2V,X1V,DELTA,IERR)                             32
      RETURN
    3 CALL MOVE(X6V,X6V,X1V,DELTA,IERR)                             35
    4 KTRIG=1
      RETURN
      END
```

```
                                                          04/18/71
      HGT          -  EFN    SOURCE STATEMENT  -  IFN(S)  -

      FUNCTION HEIGHT(X1V,X2V,X3V,X4V)
C FIND HEIGHT DIFFERENCE
      DIMENSION X1V(3),X2V(3),X3V(3),X4V(3),XI(3)
      CALL INTCPT(X1V,X2V,X3V,X4V,XI)                               2
      HEIGHT=ZCOR(X1V,X2V,XI)-ZCOR(X3V,X4V,XI)                      4    5
      RETURN
      END
```

```
                                                          04/18/71
      IEP          -  EFN    SOURCE STATEMENT  -  IFN(S)  -

      FUNCTION IEPS (X1V, X2V, X3V, X4V, DEL1)
C     CHECK ON PATH LENGTH DIFFERENCE
      DIMENSION X1V(3),X2V(3),X3V(3),X4V(3)
      IEPS =0
      DIST = AMAG (X1V,X2V)                                         3
      HDIFF = HEIGHT (X1V,X2V,X3V,X4V)                              4
      DEL2 = DEL (X1V, X2V, X3V, X4V, HDIFF, DIST)                  5
      DELM= (DEL1+DEL2)/2.
      IF((ABS(DEL2-DEL1) - 0.1-DELM/50.* (1.+DELM)) .GT.0.) IEPS=1
```

```
      RETURN
      END
```

04/18/71

IMAG - EFN SOURCE STATEMENT - IFN(S) -

```
      SUBROUTINE IMAGE(X1V,X2V,X3V,X4V)
C FIND IMAGE POINT
      DIMENSION X1V(3),X2V(3),X3V(3),X4V(3)
      AX=X2V(1)-X1V(1)
      AY=X2V(2)-X1V(2)
      AXY=AX**2+AY**2
      RATIO=0.
      IF(AXY.EQ.0.)GO TO 10
      RATIO=(((X3V(2)-X2V(2))*AX-(X3V(1)-X2V(1))*AY)*2.0)/AXY
   10 X4V(1)=X3V(1)+AY*RATIO
      X4V(2)=X3V(2)-AX*RATIO
      X4V(3)=X3V(3)
      RETURN
      END
```

04/18/71

INTCT - EFN SOURCE STATEMENT - IFN(S) -

```
      SUBROUTINE INTCPT(X1V,X2V,X3V,X4V,X5V)
C FIND INTERCEPT OF TWO LINES IN A PLANE
      DIMENSION X1V(2),X2V(2),X3V(2),X4V(2),X5V(2)
      AX=X2V(1)-X1V(1)
      AY=X2V(2)-X1V(2)
      BX=X4V(1)-X3V(1)
      BY=X4V(2)-X3V(2)
      C1=AY*X2V(1)-AX*X2V(2)
      C2=BY*X4V(1)-BX*X4V(2)
      D=AX*BY-AY*BX
      IF(D**2.LT.1.E-6)GO TO 2
      X5V(1)=(AX*C2-BX*C1)/D
      X5V(2)=(AY*C2-BY*C1)/D
```

```
      RETURN
    2 D=SQRT(AX**2+AY**2)                                                    5
      X5V(1)=X1V(1)+(AX/D)*1.E+15
      X5V(2)=X1V(2)+(AY/D)*1.E+15
      RETURN
      END
```

```
                                                                    04/19/71
         KCT          -   EFN    SOURCE STATEMENT   -   IFN(S)   -
```

```
      FUNCTION KCUT(X1V,X2V,X3V,X4V)
C DETERMINE IF TWO LINE SEGMENTS CROSS
      DIMENSION X1V(2),X2V(2),X3V(2),X4V(2),X5V(2)
      KCUT=0
      CALL INTCPT(X1V,X2V,X3V,X4V,X5V)                                       3
      IF(KPOS(X1V,X2V,X5V).NE.1)RETURN                                       5
      IF(KPOS(X3V,X4V,X5V).EQ.1)KCUT=1                                       8
      RETURN
      END
```

```
                                                                    04/18/71
         KPS          -   EFN    SOURCE STATEMENT   -   IFN(S)   -
```

```
      FUNCTION KPOS(X1V,X2V,X3V)
C FIND POSITION OF POINT ON LINE
      DIMENSION X1V(2),X2V(2),X3V(2)
      KPOS=1
      IF(((X3V(1)-X1V(1))*(X3V(1)-X2V(1))+(X3V(2)-X1V(2))*(X3V(2)-X2V(2)
     1)).GT.0.)KPOS=0
      RETURN
      END
```

```
                                                                    04/18/71
         MDP          -   EFN    SOURCE STATEMENT   -   IFN(S)   -
```

```
      SUBROUTINE MIDP (X1V, X2V, X3V)
```

```
C      FIND CENTER POINT
       DIMENSION X1V(3), X2V(3), X3V(3)
       DO 10 I=1,3
    10 X3V(I) = (X1V(I)+X2V(I))/2.
       RETURN
       END
```

```
          NRP          -  EFN   SOURCE STATEMENT  -  IFN(S)  -                04/18/71
```

```
       SUBROUTINE NRPT(X1V,X2V,X3V,X4V,DIST)
C FIND NEAREST POINT TO LINE
       DIMENSION X1V(3),X2V(3),X3V(3),X4V(3),AV(3),BV(3)
       EQUIVALENCE (AV(1),AX),(AV(2),AY),(AV(3),AZ),(BV(1),BX),(BV(2),BY
     1),(BV(3),BZ)
       DO 5 I=1,3
       AV(I)=X2V(I)-X1V(I)
       BV(I)=X3V(I)-X1V(I)
     5 CONTINUE
       RATIO=0.
       TEMP=DSQR(X2V,X1V)                                                  14
       IF(TEMP.NE.0.)RATIO=(AX*BX+AY*BY+AZ*BZ)/TEMP
       DO 10 I=1,3
       X4V(I)=X1V(I)+RATIO*AV(I)
    10 CONTINUE
       DIST=AMAG(X4V,X3V)                                                  26
       IF(DIST.EQ.0.)DIST=1.
       RETURN
       END
```

```
          NRR1         -  EFN   SOURCE STATEMENT  -  IFN(S)  -                04/18/71
```

```
       SUBROUTINE NR1(X1V,X2V,X3V,X4V,DIST,X5V,DN1)
C FIND NEAREST POINT TO LINE SEGMENT
       DIMENSION X1V(3),X2V(3),X3V(3),X4V(3),X5V(3)
       IF(KPOS(X4V,X2V,X1V).EQ.1)GO TO 2                                   2
       IF(KPOS(X1V,X4V,X2V).EQ.1)GO TO 4                                   6
```

```
      CALL REPLCE(X4V,X5V)                                              10
      DN1=DIST
      RETURN
    2 CALL REPLCE(X1V,X5V)                                              13
      GO TO 6
    4 CALL REPLCE(X2V,X5V)                                              16
    6 DN1=AMAG(X5V,X3V)                                                 18
      RETURN
      END
```

```
                                                              04/18/71
       REPLC        -  EFN    SOURCE STATEMENT  -  IFN(S)  -
```

```
      SUBROUTINE REPLCE(X1V,X2V)
      DIMENSION X1V(3),X2V(3)
      X2V(1)=X1V(1)
      X2V(2)=X1V(2)
      X2V(3)=X1V(3)
      RETURN
      END
```

```
                                                              04/18/71
       SCTN         -  EFN    SOURCE STATEMENT  -  IFN(S)  -
```

```
      SUBROUTINE SECTN(X1V,X2V,X3V,X4V,X5V,X6V,X7V)
C FIND EFFECTIVE BARRIER SECTION
      DIMENSION X1V(3),X2V(3),X3V(3),X4V(3),X5V(3),X6V(3),X7V(3)
      CALL INTCPT(X1V,X3V,X4V,X5V,X6V)                                   2
      X6V(3)=ZCOR(X4V,X5V,X6V)                                           4
      CALL INTCPT(X2V,X3V,X4V,X5V,X7V)                                   5
      X7V(3)=ZCOR(X4V,X5V,X7V)                                           7
      RETURN
      END
```

```
                                                              04/18/71
       TRII         -  EFN    SOURCE STATEMENT  -  IFN(S)  -
```

```
      DIMENSION B1(3,2,10),RI(3,2,10),RB1(3,2,10),TA1(3,2,10)
      DIMENSION KBCODE(10),KNUMB(10),KRNUMB(10),KRDNUM(10)
      DIMENSION KGCODE(10),BGT(10),IKIN(10)
      DIMENSION
     D DELP0(4),DELP1(4),DELP2(4),FB(9,4),DELR(4,10),FG(9),HGA(2),
     X XB1(3),XB2(3),XDB1(3),XDB2(3),XDB3(3),XDB4(3),
     X XRB1(3),XRB2(3),XRB3(3),XRB4(3),XRC(3),XRCI(3),
     X XR1(3),XR2(3),XR1I(3),XR2I(3),XR10(3),XR20(3),
     X XK(3),XKI(3),XJ(3),XNPT(3),XNPTI(3),XNPTJ(3),
     X XN1(3),XN2(3),XN1I(3),XIMG(3,10),XR2D(3),XR2G(3),
     X XLA(5,4),
     Z ZS(4)
      COMMON/INOU/INPT,IOUT
      COMMON/BLK2/NQ
      COMMON/INPT1/RDIN(15)
      COMMON/INPT2/NR,NB,NG
      COMMON/DRIV2/NQS(10,4),NF
      COMMON/DRIV3/CQ(4),XLE(9)
      COMMON/STORE4/XMPH(10,5,4),VEXPH(10,5,4)
      COMMON/STORE1/BX(10,5),BY(10,5),BZ(10,5),IBLAST(10),NBSMI(10)
      COMMON/STORE2/XXG1(10,2),YYG1(10,2),ZZG1(10,2),BGS(10),IDUM(10)
      COMMON/DRIV4/XKA(5)
      COMMON/GEO1/IBAR,ISEG,IGRA
      COMMON/INTER1/XLREF(10,5,9,4)
      EQUIVALENCE (RDIN(7),ZS(1))
      DATA   IR/2HR /
      DATA HGA(1),HGA(2)/10.,30./
      XRC(1)=XR
      XRC(2)=YR
      XRC(3)=ZR
      IERR=0
      CALL NRPT(XR10,XR20,XRC,XNPT,DIST)                                    3
      ANG1=ANGLE(XR10,XNPT,XRC)
C     ICODE=1
C     WRITE(IOUT,1000)ICODE,XR10,XR20,XRC,XNPT
C PRELIMINARY SELECTION OF BARRIERS                                         5
      NDIFF=0
      IF(NB.EQ.0) GO TO 16
      KBAR=0
      DO 15 IBAR=1,NB
```

```
      KAR=IBLAST(IBAR)
      XB1(1)=BX(IBAR,1)
      XB1(2)=BY(IBAR,1)
      XB1(3)=BZ(IBAR,1)
      NLIM=NBSMI(IBAR)+1
      DO 15 ISEG=2,NLIM
      XB2(1)=BX(IBAR,ISEG)
      XB2(2)=BY(IBAR,ISEG)
      XB2(3)=BZ(IBAR,ISEG)
      KBAR=KBAR+1
C     ICODE=2
C     WRITE(IOUT,1001)ICODE,KBAR,XB1,XB2
```

```
                                                                  04/18/71
      GEOM        -  EFN   SOURCE STATEMENT  -  IFN(S)  -

      IF(KCUT(XR10,XR20,XB1,XB2).NE.1)GO TO 10                        27
      IERR=1
      RETURN
   10 CALL BLOCN(XR10,XR20,XRC,XB1,XB2,XK,LOC)                        33
      IF(LOC.EQ.0)GO TO 11
      NDIFF=NDIFF+1
      KNUMB(NDIFF)=KBAR
      KBCODE(NDIFF)=LOC
      CALL REPLCE(XB1,B1(1,1,NDIFF))                                  41
      CALL REPLCE(XB2,B1(1,2,NDIFF))                                  44
   11 CALL REPLCE(XB2,XB1)                                            47
   15 CONTINUE
C PRELIMINARY SELECTION OF STRIPS
   16 KGA=0
      IF(NG.EQ.0)GO TO 20
      DO 18 IGRA=1,NG
      XG1(1)=XXG1(IGRA,1)
      XG1(2)=YYG1(IGRA,1)
      XG1(3)=ZZG1(IGRA,1)
      XG2(1)=XXG1(IGRA,2)
      XG2(2)=YYG1(IGRA,2)
      XG2(3)=ZZG1(IGRA,2)
C     ICODE=3
C     WRITE(IOUT,1001)ICODE,IGRA,XG1,XG2
```

```
      IF(KCUT(XR10,XR20,XG1,XG2).NE.1)GO TO 17                        66
      IERR=2
      RETURN
   17 CALL BLDCN(XR10,XR20,XRC,XG1,XG2,XK,LOC)                        72
      IF(LOC.EQ.0)GO TO 18
      KGA=KGA+1
      KGCODE(KGA)=LOC
      CALL REPLCE(XG1,TA1(1,1,KGA))                                   79
      CALL REPLCE(XG2,TA1(1,2,KGA))                                   82
      BGT(KGA)=BGS(IGRA)
      IKIN(KGA)=IDUM(IGRA)
   18 CONTINUE
C DIFFRACTION OF DIRECT RAY
   20 CALL REPLCE(XR10,XR1)                                           93
   25 CALL REPLCE(XR20,XR2)                                           95
      DO 30 IQ=1,NQ
      DELPO(IQ)=-.2
   30 CONTINUE
C     ICODE=4
C     WRITE(IOUT,1000)ICODE,XR1,XR2
      IF(NDIFF.EQ.0)GO TO 71
      ITRIG=0
      DO 70 KDIFF=1,NDIFF
      KBAR=KNUMB(KDIFF)
      KCD=KBCODE(KDIFF)
      CALL REPLCE(B1(1,1,KDIFF),XB1)                                  116
      CALL REPLCE(B1(1,2,KDIFF),XB2)
C     ICODE=5
C     WRITE(IOUT,1001)ICODE,KBAR,XR1,XR2,XB1,XB2                      119
      IF(KCD.EQ.3)GO TO 40
      CALL ENDPT(XR1,XR2,XRC,XB1,XB2,XK,KTRIG,IERR)                   124
      IF(IERR.EQ.4) RETURN

                                                              04/18/71
          GEOM          -  EFN   SOURCE STATEMENT  -  IFN(S)  -

      IF(KTRIG.EQ.0)GO TO 70
      GO TO 45
   40 CALL REPLCE(XR2,XK)                                             133
   45 MODD=0
```

```
C       ICODE=6
C       WRITE(IOUT,1001)ICODE,KCD,XR2,XK
        DO 60 IQ=1,NQ
        CALL NR1(XR1,XK,XRC,XNPT,DIST,XN1,DN1)                    139
        ZN10=XN1(3)
        XN1(3)=ZN10+ZS(IQ)
        HDIFF=HEIGHT(XN1,XRC,XB1,XB2)                             141
        IF(IQ.NE.1)GO TO 50
        IF(HDIFF.GT.20.)GO TO 70
     50 DELP=DEL(XN1,XRC,XB1,XB2,HDIFF,DN1)                       149
        IF(DELP.GT.DELPO(IQ))GO TO 53
     52 IF(MODD.EQ.1)GO TO 65
        GO TO 70
     53 IF (IQ.NE.1) GO TO 59
        DR1 = AMAG(XRC,XR1)                                       161
        IF (ABS(DR1-DN1).LT.1.) GO TO 54
        HDIFA = HEIGHT (XR1,XRC,XB1,XB2)                          165
        DELPA = DEL (XR1,XRC,XB1,XB2,HDIFA,DR1)                   166
        DELM= (DELPA + DELP)/2.
C       ICODE=107
C       WRITE(IOUT,1000)ICODE,DELP,DELPA
        IF((ABS(DELPA-DELP)-0.1-DELM/50.*(1.+DELM)).LE.0.) GO TO 55
        CALL MIDP (XR1,XN1,XK)
C       ICODE=7
C       WRITE(IOUT,1000)ICODE,XR1,XN1,XK                          170
        DELP = DELPA
     54 IF (IEPS(XK,XRC,XB1,XB2,DELP).EQ.0) GO TO 58              173
        CALL MIDP (XR1,XK,XK)
C       ICODE=8
C       WRITE(IOUT,1000)ICODE,XR1,XK                              176
        GO TO 54
     55 DRK= AMAG (XRC,XK)                                        179
        IF (ABS(DRK-DN1).LT.1.) GO TO 58
     56 IF(IEPS (XK,XRC,XB1,XB2,DELP).EQ.0) GO TO 58              184
        CALL MIDP(XN1,XK,XK)                                      187
        GO TO 56
     58 IF (DELP.LE.DELPO(1)) GO TO 52
     59 DELPO (IQ) = DELP
C       ICODE=10
C       WRITE(IOUT,1000)ICODE,DELPO(1),DELPO(2)
```

```
      MODD=1
   60 CONTINUE
      IF(DELPO(2).LT.12.5)GO TO 65
      ITRIG=1
      IF(KCD.NE.3.OR.KCD.NE.2)GO TO 65
      CALL REPLCE(XR20,XR2)                                         203
      GO TO 66
   65 CALL REPLCE(XK,XR2)                                           206
   66 IF(ITRIG.EQ.1)GO TO 71
   70 CONTINUE
C     ICODE=11
C     WRITE(IOUT,1000)ICODE,XR2
```

04/18/71

GEOM - EFN SOURCE STATEMENT - IFN(S) -

```
   71 ANG=ANGLE(XR1,XR2,XRC)                                        214
      DELP=DELPO(1)
      FB(1,1)=BARFAC(1,DELP)
C     ICODE=12
C     WRITE(IOUT,1002)ICODE,FB(1,1)                                 215
      CALL REPLCE(XR2,XR2D)
C GROUND ABSORPTION                                                 216
   72 DO 73 KF=1,NF
      FG(KF)=1.
   73 CONTINUE
C     ICODE=13
C     WRITE(IOUT,1000)ICODE,XR1,XR2,XR2D
      IF(DELPO(1).GT.-0.2)GO TO 78
      IF(KGA.EQ.0)GO TO 78
      DO 77 IGRA=1,KGA
      LOC=KGCODE(IGRA)
      CALL REPLCE(TA1(1,1,IGRA),XG1)                                237
      CALL REPLCE(TA1(1,2,IGRA),XG2)                                240
      BG=BGT(IGRA)
      IKIND=IKIN(IGRA)
C     ICODE=14
C     WRITE(IOUT,1001)ICODE,LOC,XG1,XG2
      IF(LOC.EQ.3)GO TO 74
      CALL ENDPT(XR1,XR2,XRC,XG1,XG2,XK,KTRIG,IERR)                 243
```

```
      IF(IERR.EQ.4) RETURN
      IF(KTRIG.EQ.0)GO TO 77
      GO TO 75
   74 CALL REPLCE(XR2,XK)                                          257
   75 CALL NR1(XR1,XK,XRC,XNPT,DIST,XN1,DN1)                       259
      HDIFF=HEIGHT(XN1,XRC,XG1,XG2)                                261
      IF(HDIFF.GT.HGA(IKIND))GO TO 77
      CALL SECTN(XR1,XK,XRC,XG1,XG2,XG3,XG4)                       266
      DL=1.57/(1./BG+1./AMAG(XG3,XG4))                             268
      DO 76 IK=1,NF
      PP=IK
      IF(IK.EQ.1)PP=5.
      IF(IKIND.EQ.1)A=(.0016*PP-0.0028)*DL
      IF(IKIND.EQ.2)A=2.**(PP/3.)/1310.*DL                         276
      IF(A.GT.3.)A=3.
      FG(IK)=FG(IK)/10.**A                                         280
      IF(FG(IK).LT.1.E-3)FG(IK)=1.E-3
      CALL REPLCE(XK,XR2)                                          287
   76 CONTINUE
C     ICODE=15
C     WRITE(IOUT,1000)ICODE,FG(1),FG(9)
   77 CONTINUE
   78 CALL REPLCE(XR2,XR2G)
C PRELIMINARY SELECTION OF REFLECTORS                              294
      NREF=0
      IF(NB.EQ.0)GO TO 91
      NRFDF=0
      KBAR=0
      DO 85 IBAR=1,NB
      KAR=IBLAST(IBAR)
      XRB1(1)=BX(IBAR,1)
      XRB1(2)=BY(IBAR,1)
```

GEOM - EFN SOURCE STATEMENT - IFN(S) - 04/18/71

```
      XRB1(3)=BZ(IBAR,1)
      NLIM=NBSM1(IBAR)+1
      DO 85 ISEG=2,NLIM
      XRB2(1)=BX(IBAR,ISEG)
```

```
      XRB2(2)=BY(IBAR,ISEG)
      XRB2(3)=BZ(IBAR,ISEG)
      KBAR=KBAR+1
C     ICODE=16
C     WRITE(IOUT,1001)ICODE,KBAR,XR1,XR2,XRB1,XRB2
      CALL IMAGE(XRB1,XRB2,XRC,XRCI)                                    318
      CALL NRPT(XR1,XR2,XRCI,XNPTI,DISTI)
C     ICODE=17
C     WRITE(IOUT,1000)ICODE,XRCI,XNPTI                                  320
      ANGIMG=ANGLE(XRB1,XRB2,XRCI)                                      322
      FCTR=(DIST*ANGIMG)/DISTI/ANG/FB(1,1)
      IF(FCTR.LT.0.1)GO TO 80
      NRFDF=NRFDF+1
      KRDNUM(NRFDF)=KBAR
      CALL REPLCE(XRB1,RB1(1,1,NRFDF))                                  328
      CALL REPLCE(XRB2,RB1(1,2,NRFDF))
C     ICODE=18
C     WRITE(IOUT,1001)ICODE,NRFDF,XRB1,XRB2                             331
      IF(KAR.NE.IR)GO TO 80
      CALL BLOCN(XR1,XR2,XRCI,XRB1,XRB2,XK,LOC)                         337
      IF(LOC.EQ.0)GO TO 80
      NREF=NREF+1
      KRNUMB(NREF)=KBAR
      CALL REPLCE(XRB1,R1(1,1,NREF))                                    344
      CALL REPLCE(XRB2,R1(1,2,NREF))
C     ICODE=19
C     WRITE(IOUT,1001)ICODE,NREF,XRB1,XRB2                              347
   80 CALL REPLCE(XRB2,XRB1)                                            350
   85 CONTINUE
C BEGIN REFLECTOR PROBLEM
   91 IDXR=0
      IF(NREF.EQ.0)GO TO 180
      DO 175 KREF=1,NREF
      KBAR1=KRNUMB(KREF)
      CALL REPLCE(R1(1,1,KREF),XRB1)                                    365
      CALL REPLCE(R1(1,2,KREF),XRB2)
C     ICODE=20
C     WRITE(IOUT,1001)ICODE,KBAR1,XRB1,XRB2,XR1,XR2                     369
      CALL IMAGE(XRB1,XRB2,XRC,XRCI)                                    371
      CALL ENDPT(XR1,XR2,XRCI,XRB1,XRB2,XK,KTRIG,IERR)                  373
```

```
      IF(IERR.EQ.4) RETURN
C     ICODE=21
C     WRITE(IOUT,1001)ICODE,KTRIG,XRCI,XK
      IF(KTRIG.EQ.0)GO TO 175
      CALL NRPT(XR1,XR2,XRCI,XNPTI,DISTI)                                   390
      CALL NR1(XR1,XK,XRCI,XNPTI,DISTI,XN1I,DN1I)
C     ICODE=22
C     WRITE(IOUT,1000)ICODE,XNPTI,XN1I                                      382
      XN1I(3)=XN1I(3)+ZS(2)
      HDIFF=HEIGHT(XN1I,XRCI,XRB1,XRB2)                                     384
      IF(HDIFF.GT.-2.0)GO TO 175
      CALL REPLCE(XK,XR2)                                                   388

                                                                   04/18/71
        GEOM        -  EFN   SOURCE STATEMENT  -  IFN(S)  -

      CALL SECTN(XR1,XR2,XRCI,XRB1,XRB2,XRB3,XRB4)
C     ICODE=23
C     WRITE(IOUT,1000)ICODE,XR1,XR2,XRB3,XRB4                               390
      MDIFF=0
      DO 95 IQ=1,NQ
      DELP1(IQ)=-0.2
      DELP2(IQ)=-0.2
   95 CONTINUE
C DIFFRACTION BEFORE REFLECTION
      IF(NREDF.EQ.0)GO TO 115
      DO 110 KREDF=1,NREDF
      KBAR2=KRDNUM(KREDF)
      CALL REPLCE(RB1(1,1,KREDF),XDB1)                                      407
      CALL REPLCE(RB1(1,2,KREDF),XDB2)
C     ICODE=24
C     WRITE(IOUT,1001)ICODE,KBAR2,XDB1,XDB2                                 410
      IF(KBAR2.EQ.KBAR1)GO TO 110
      CALL BLOCN(XRB3,XRB4,XRCI,XDB1,XDB2,XK,LOC)                           415
      IF(LOC.NE.0)GO TO 110
      CALL ENDPT(XR1,XR2,XRCI,XDB1,XDB2,XK,KTRIG,IERR)                      420
      IF(IERR.EQ.4) RETURN
C     ICODE=25
C     WRITE(IOUT,1001)ICODE,KTRIG,XK
      IF(KTRIG.EQ.0)GO TO 110
```

```
      CALL NR1(XR1,XK,XRCI,XNPTI,DISTI,XN1I,DN1I)
C     ICODE=26
C     WRITE(IOUT,1000)ICODE,XN1I                                427
      ZN10=XN1I(3)
      XN1I(3)=ZN10+ZS(1)
      HDIFF=HEIGHT(XN1I,XRCI,XDB1,XDB2)                          429
      IF(HDIFF.GT.20.0)GO TO 110
      CALL SECTN(XR1,XK,XRCI,XDB1,XDB2,XDB3,XDB4)
C     ICODE=27
C     WRITE(IOUT,1000)ICODE,XDB3,XDB4                           433
      CALL NRPT(XDB3,XDB4,XRCI,XNPTJ,DISTJ)                     435
      CALL NR1(XDB3,XDB4,XRCI,XNPTJ,DISTJ,XN2,DN2)
C     ICODE=127
C     WRITE(IOUT,1000)ICODE,XNPTJ,XN2,DISTJ,DN2                 437
      HDIFF=HEIGHT(XN2,XRCI,XRB1,XRB2)
C     ICODE=227
C     WRITE(IOUT,1002)ICODE,HDIFF                               439
      IF(HDIFF.GT.-2.0)GO TO 170
      DO 105 II=1,NQ
      IQ=NQ+1-II
      XN1I(3)=ZN10+ZS(IQ)
      HDIFF=HEIGHT(XN1I,XRCI,XDB1,XDB2)                          447
      DELP=DEL(XN1I,XRCI,XDB1,XDB2,HDIFF,DN1I)
C     ICODE=327
C     WRITE(IOUT,1002)ICODE,DELP                                449
      IF(DELP.GE.12.5.AND.IQ.EQ.2)GO TO 170
      IF(DELP.GT.DELP1(IQ))GO TO 100
      IF(IQ.EQ.1)GO TO 110
      GO TO 105
  100 MDIFF=1
      DELP1(IQ)=DELP
  105 CONTINUE
```

GEOM - EFN SOURCE STATEMENT - IFN(S) - 04/13/71

```
C     ICODE=28
C     WRITE(IOUT,1000)ICODE,DELP1(1),DELP1(2)
      CALL REPLCE(XK,XR2)
  110 CONTINUE                                                  455
```

```
C       ICODE=29
C       WRITE(IOUT,1000)ICODE,XR2
C DIFFRACTION AFTER REFLECTION
        CALL  IMAGE(XRB1,XRB2,XNPTI,XNPTJ)                              468
        CALL  IMAGE(XRB1,XRB2,XR1,XR1I)                                 470
  115 CALL  IMAGE(XRB1,XRB2,XR2,XR2I)
C       ICODE=30
C       WRITE(IOUT,1000)ICODE,XR1I,XR2I,XNPTI                           472
        IF(NREDF.EQ.0)GO TO 145
        DO 140 KREDF=1,NREDF
        KBAR2=KRDNUM(KREDF)
        CALL  REPLCE(RB1(1,1,KREDF),XDB1)                               482
        CALL  REPLCE(RB1(1,2,KREDF),XDB2)
C       ICODE=31
C       WRITE(IOUT,1001)ICODE,KBAR2,XDB1,XDB2                           485
        IF(KBAR2.EQ.KBAR1)GO TO 140
        CALL  INTCPT(XRB1,XRB2,XRC,XR2I,XRB4)                           490
        CALL  BLOCN(XRB3,XRB4,XRC,XDB1,XDB2,XJ,LOC)                     492
        IF(LOC.EQ.0)GO TO 140
        IF(LOC.EQ.3)GO TO 120
        CALL  INTCPT(XR1I,XR2I,XRC,XJ,XKI)                              500
        XKI(3)=ZCOR(XR1I,XR2I,XKI)                                      502
        DELTA=-0.5
        CALL  MOVE(XKI,XKI,XR1I,DELTA,IERR)                             503
        IF(IERR.EQ.4) RETURN
        IF(LOC.NE.1)GO TO 135
        GO TO 125
  120 CALL  REPLCE(XR2I,XKI)                                            512
  125 CALL  INTCPT(XR1I,XRC,XDB1,XDB2,XDB3)
C       ICODE=32
C       WRITE(IOUT,1001)ICODE,LOC,XJ,XKI,XDB3                           514
        XD33(3)=ZCOR(XDB1,XDB2,XDB3)                                    516
        HDIFF=HEIGHT(XR1I,XDB3,XRB1,XRB2)                               517
        IF(HDIFF.GT.-2.0)GO TO 165
        CALL  NR1(XR1I,XKI,XRC,XNPTJ,DISTI,XN1I,DN1I)                   521
        ZN10=XN1I(3)
        DO 130 II=1,NQ
        IQ=NQ+1-II
        XN1I(3)=ZN10+ZS(IQ)
        HDIFF=HEIGHT(XN1I,XRC,XDB1,XDB2)                                527
```

```
      IF(HDIFF.GT.20.0.AND.IQ.EQ.2)GO TO 140
      DELP=DEL(XNII,XRC,XDB1,XDB2,HDIFF,DNII)
      IF(MDIFF.EQ.1.AND.DELP.GT.-0.2)GO TO 165
      IF(DELP.LE.DELP2(IQ))GO TO 140
      DELP2(IQ)=DELP
  130 CONTINUE
C     ICODE=33
C     WRITE(IOUT,1000)ICODE,DELP2(1),DELP2(2)
      IF(DELP2(2).GE.12.5)GO TO 165
  135 CALL REPLCE(XKI,XR2I)
  140 CONTINUE
  145 CALL REPLCE(XR2I,XKI)
```

04/19/71

GEOM - EFN SOURCE STATEMENT - IFN(S) -

```
      IDXR=IDXR+1
C     ICODE=34
C     WRITE(IOUT,1001)ICODE,IDXR,XR2I
      IF(IDXR.LT.11)GO TO 150
      IERR=3
      RETURN
  150 DO 155 IQ=1,NQ
      DELB(IQ,IDXR)=AMAX1(DELP1(IQ),DELP2(IQ))
  155 CONTINUE
      DO 160 I=1,3
      XIMG(I,IDXR)=XRCI(I)
  160 CONTINUE
      GO TO 165
  165 CALL IMAGE(XRB1,XRB2,XKI,XK)
  170 CALL REPLCE(XK,XR2)
C     ICODE=35
C     WRITE(IOUT,1000)ICODE,XR2
  175 CONTINUE
C     ICODE=36
C     WRITE(IOUT,1000)ICODE,XR1,XR2
C BEGIN BARRIER FACTOR COMPUTATION
  180 NIMG=IDXR
      ANG=ANGLE(XR1,XR2,XRC)
      ADSI=ANG/DIST
```

```
      IF(KPOS(XNPT,XR2,XR1).EQ.1)GO TO 190                             592
      ANG2=ANG1-ANG
      GO TO 195
  190 ANG2=ANG1+ANG
C CONTRIBUTION FROM DIRECT RAY
  195 DO 205 IQ=1,NQ
      IF(NOS(MR,IQ).EQ.0)GO TO 205
      DELP=DELPO(IQ)
      DO 200 KF=1,NF
      FB(KF,IQ)=BARFAC(KF,DELP)                                        599
  200 CONTINUE
C     ICODE=37
C     WRITE(IOUT,1002)ICODE,FB(1,IQ),FB(9,IQ)
  205 CONTINUE
C CONTRIBUTION FROM REFLECTIONS
      IF(NIMG.EQ.0)GO TO 230
      DO 225 KIMG=1,NIMG
      DO 210 I=1,3
      XRCI(I)=XIMG(I,KIMG)
  210 CONTINUE
      ANGI=ANGLE(XR1,XR2,XRCI)                                         618
      CALL NRPT(XR1,XR2,XRCI,XNPTI,DISTI)                              619
      RATIO=(ANGI/DISTI)/ADST
      DO 220 IQ=1,NQ
      IF(NOS(MR,IQ).EQ.0)GO TO 220
      DELP=DELR(IQ,KIMG)
      DO 215 KF=1,NF
      FB(KF,IQ)=FB(KF,IQ)+BARFAC(KF,DELP)*RATIO                        630
  215 CONTINUE
  220 CONTINUE
C     ICODE=38
C     WRITE(IOUT,1002)ICODE,FB(1,1),FB(1,2),FB(9,1),FB(9,2)

          GEOM        -  EFN   SOURCE STATEMENT  -  IFN(S)  -        04/19/71

  225 CONTINUE
C COMPUTE MEAN ENERGY LEVEL
  230 CALL NR1(XR1,XR2,XRC,XNPT,DIST,XN1,DN1)                          641
      DO 232 IQ=1,NQ
```

```
      NQQ=NOS(MR,IQ)
      IF(NQQ.EQ.0)GO TO 232
      DO 231 I=1,NQQ
      XLA(I,IQ)=0
  231 CONTINUE
  232 CONTINUE
      DO 242 IF=1,NF
      IP=IF
      IF(IF.EQ.1)IP=5
      A=10.**(-1.E-8*4.**IP*DN1)
      T1=0.
      DO 240 IQ=1,NQ
      IF(NOS(MR,IQ).EQ.0)GO TO 240
      NQQ=NOS(MR,IQ)
      T2=0.
      DO 235 I=1,NQQ
      T3=XLREF(MR,I,IF,IQ)*A*FR(IF,IQ)*FG(IF)
      T2=T2+VEXPH(MR,I,IQ)*T3
      IF(NF.GT.6.AND.IF.EQ.1)GO TO 235
      IF(NF.LE.6.AND.IF.NE.1)GO TO 235
      XLA(I,IQ)=XLA(I,IQ)+T3
  235 CONTINUE
      T1=T1+CO(IQ)*T2
  240 CONTINUE
      XLE(IF)=XLE(IF)+ADST*T1
  242 CONTINUE
      SA1=SIN(ANG1)
      SA2=SIN(ANG2)
      CA1=COS(ANG1)
      CA2=COS(ANG2)
      CP2EV=ANG
      DO 246 N=1,4
      T1=0.
      DO 245 IQ=1,NQ
      IF(NOS(MR,IQ).EQ.0)GO TO 245
      NQQ=NOS(MR,IQ)
      T2=0.
      DO 244 I=1,NQQ
      T2=T2+VEXPH(MR,I,IQ)*XLA(I,IQ)**N
  244 CONTINUE
```

```
      T1=T1+CO(IO)**(N**2)*T2                                                 723
  245 CONTINUE
      IT3=2*N-1
      T3=IT3
      XKA(N)=XKA(N)+T1/DIST**IT3*CPREV                                        727
      AEN=N
      CPREV=0.5/AEN*(SA2*CA2**IT3-SA1*CA1**IT3+T3*CPREV)                      730   731
  246 CONTINUE
C     ICODE=39
C     WRITE(IOUT,1002)ICODE,XLE(1),XLE(9),CAP2
      ANG1=ABS(ANG2)
C     ICODE=40

                                                              04/18/71
          GEOM        -   EFN   SOURCE STATEMENT  -  IFN(S)  -

C     WRITE(IOUT,1000)ICODE,XR2,XR2G,XR2D,XR20
      IF(DSQR(XR2,XR20).LT.1.)RETURN                                          735
      DELTA=1.
      CALL MOVE(XR2,XR1,XR10,DELTA,IERR)                                      739
      IF(IERR.EQ.4) RETURN
C     ICODE=41
C     WRITE(IOUT,1000)ICODE,XR1
      IF(DSQR(XR2,XR2G).LT.1.0)GO TO 247                                      742
      CALL REPLCE(XR2G,XR2)                                                   744
      GO TO 91
  247 IF(DSQR(XR2,XR2D).LT.1.0)GO TO 25                                       749
      CALL REPLCE(XR2D,XR2)                                                   752
      GO TO 72
C1000 FORMAT(6HOCODE I3,6F9.2/6F9.2)
C1001 FORMAT(6HOCODE I3,I4,6F9.2/6F9.2)
C1002 FORMAT(6HOCODE I3,6E12.2)
      END
```