FAA-EM-80-8-IV
DOT-TSC- FAA-82-4- IV

Airport Landside
Volume IV: Appendix A ALSIM AUXILIARY and MAIN Programs

L. McCabe
M. Gorstein

Transportation Systems Center
Cambridge MA 02142

June 1982
Final Report

This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.

U.S. Department of Transportation
Federal Aviation Administration

Office of Systems Engineering and Management
Washington DC 20591

NOTICE

This document is disseminated under the sponsorship of the Department of Transportation in the interest of information exchange. The United States Government assumes no liability for its contents or use thereof.

NOTICE

The United States Government does not endorse products or manufacturers. Trade or manufacturers' names appear herein solely because they are considered essential to the object of this report.

Technical Report Documentation Page

1. Report No.
FAA-EM-80-8-IV

2. Government Accession No.
A117 602

3. Recipient's Catalog No.

4. Title and Subtitle
AIRPORT LANDSIDE
Volume IV: Appendix A ALSIM
Auxiliary and Main Programs

5. Report Date
June 1982

6. Performing Organization Code
DTS-53

7. Author(s)
L McCabe and M. Gorstein

8. Performing Organization Report No.
DOT-TSC-FAA-82-4-IV

9. Performing Organization Name and Address
U.S. Department of Transportation
Research and Special Programs Administration
Transportation Systems Center
Cambridge MA 02142

10. Work Unit No. (TRAIS)
R113/FA032

11. Contract or Grant No.

12. Sponsoring Agency Name and Address
U.S. Department of Transportation
Federal Aviation Administration
Office of Systems Engineering Management
Washington DC 20591

13. Type of Report and Period Covered
Final Report
Jan. 1978 - Sep. 1980

14. Sponsoring Agency Code
[illegible]

15. Supplementary Notes

16. Abstract

This Appendix describes the Program Logic of the Airport Landside Simulation Model (ALSIM) AUXILIARY and MAIN Programs. Both programs are written in GPSS-V. The AUXILIARY program is operated prior to the MAIN Program to create GPSS transactions representing Enplaning Passenger groups from the input flight schedule. The transactions are written on a JOBTAPE file for subsequent use by the MAIN program. The MAIN Program creates greeter and deplaning passenger transactions and enacts the movements of all passengers and visitors through the landisde.

Details of program logic and flow charts at the GPSS block level are provided. A listing of both programs is included.

Other volumes of the Airport Landside report are: Volume I: Planning Guide; Volume II: ALSIM Description and User's Guide; Volume III: ALSIM Calibration and Validation; and, Volume V: Appendix B ALSIM Subroutines.

17. Key Words
GPSS-V
Simulation

18. Distribution Statement
DOCUMENT IS AVAILABLE TO THE U.S. PUBLIC THROUGH THE NATIONAL TECHNICAL INFORMATION SERVICE, SPRINGFIELD, VIRGINIA 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	286	

SUMMARY

This document provides a detailed description of the MAIN and AUXILIARY programs of the Airport Landside Simulation Model (ALSIM). Both programs are written in GPSS-V. The MAIN program simulates the movement of enplaning and deplaning passengers through the landside. Statistics are obtainable from this program for every simulated facility. These include queue length averages and maxima, queueing time averages and distributions, and service characteristics of the facility.

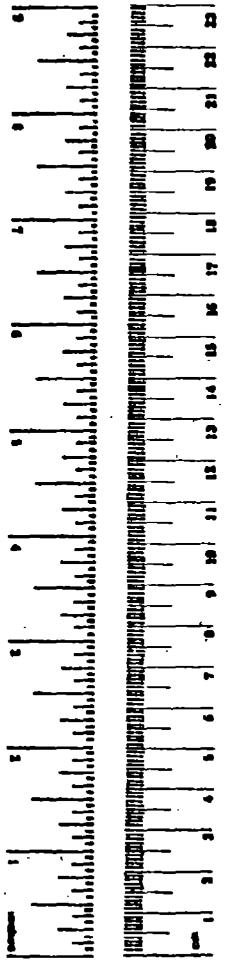
The AUXILIARY program is operated prior to the MAIN program to generate and store transactions representing enplaning passengers. These transactions are fed into the MAIN program as the simulation clock advances to the time of arrival on the landside by the simulated passenger.

This document describes the general structure, input variables and matrices used by each program. The use of passenger transaction parameters and a detailed description of program logic are also contained in this document. Flowcharts and listings for each program are exhibited.

METRIC CONVERSION FACTORS

Approximate Conversions to Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
in	inches	2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
		AREA		
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
		MASS (weight)		
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t
		VOLUME		
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C



Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
		AREA		
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
		MASS (weight)		
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	
		VOLUME		
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	35	cubic feet	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
		TEMPERATURE (exact)		
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F

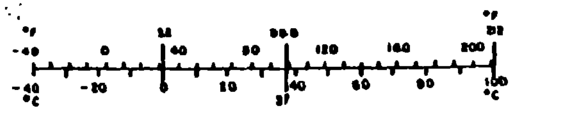


TABLE OF CONTENTS

VOLUME IV

APPENDIX A-1

ALSIM DOCUMENTATION-GPSS-V AUXILIARY AND MAIN PROGRAMS

L. McCABE AND R. WALKER

AUXILIARY PROGRAM

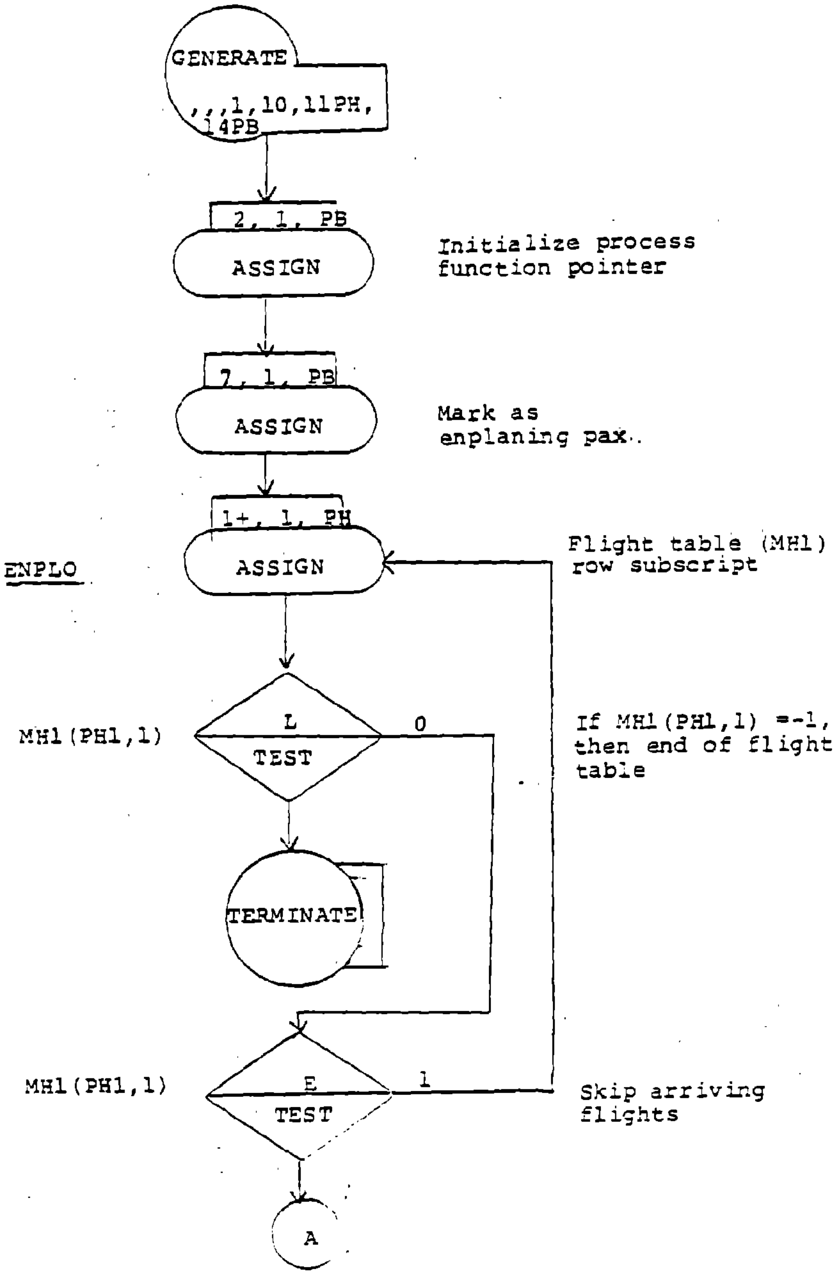
General Description

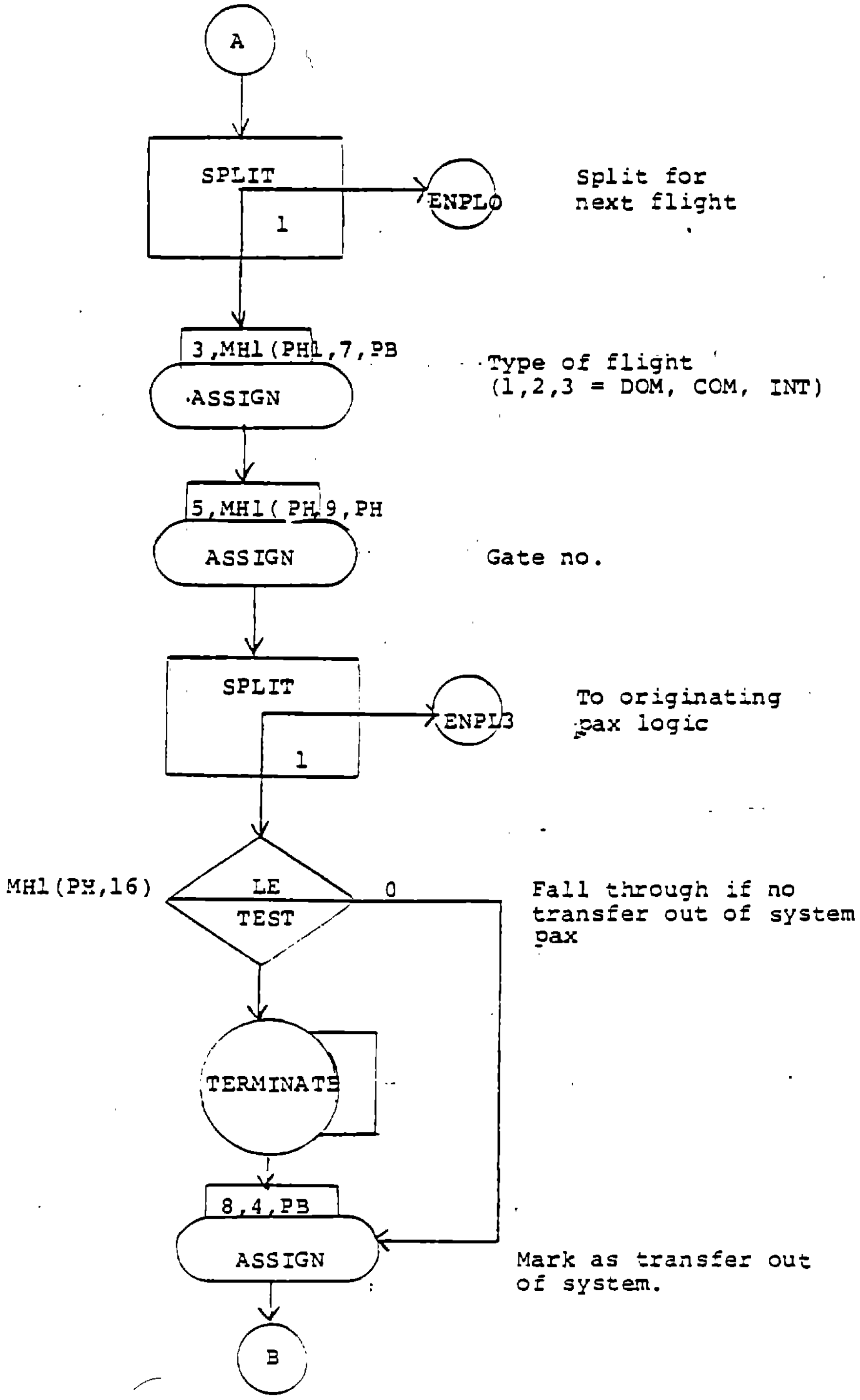
The Auxiliary Program shown in Figure A-1 generates transactions representing enplaning passenger groups. These are stored on a JOBTAPE and utilized by the main simulation program when required. This program assigns attributes to the transaction parameters for representing ground transportation modal choice, ticketed or non-ticketed status, passenger group size and number of accompanying well wishers. The program contains three sections, (1) A Program Definition Statements Section, (2) Enplaning Passenger Creation Section and, (3) A Program Control Section. The Program Definition Statements Section defines matrix sizes, cumulative distribution functions and simulation variables. The Enplaning Passenger Creation Section generates enplaning passenger transactions and assigns well wisher numbers and transaction attributes. The program control section initiates linkage programs between the GPSS Auxiliary Program and the FORTRAN supporting subprogram FORTM. It also creates dummy transactions to begin and end the JOBTAPE.

Program Definition Statement Section

The section begins with an R MULT statement to specify starting numbers for random number generator multipliers. The subrouting FORTM is kept resident in core during the operation of this simulation by a LOAD block. Halfword save value 1 is defined as CLKXH by an EQU statement.

Auxiliary Program





Split for next flight

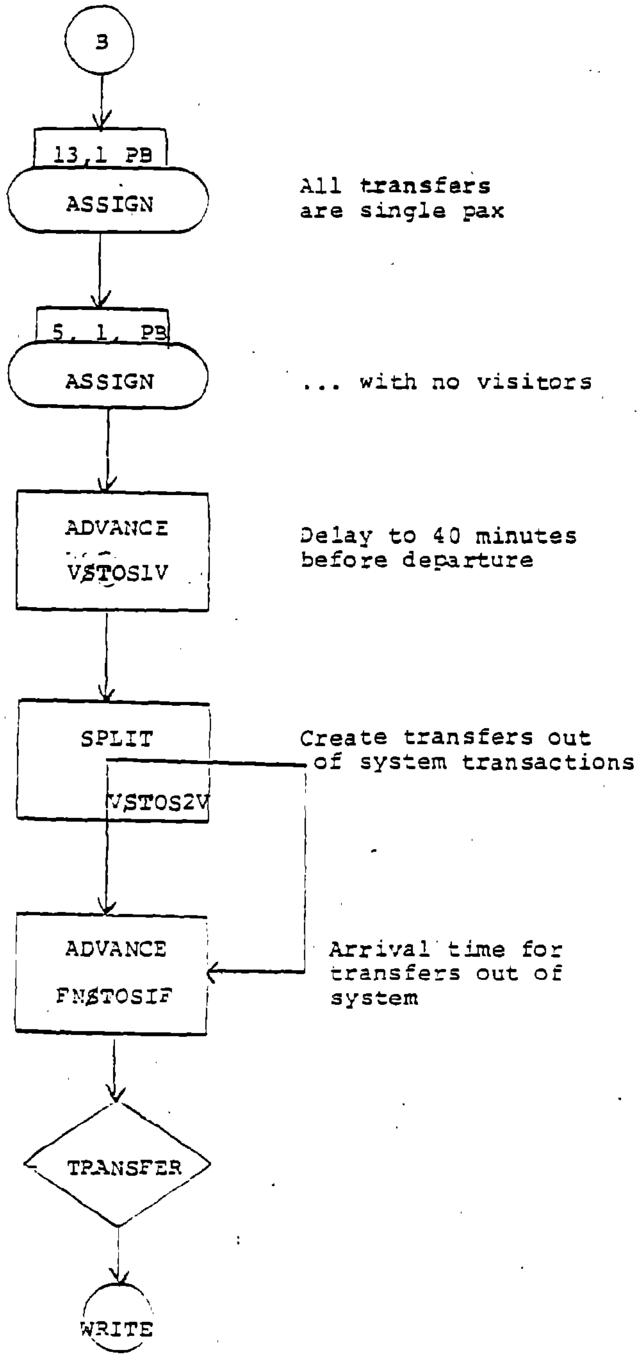
Type of flight (1,2,3 = DOM, COM, INT)

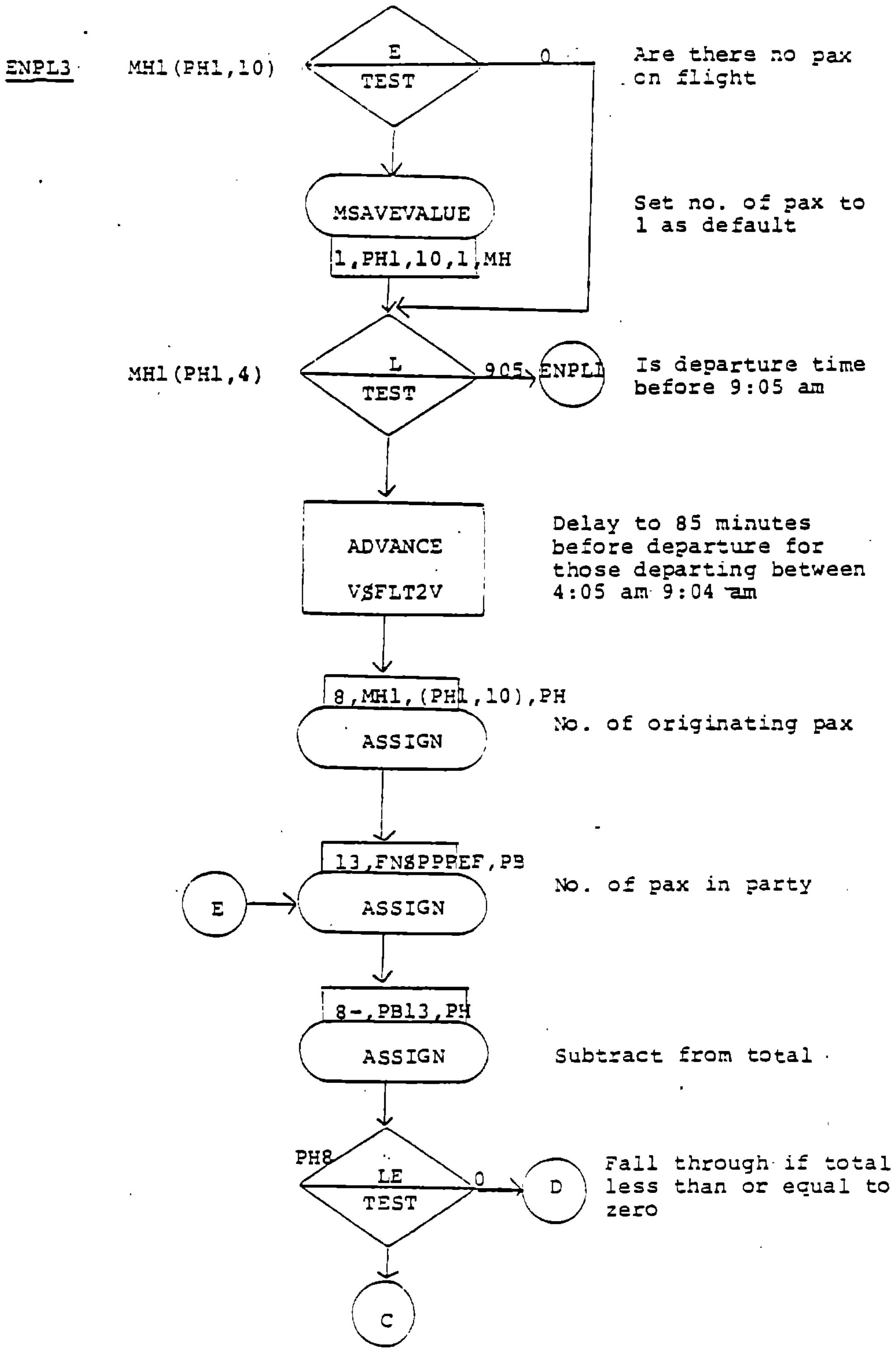
Gate no.

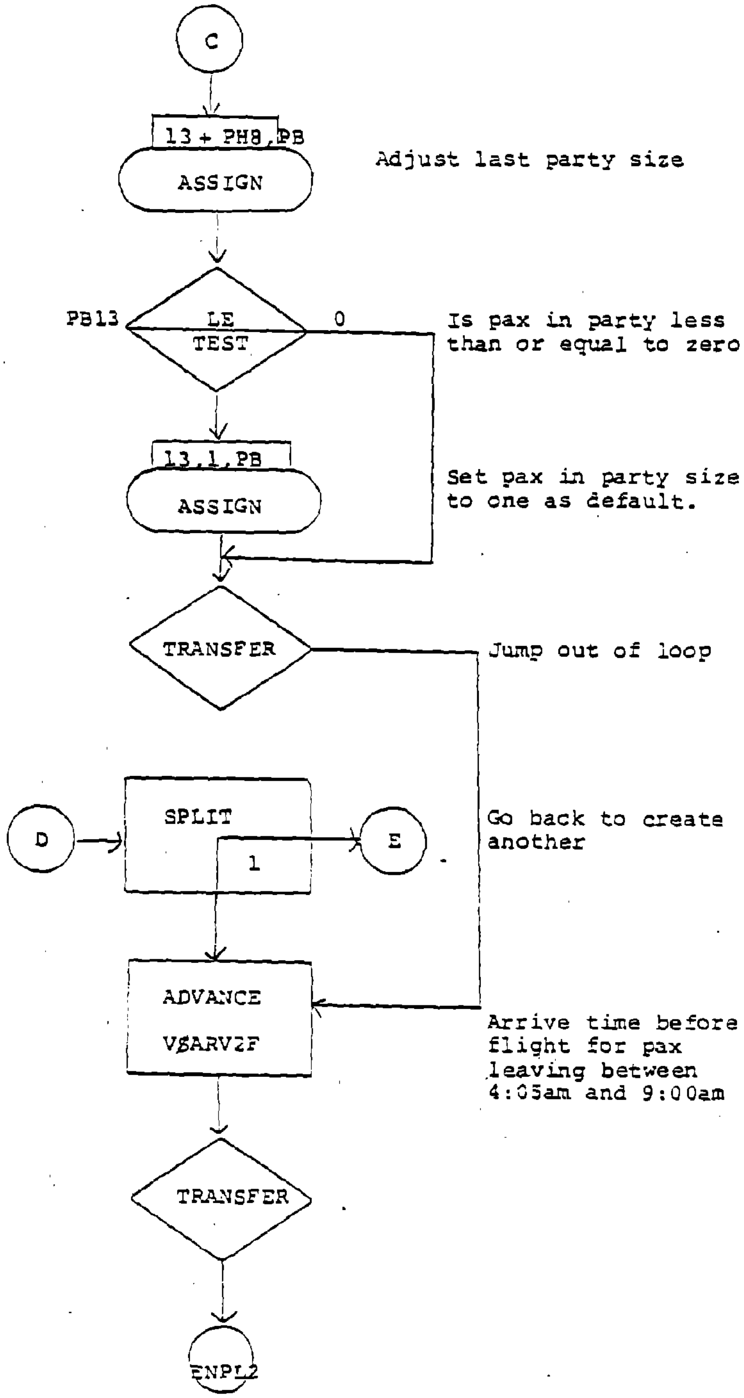
To originating pax logic

Fall through if no transfer out of system pax

Mark as transfer out of system.


B
13,1 PB
ASSIGN
All transfers are single pax
5, 1, PB
ASSIGN
... with no visitors
ADVANCE
V$TOS1V
Delay to 40 minutes before departure
SPLIT
V$TOS2V
Create transfers out of system transactions
ADVANCE
FN$TOS1F
Arrival time for transfers out of system
TRANSFER
WRITE


ENPL3
MH1(PH1,10)
E
TEST
0
Are there no pax on flight
MSAVEVALUE
1,PH1,10,1,MH
Set no. of pax to 1 as default
MH1(PH1,4)
L
TEST
905
ENPL1
Is departure time before 9:05 am
ADVANCE
V$FLT2V
Delay to 85 minutes before departure for those departing between 4:05 am 9:04 am
8,MH1,(PH1,10),PH
ASSIGN
No. of originating pax
13,FN$PPPEF,PB
E
ASSIGN
No. of pax in party
8-,PB13,PH
ASSIGN
Subtract from total
PH8
LE
TEST
0
D
Fall through if total less than or equal to zero
C



Adjust last party size

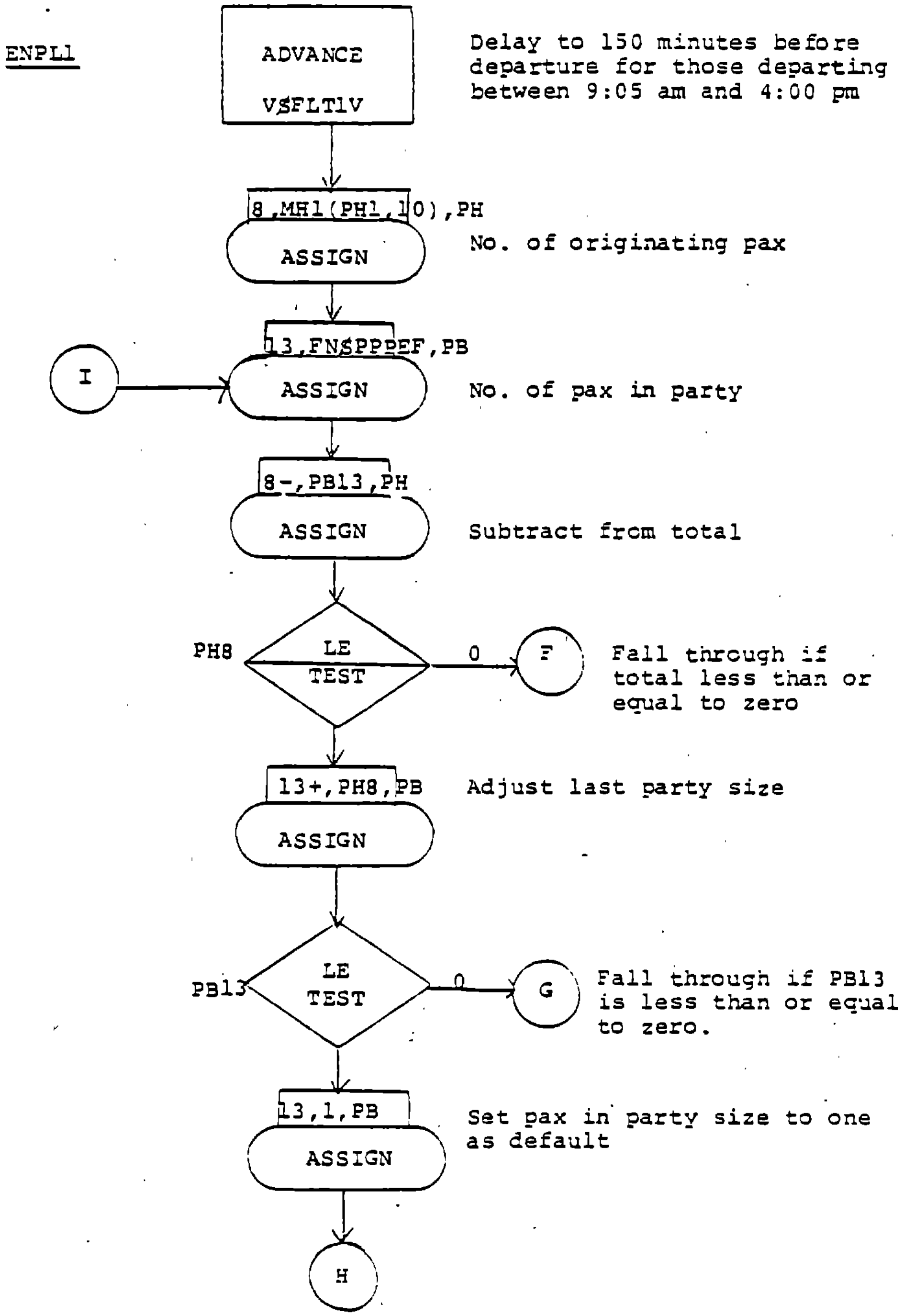
Is pax in party less than or equal to zero

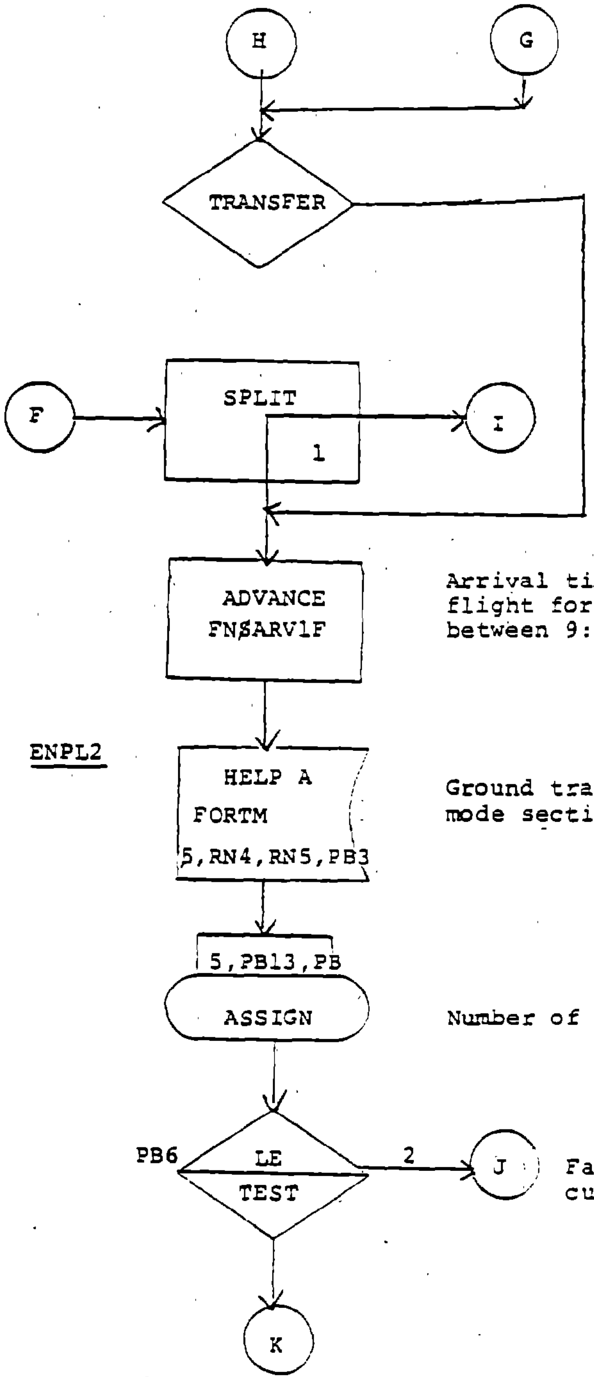
Set pax in party size to one as default.

Jump out of loop

Go back to create another

Arrive time before flight for pax leaving between 4:05am and 9:00am


ENPL1
ADVANCE
V$FLT1V
Delay to 150 minutes before departure for those departing between 9:05 am and 4:00 pm
8,MH1(PH1,10),PH
ASSIGN
No. of originating pax
13,FN$PPPEF,PB
I
ASSIGN
No. of pax in party
8-,PB13,PH
ASSIGN
Subtract from total
PH8
LE
TEST
0
F
Fall through if total less than or equal to zero
13+,PH8,PB
Adjust last party size
ASSIGN
PB13
LE
TEST
0
G
Fall through if PB13 is less than or equal to zero.
13,1,PB
Set pax in party size to one as default
ASSIGN
H



Jump out of loop

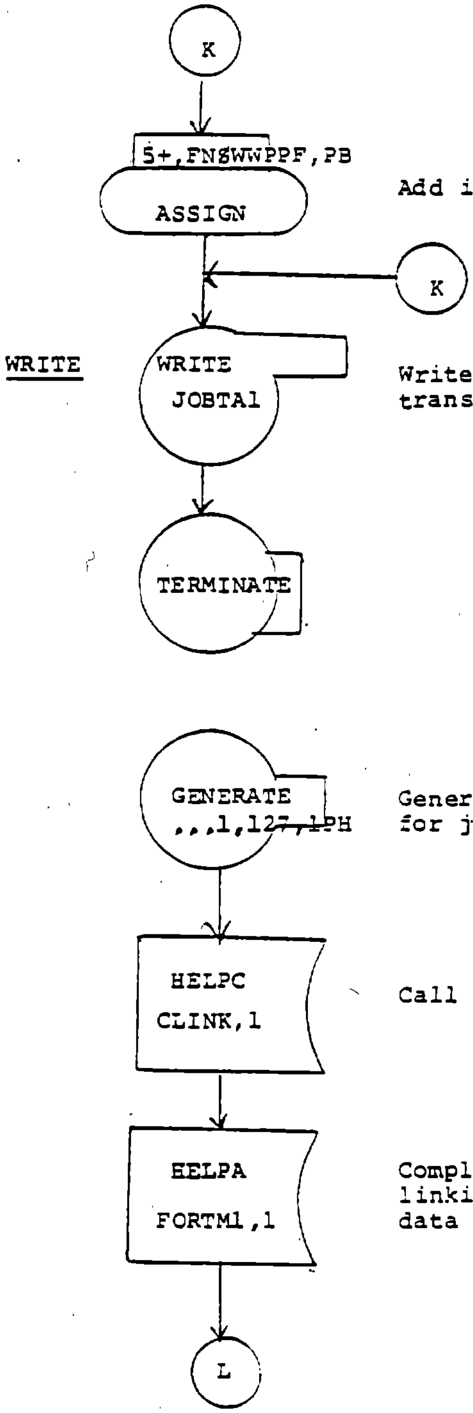
Go back to create another

Arrival time prior to flight for pax departing between 9:05 am and 4:00 pm

Ground transportation mode section

Number of party

Fall through if curb or park



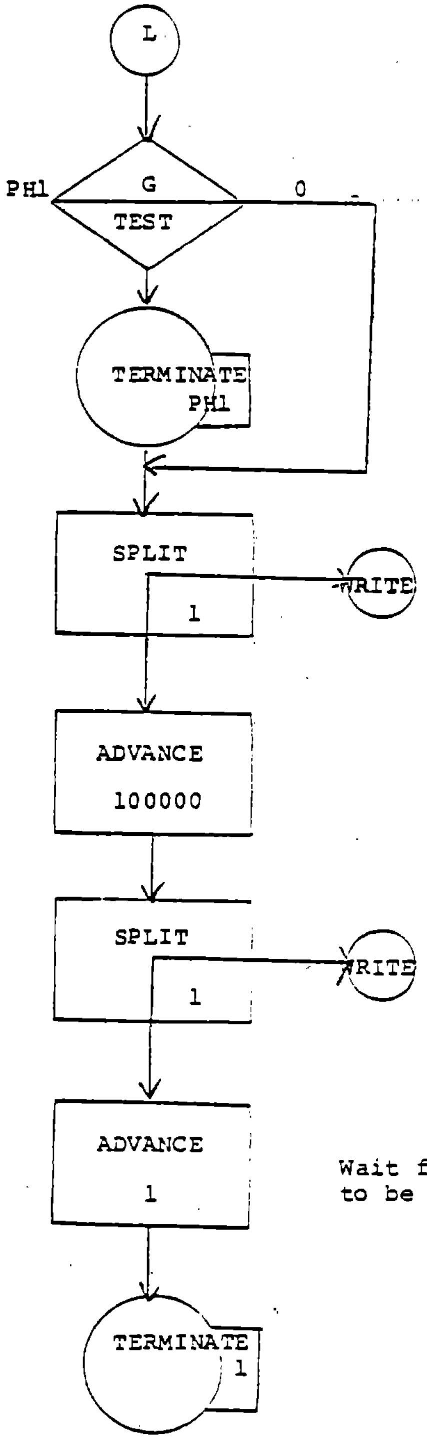
Add in well-wishers

Write enplaning transactions on jobtape

Generate dummy transaction for jobtape

Call linking subroutines

Completes FORTM-GPSS linking, and reads input data



Test for error
in FORTM

Used to set up
interarrival times

Prevent jobtape
end message

Wait for above transaction
to be written

Data required for Auxiliary Program execution is contained in three matrices and five functions. Matrices used are: halfword matrix MH1, the flight schedule; halfword matrix MH4, the percent of enplaned passengers preticketed; and, floating point matrix ML2, ground transaction modal choice percentages. Information contained in these matrices is provided by input data read and placed in the matrix elements by FORTM.

The numbers of columns of MH1, MH4, and ML2 are represented by the entity symbols CMH01, CMH04 and CML02 respectively. These are assigned numerical values by SYN statements. These values will be transferred to FORTM by the use of a mnemonic link function which appears in the program control section and the assembler subprogram MNLINK called by FORTM

Matrix type and dimensions of MH1, MH4, and ML2 are specified by matrix definition statements. The column numbers used must agree with preceding SYN statements. Tables 1 and 2 describe matrix contents.

Functions used by this program are: ARV1F and ARV2F, arrival times at the airport landside of originating passengers PPEF, passengers per enplaning group; WWPPF, well wishers per enplaning group and TOS1F, the arrival times of transfer passengers at a concourse.

Because of a time dependence in arrival rates, two cumulative distribution functions are used to specify the

Auxiliary Program

Table 1

Contents of Flight Table Matrix MH1

Column	Usage	FORTRAN Data Item
1	Designates arriving or departing flight. Indicates completion of aircraft bag unloading for arrival flights. 0 = Arriving Flight 1 = Departing Flight 2 = Bag unloading complete -1 = End of table	ARRV DEPT
2	Flight number.	FLTNO
3	Airline number.	AIRLIN
4	Scheduled arrival or departure time.	TIME
5	Number of deplaning passengers met by private car.	
6	Flight arrival or departure time in minutes relative to simulation start time.	

7	Domestic, commuter, or international flight. Domestic is default value.	DOM = 1
		INT = 1
		COM = 1

8	Aircraft type.	AC
9	Gate number.	GATE
10	Numbers of orignating or terminating passengers.	PAX - TPAX
11	Transfer passengers.	TPAX(1)
12	(a) Bag claim area for arriving flight. (b) Number of enplaning passengers waiting at departure lounge for boarding.	(a) BAG (b) No input data.

Table 1 (cont.)

Column		FORTRAN Data Item
13	Number of through passengers on flight using terminal facilities.	TPAX(2)
14	Total terminating bags on arriving flight.	
15	Total transfer bags on arriving flight.	
16	Number of passengers originating from or proceeding to flights outside of simulated system.	TPAX(3)

Table 2.

Contents of Matrices MH4 and ML2

MH4 Element	Usage	FORTRAN Input % Preticketed Card
1,1	Percent of enplaning passengers preticketed for domestic flights.	DOM
1,2	Percent of preticketed enplaning domestic passengers proceeding from terminal entrance directly to security.	DOMDIR
2,1	Same as 1,1 for commuter flights.	COM
2,2	Same as 1,2 for commuter flights.	COMDIR
3,1	Same as 1,1 for international flights.	INT
3,2	Same as 1,2 for international flights	INTDIR

ML2 Element	n = 1/2/3 DOM/COM/INT	FORTRAN Input
n,1	Percent of enplaning passengers using auto mode	PVTCAR
n,2	Percent of enplaning passengers using rental car.	CRENT
n,3	Percent of enplaning passengers using bus or limousine.	BUS
n,4	Percent of enplaning passengers using taxi.	TAXI

starting times of transactions representing originating passengers. ARV1F is the distribution of arrival times prior to flight of passengers on flight departing between 0905 and 2400 hours. ARV2F is the distribution used for all other departure times.

The arrival time distribution, TOS1F, is used for simulation of transfer passengers when the system modeled does not include all concourses of an airport under study. The transfer passengers in this case originate outside the system. This function provides a starting time for transactions representing this passenger type.

Variable FLT1V and FLT2V delay the entry of a transaction into the simulation until 150 and 85 minutes respectively. FLT1V is used for transactions representing orignating passengers of flights departing between 0905 and 2400. FLT2V is applied to all other originating passenger transactions representing transfers outside the system until 40 minutes before flight departure. A fourth variable, TOS2V is equated to MH1(PH1,16)-1. This is the number of passengers orignating outside the system, minus one.

Enplaning Passenger Creation Section

A single transaction representing all departing flights is generated. This has a priority level 10 and contains 11 halfword parameters and 14 byte parameters. This transaction will later be split to represent each flight and subsequently split to represent individual passenger groups. Table 3 lists

Table 3.

Passenger Transaction Parameters

Halfword Parameters

PH 1 Flight table row number

2 Current location, point number

3 a. Maximum random number from subroutine BAGS for deplaning passengers.
b. Passenger storage number in deplaning curb logic.

4 Address parameter,

5 Scratch parameter. Initially contains gate number for deplaning passenger. Used to designate GPSS storage and queue numbers during passenger processing.

6 a. User chain number for bag claim simulation.
b. Storage number of deplaning curb.

7 Facility matrix, MH9, row number for current processing operation.

8 Scratch.

9 Cumulative walking time.

10 Transaction sequence number for passenger and greeter matching.

11 Cumulative passenger waiting time.

Byte Parameters

PB 1 GPSS number of processing function assigned to this transaction.

2 Pointer indicating current step of processing function.

3 Flag indicating type of flight (1 = Domestic, 2 = Commuter, 3 = International).

4 Number of bags.

5 Number of passengers and visitors in party

6 Mode of ground transportation (PVTCAR CURB=1, PVTCAR PARK=2, RENTAL=3, BUS or LIMO=4, TAXI=5)

Table 3 (cont.)

PB 7 Flag indicating deplaning or enplaning passenger.

0 = Deplaning 1 = Enplaning

8 Flag indicating category of deplaning passenger.

1 = Terminating 2 = Transfer
3 = Transmit 4 = Transfer out of system

9 Flag to designate ticketed or non-ticketed status.

10 Car rental agency number.

11 Number of facility type currently entered by transaction.

12 Flag to designate meeting location of deplaning passenger and greeter.

Gate = 1
Bag Claim = 2
Ticketing = 3

13 Passengers in party.

14 a. GPSS number of bag per passenger distribution function.
b. Parking lot number.

the information contained in the transaction parameters describing the passenger group.

Byte parameters 2 and 7 are assigned a value of 1, to respectively initialize the process function pointer and mark the transaction as a representation of an enplaning passenger. At block ENPLO, PH1 is first initialized to a value of 1 and on successive passes is incremented by one to represent the flight table row subscript.

A loop beginning at ENPLO and ending with the block SPLIT 1, ENPLO creates a transaction for each departing flight. The element PH1,1 of MH1 is tested for a negative value contained in the row following the last flight. The last transaction is terminated for this condition. All previous transactions are tested for a value of 1 in MH1(PH1,1) to determine if the row represents an arriving or departing flight. Arriving flights are ignored in the Auxiliary Program and a branch back to ENPLO is executed. Departing flight transactions continue to the SPLIT 1, ENPLO block. The parent transaction proceeds to the next block and the copy transaction to ENPLO for the next flight.

Flight type and gate number are assigned to PB3 and PH5 from columns 7 and 9 of MH1, respectively. A split block to generate two types of enplaning passengers sends one copy to ENPL3. The first category consists of transfer passengers originating outside of the system. The model tests MH1(PH1,16) for values less than or equal to zero to determine if any passengers of this class were input to the flight schedule. If none, the parent transaction is terminated. For positive values, PB8 is assigned a value of 4 to mark this transaction as a transfer out of the system and PB13 and PB5 are set to one to indicate a single passenger with no visitors. The time of entry into the simulation is first calculated as 40 minutes before flight departure time by the variable TOS1V. A SPLIT block creates all of the transactions to fill the number, TOS2V, of transfers out of the system for this flight and directs them to the next block. Individual transactions are further assigned a simulation entry time between 20 and 40 minutes before flight time by delaying them a time randomly selected from the function TOS1F. When the simulation clock advances to this entry time, they are transferred to the WRITE block for writing onto the JOBTAPE.

At ENPL3, transactions representing originating passengers test MH1(PH1,10), for a zero value for this occurence, a default value of 1 is placed in the element. Otherwise, the program branches to a test of the flight departure time in MH1(PH1,10), to determine if it occurs between 0000 and 0905. For departures outside this interval, the program branches to ENPL1. Flights inside the interval proceed to the next block which holds the transaction until 85 minutes before departure. Transaction parameter PH8 is made equal to the number of originating passengers. The number of passengers in a party is assigned to PB13 by a random draw of function PPPEF. These are subtracted from the total in PH8. A test is performed to determine if PH8 is zero or negative. A positive value, indicating that more passenger group transactions will be required, transfers the transaction down to a SPLIT block. This creates a copy transaction to represent another passenger group and transfers it back to assign PPPEF to PB13. A zero or negative value of PH8 allows the transaction to drop through to the next block where the value in this parameter is added to PB13 to adjust the last party size. If PB13 becomes negative, the value of this parameter is made equal to 1. The transaction transfers out of the loop to an ADVANCE block which further delays the transaction activity by an amount obtained from a random draw on function ARV2F.

Flights departing between 0905 and 2400 are routed to ENPL1. The creation of passenger group transactions is performed exactly as before except that these are delayed by variable FLT1V and a random draw on function ARV1F.

After delays, all transactions proceed to ENPL2 for a HELPA call to subroutine FORTM. Random numbers produced by generators 4 and 5 and flight type in PB3 are used as operands. The subroutine assigns a preticketed status and ground transportation modal choice to PB9 and PB6, respectively.

The number of passengers in the party is assigned to PB5. If PB6 is 1 or 2, indicating a private vehicle proceeding to curbside or parking lot, well wishers are added to PB5, based upon a random draw on function WWPPF.

All transactions are written in time sequence on JOBTA1 at location WRITE. These transactions are then terminated.

Program Control Section

A single transaction with priority level 127 and one halfword parameter is generated to initiate and, subsequently, to terminate the activity of the Auxiliary Program. A HELPA call to subroutine CLINK initiates linking of this program and the supporting FORTRAN subroutine FORTM. The next block is a HELPA call to FORTM which completes the linking. After execution of this statement, any HELPA call to FORTM appearing in the Auxiliary Program will operate with two-way communication between the two programs. FORTM also reads input data and places the information in matrices MH1, MH4, and ML2. Any errors detected by FORTM are indicated by setting PH1 of this transaction to a value greater than zero. Upon return from FORTM, PH1 is tested for a value greater than zero. For this condition, the transaction is terminated with a termination count equal to PH1 and the simulation stops.

A zero value of PH1, when no errors are indicated, transfers the transaction to a SPLIT block. The copy is sent to the program location WRITE to become the first transaction on the JOBTAPE. This is performed to provide an initial time, coincident with the start of the simulation to which all succeeding transaction arrival times are referenced on the JOBTAPE.

The first dummy transaction is terminated immediately after entry into the main program from the JOBTAPE.

The original transaction is delayed 100,000 seconds, a time far exceeding a normal simulation time length. Following this delay, the transaction is again split and the second copy is written as the last transaction on the JOBTAPE. This writing prevents an END OF JOBTAPE message during main program execution.

The original transaction is delayed 1 second to permit the WRITE block to execute before the final terminate block is executed. When this occurs the termination count is incremented by 1.

List function number 1 is used to transmit values between the main program and FORTM. Mnemonics are used as Y values of this function. These mnemonics have been previously assigned numerical values by the program. Corresponding mnemonics are used in the FORTRAN subroutine FORTM after calling subroutine MNLINK. Values represented by these mnenomics may be passed in either direction. The entity symbols CMH01, CMH04, CML02 and CLKXH were previously assigned values by SYN statements and are passed to FORTM. This list is placed in this program location for potential expansion for future requirements. Any GPSS entity symbolically represented may be used in this function.

A START statement with a run termination count 1 starts the program and normally stops it after the control transaction is written on the JOBTAPE. This block will also cease program operation if errors are determined in the subroutine FORTM.

MAIN PROGRAM

General Description

The Airport Landside Simulation main program enacts the movement of passengers and visitors through the terminal area. This program generates deplaning passenger and greeter transactions, merges them with enplaning passenger and well-wisher transactions generated by the auxiliary program and models their flow through a sequence of simulated landside processing facilities. Flow, waiting time, queue length and occupancy data are produced for analysis.

The main program contains the following sections: program definitions area, deplaning and enplaning logic sections, facility modules, control section, timer section, transfer flight schedule area, and facility server reallocation section. This document describes those sections in detail and provides a flow chart of GPSS blocks used.

PROGRAM DEFINITIONS AREA

This section contains a description by statements defining limits of GPSS entity numbers, system instructions, SYN and EQU statements to identify facilities, matrix definition statements, function and variable definitions and table definition statements.

Limits on GPSS entity numbers are established by REALLOCATE statements. This program requires entity limits differing from those provided by default values. The program listing contains the values currently used for this simulation model.

System instructions includes an RMULT statement to provide different random number sequences by changing random number generator multipliers. A LOAD statement retains the FORTRAN subroutine, FORTM, and the assembler subroutine BAGS, in core during program execution. The GPSS output module, DAGO6, and an assembler program XACNO, are also kept resident. A JOBTAPE statement defines JOBTA1 as the file of enplaning passenger transactions created by the auxiliary program, and ENPLO as their entry block.

SYN statements are used to pass argument values symbolically to the FORTRAN subroutine, FORTM. A list function in the timer section contains the entity symbols used in the SYN statements as Y values. A HELPA call to FORTM provides a call to the assembler program MNLINK. The arguments of the FORTRAN CALL statement contain variable names which are positionally identified with the list function Y values after NMLINK is executed. The absolute values of the symbols are assigned by the SYN statements of this section and are passed to FORTM. These values specify the first entity numbers assigned by EQU statements used to reserve a sequenced set of GPSS entities. The sequential set is used for simulation of all landside facilities of a given type, for example, all security stations. The types of GPSS entities specified are logic switches, user chains, queues and storages. When the program is executed, the number contained in the EQU sequential set size field for a given facility type must equal or exceed the input parameter FACNO for that type.

SYN statements are also used to transfer the number of colums contained in matrices to FORTM. Halfword matrices 1 through 4, 6 through 9, and floating point matrix 2, utilize this feature.

Matrix definition statements specify matrix numbers type and size. Numbers of columns must agree with preceding SYN statements. Contents of halfword matrices 2, 3, five, 6, and 7 are listed in Table 4. Information for halfword matrix 8 is in Table 5 and matrix 9 in Table 6. Matrices MH 11 and MH 12 are one column matrices used to accumulate counts of persons entering and leaving concourses.

Floating point matrix, ML1, is a single row, 127 column matrix used as a table of floating point random numbers ranging from 0.0 to 1.0. This matrix is filled with random numbers by a group of blocks in the timer section.

TABLE 4

CONTENTS OF MATRICES 2, 3, 5, 6, 7

MH2 - Airline Information Table

1 row per carrier

Column	FORTRAN Input Variable
1 - Enplaning curb facility no.	Epcurb
2 - Per cent preticketed pax using express check-in (0 = No Express Check-in	Expchk
3 - Enplaning curb point for bus stop (If different from column 1)	Bustop

MH3 - Table of Points

1 row per carrier

Column	FORTRAN Input Variable
1 - X - coordinate	POINTX or XY(1)
2 - Y - coordinate	POINTY or XY(2)
3 - Nearest exit point no.	
4 - Nearest entrance point no.	

MH5 - Transfer Flight Table

Contains MH1 row of departing flights taking transfer passengers

MH6 - Walking time between points.

MH7 - Used as work area by bag claim routines

1 row for each possible random number, 1-64, generated by "BAGS"

TABLE 5

MH8 - Used to access facility data in MH9 (contains same information as FORTRAN 'NFASCM' array)

	Number of Facilities in Type	Index Number of Facility Type (one less than number of first facility in type in MH9)
Gate	1,1	1,2
Check-in	2,1	2,2
Security	3,1	3,2
Bagclaim	4,1	4,2
Customs	5,1	5,2
Entrance	6,1	6,2
Exit	7,1	7,2
Enplaning curb	8,1	8,2
Transfer (stairs, etc.)	9,1	9,2
Parking	10,1	10,2
Rentacar	11,1	11,2
Deplaning curb	12,1	12,2
Immigration	13,1	13,2
Tickets & check-in	14,1	14,2

TABLE 6

MH9 - Facility Table

1 Row per actual or dummy facility

Column

1	Facility type
2	Facility number within type
3	Location (point Number)
4-6	Used to identify a facility with other model components

Facility(1)	Column No.	Facility(2)	FORTRAN Input Variable
Bagclaim	4	Deplaning curb	NDELPC
Customs	4	Deplaning curb	NDELPC
Gate	4	Security	NSECUR
Gate	5	Immigration area	NIMMI
Immigration	4	Customs	NCUST
Rentacar	4	Agency code	AGENCY
Rentacar	5	Parking	NPARKL
Tickets & Check-in	4	Airline Code	AIRLIN
Concession	4	Security	NSECUR

The halfword matrices 1 and 4, and floating point matrix 2 have been described previously in Tables 1 and 2 of the Auxiliary Program Description Section.

Functions defined in this section of the program are used for three purposes: (1) retention of program locations representing processing facilities encountered by passenger and visitor transactions, (2) selection of functions containing these locations by random numbers draw; and (3) provision of random variable values to the program by describing cumulative distributions subject to random draw during program operation.

The first group are lists of numerical-valued functions containing symbolic program locations required to be accessed by all transactions of a class. For example, the deplaning domestic passenger is assigned the function DDPlF. Processing of this passenger requires routines to the concourse exit, CNCRO; a rental car counter, RCARO; bag claim, BAGCO; terminal building exit, EXITO; and a ground transportation facility, CGTRO at each of these locations, a transaction parameter is examined to determine if further processing is required. When no processing is simulated the transaction is directed to the next location contained in the function. Otherwise, processing is completed and the transaction is directed to the next program location. The program presently uses 22 functions of this type to describe movement of passengers, well-wishers and greeters through the landside.

Selection functions are either entity functions or discrete attribute-valued functions. Entity functions are used by transfer or transit passenger transactions to select routing functions based upon a random number draw. The X values are treated as arguments of discrete numerical valued functions. The argument values are supplied by a random number generator. Y values are function names.

The other type of selection function used selects ticket counter processing time based upon the airline number of the transaction currently being processed. A discrete attribute valued function performs this task. A list of the selection functions is contained in Table 7.

Cumulative distribution functions used by this program are continuous or discrete. Continuous functions describe landside facility service times, unloading times or empty vehicle parking times. Discrete distributions are used to assign quantities to transaction parameters. For example, passenger and visitor group sizes or number of bags are assigned by random selection from this function type. A list of numerical valued functions is contained in Table 8.

Arithmetic variables are used for assignment of service times to facilities. Values of functions representing the service time distribution are multiplied by the halfword representing the scale factor SCLXH. When service times are applicable to individuals instead of passenger groups, the service time is determined by multiplying the value from the distribution, the scale factor and PB13, then number of passengers

TABLE 7

Functions Used for Selection

TADF	-	Transfer passenger, long-stay/different concourse
TDLSF	-	Transfer passenger, long-stay/same concourse
TDLSF	-	Transfer passenger, short-stay/different concourse
TDSSF	-	Transfer passenger, short-stay/same concourse
TDPXF	-	Transit passenger
ATKIF	-	Choose airline ticket/check-in variable
BUNIF	-	Baggage unload entity function (aircraft type)

TABLE 8

Cumulative Distribution Functions

Continuous Functions

BUN2F to BUN3F	-	Baggage time to claim area by aircraft type
CHK1F to CHK4F	-	Ticket check-in time by airline
GAT3F to GAT4F	-	Gates process time by airline
IMM1F	-	Immigration process time
CUS1F	-	Customs process time
PAR1F	-	Parking lot exit service time
RCA1F	-	Car rental processing time
SEC1F	-	Security service time per person
CSCKF	-	Curbside checkin process time

Discrete Functions

PPPDF	-	Passengers per party deplaning
GRPPF	-	Greeters per party (parties with greeters only)
DBAGF	-	No. of bags - domestic flight
RCAZF	-	Car rental agency selection

plus visitors in the group. A list of service time variables is given in Table 9.

Other arithmetic variables are used to calculate aircraft and vehicle unloading times, numbers of transactions assigned to ground transportation modes, starting times relative to flight departures and GPSS storage numbers for deplaning and enplaning curb, double parking and storage areas. Two arithmetic functions are used for random number generation.

Boolean variables are used for testing conditions at curbside to determine if congestion due to double parking will cause queueing of traffic attempting to bypass a section. Other Boolean variables test passenger transaction parameters for transfer passengers status and preticketed status.

TABLE 9

Service Time Variables

CHK1V - CHK4V	-	Ticket and check-in time by airline
CUS1V	-	Customs time
GAT3V to GAT4V	-	Gate service time by airline
IMM1V	-	Immigration
PAR1V	-	Parking lot service time
RCA1V	-	Car rental checkout time
SEC1V	-	Security service time (party)
CIRCV	-	Vehicle recirculation time

Control Section

This section provides a method of routing transactions from one program logic section to another. Routing functions are assigned the byte parameter 1 by other program sections such as the enplaning and deplaning passenger logic. The transaction is transferred to the control section which, in turn, transfers it to the next program location.

The first statement of the control section at CTRLO assigns the value of the function contained in PB1 to PH4. The argument of the function is contained in PB2. Byte parameter 2 is assigned a value 1 when the transaction is created. After the transaction is routed through CTRLO, the second block of this section increments PB2 by 1. This will make PB2 point to the next value of the routing function if the transaction is later returned to CTRLO. The third block of this section CTRL1 transfers the transaction to the program location contained in PH4. This location is normally the current value of the routing function in PB1.

When a transaction is directed to a location outside the routing function, the transfer is performed without incrementing PB2. Under this condition, the program logic section specifying the transfer, places the name of the next program location in PH4 and directs the transaction to CTRL1. An example of this operation is in the enplaning curb logic. A vehicle transaction unable to locate a parking space has the address of the recirculation road logic placed in PH4 and is transferred to CTRL2 to effect this routing. After simulating the recirculation process the vehicle is directly transferred to the enplaning curb logic. Successful use of the simulated curbside will then provide a transfer to CTRLO to continue transaction routing according to the PB1 function.

A special purpose block at CTRL8 transfers deplaning passenger transactions using the parking lot from the ground transportation module to the location PARKO. This location is the starting block of a module representing the parking facility payment booth. The transfer is unconditional for transactions entering the CTRL8 block.

Secondary functions of the control section are, (a) the provision of ground transportation facility program locations to represent buses and (b) provision of locations for transaction terminations after simulation of landside processing is completed.

For transactions representing terminating deplaning passengers, the selection function CTR1F routes the transaction to the deplaning curb, parking facility, or rental car parking facility. The mode selection defined in PB6 determines the facility. Program locations to represent these facilities are placed in PH4 at CGTRO and the transfer is executed in the next block. Originating enplaning passenger transactions use the function CTR2F to select the enplaning curb parking facility or rental car return location. Transfers are performed by the value in PH4.

Transactions representing buses are generated at intervals specified by input data. The time between creation of transactions is ABUXH for buses proceeding to the enplaning curb and DBUXH for buses departing the deplaning curb. These are rotated to program locations ENPC6 and DPLC6 respectively.

The section providing transaction termination begins at the block DEP99. The total walking time PAXWT, accumulated on the landside is recorded by a TABULATE block for each transaction routed to DEP99, END99, or TRX99.

This section also tabulates the total simulated waiting time spent on the landside by transactions. A new table is produced for each simulated hour. The hour is designated by the value of the byte savevalue PXTBN. The transaction is entered into the table by a second TABULATE block and then terminated.

In general, deplaning passenger transactions are routed to DEP99, enplaning passenger transactions to ENP99 and transfer passenger transactions to TRX99.

DEPLANING PASSENGER LOGIC

This section of the program generates transactions representing individual flights. It then creates greeter and deplaning passenger transactions through the use of SPLIT blocks. Attributes assigned to transaction parameters by this section include: number of bags, meeting location, modal choice, sequence numbers for passenger-greeter matching, process function, party size, and parking lot number. The deplaning passenger logic simulates the discharge of passengers from the aircraft and performs the matching of those deplaning passenger and greeter transactions assigned to meet at the airplane gate.

An initial GENERATE block provides one transaction of priority level 10. This transaction contains 11 halfword parameters and 14 byte parameters which are available for attribute assignment. The process function pointer, PB2, is initialized to 1. A copy of this transaction is sent to program location XFLTO to initialize a table of transfer flights in halfword matrix five. An ADVANCE block delays the original transaction by one second to permit the transfer table initialization before flight and passenger transaction creation.

Byte parameter 11 is assigned the value 1 to indicate that the transaction begins activity at the gate. A loop beginning at program location DEPL0 creates a transaction for each arriving and departing flight. At DEPL0 the MH1 row counter, PH1, is incremented by 1 each time a transaction passes through the block. Column 1 of each row of MH1 is tested for a negative value. The row after the last flight in the schedule contains the value -1 in this position and the last transaction is terminated when this is reached. Rows representing flights cause the transaction to branch around the terminate block to an ADVANCE block. At this location, the transaction is delayed until the difference between MH1 (PH1,6), the flight arrival or departure time relative to simulation start, and the absolute clock time is one hour. A SPLIT block completes the loop by sending the flight transaction to the next block and the copy back to DEPLO.

The flight transaction proceeds to a TEST block to determine if an arriving or departing flight is represented. Departing flights proceed to program location GATE9, where the boarding process is simulated. Arriving flights transfer to DEPL5. At this location, the value 1,2, or 3 is assigned to PB3 to indicate if the flight is a domestic, commuter or international type, respectively.

Byte parameter 3 is tested for a value of 1. Nondomestic flight transactions are transferred to DEPL1. Domestic flight transactions are assigned DDP1F, the routing function for domestic passengers to PB1, and DBAGF, the bag distribution function for domestic passengers, to PB14. These transactions are then transferred to DEPL3.

At DEPL1, PB3 is tested for a value of 2. Commuter flight transactions proceed to the next block where the commuter passenger routing function, DCP1F, is assigned to PB1. The commuter bag distribution function, CBAGF, is assigned to PB14 and the transaction transfers to DEPL3.

International flight transactions are transferred from DEPL1 to DEPL2. At DEPL2, the international deplaning passenger routing function DIP1F, is assigned to PB1. Byte parameter 14 is assigned IBAGF, the international passenger bag distribution function. This type of transaction proceeds to the next block at location DEPL3.

At DEPL3, the halfword savevalue SEQ1H, used for assigning sequence numbers to transactions, is tested for a value equalling or exceeding 32,000. Under these conditions, SEQ1H is reset to zero, then the transaction proceeds to the first of three consecutive SPLIT blocks. Otherwise, the transaction is transferred directly to the first SPLIT block.

Three program locations, DEP25, DEP15 and DEP17 are the destinations of the copy transactions from the SPLIT blocks. At DEP25, the transaction is initiated which will represent greeters using both the parking facility and the curbside after meeting passengers inside the terminal building. Greeter transactions to be created for merging with terminating passenger transactions after meeting inside the terminal building and departing the airport directly from the parking facility are started at DEP15. Program location DEP17 is the starting point for transactions to be generated representing greeters meeting passengers at curbside.

The flight transaction proceeds to DEPL7 where the creation of deplaning passenger transactions is initiated. A percentage of these passenger transactions will be matched later with greeter transactions at locations specified by the logic of both the following sections and the sections creating deplaning passengers.

The first copy transaction proceeds to DEP25 where PB6 is given a value of 1 to indicate that the curbside will be used after parking. The number of greeters in this category for this arriving flight is determined by the variable FLT2V. This number is assigned to PH8. The transaction transfers to location DEP16 to Test PH8 for the existence of this greeter type on this flight. If PH8 is zero, this copy transaction is terminated. For non-zero values of PH8, the transaction bypasses the TERMINATE block and adds the number in PH8 to MH1 (PH1,5).

The random number table in ML1 is used as the argument of functions which will assign the numbers of passengers per party and bags per party to be transacted. The MH1 row number, PH1, is used to assign the starting column of ML1. Variable DPL3V is PH1, modulo125 and is assigned to PB10 to designate starting the ML1 column number for this flight. At DEP 12, PB10 is incremented by one, then tested for a value in excess of 124. When PB10 is greater than 124, it is reset to 1.

The value of the function PPPDF; the number of passengers per deplaning party, is subtracted from the total on the flight. PB10 is increased by 1, then the number of bags drawn from the function contained in PB14 by using ML1 (1,PB10) as the argument value, is assigned to PB4. The pointer PB10 is again incremented by 1 and then the random number drawn from ML1(1, PB10) is tested for a value less than or equal to GRGXL, the percentage of passengers meeting greeters at the gate. When this condition occurs, the transaction is marked for gate meeting by assigning the function GREGF to PB1 and the value 1 to PB12. This transaction type is then transferred to DEP14.

Greeter transactions not meeting at the gate are transferred to a TEST block to determine if the passenger to be greeted will have checked bags. The number of bags is contained in PB4 and is tested for zero.

For the zero value, the function GRELF is assigned to PB1. This function routes the greeter to the enplaning level for meeting the deplaning passenger in the ticket lobby.

Greeter transactions with a non-zero value in PB4 are assigned the function GREBF. This directs all remaining greeters proceeding inside the terminal building to the bag claim area for meeting.

Halfword parameter 8 is tested for a value greater than zero to determine if another greeter transaction is required. When PH8 is greater than zero, a copy transaction is sent back to DEP12 by a SPLIT block to become the next greeter transaction. The parent transaction from the SPLIT block is assigned the parking lot number in PB14 and the number of greeters in PB5 from the distribution function GRPPF. The sequence counter, SEQ1H, is incremented by 1 and the current value is assigned to PH10.

The transaction is held at an ADVANCE block for a time duration drawn from the function DCA2F, the distribution of arrival times at the landside for greeters. The times of greeter arrival are between one hour prior to flight arrival and 10 minutes after. After departure from this block, the transaction proceeds to CTRLO to begin simulation activity with routines prescribed by the function assigned to PB1.

The second copy transaction proceeds to DEP15. Byte parameter 6 is assigned a value of 2. This indicates that only this parking facility will be used by the greeter vehicle. After greeter and terminating passengers are matched inside the terminal building the group departs the landside directly after leaving the parking facility. The numbers of greeters from this category is calculated from the variable FLT3V and assigned to PH8. At the next block, DEP16, the test is performed for the existence of this greeter type in PH8. All further processing of this transaction is identical to that of the first copy transaction.

The third copy is routed to DEP17. The value 1 is assigned to PB6 to indicate curb usage. This transaction will represent greeters meeting their assigned passengers at this curbside. The number of passengers of this category is determined by the variable FLT4V and is assigned to PH8. A test is made to determine if PH8 is zero. The absence of this greeter type terminates this copy. When greeters of this type are to be simulated, the number of these is added to the total of terminating passengers with greeters. This number is contained in MH1 (PH1,5). The starting column number of the random number table in MLI is placed in PB10 as before. Halfword parameter 4 is given the program location DCARO as the first and only destination for this transaction.

The table pointer is incremented as before and the party size is drawn at random from PPPDF and subtracted from PH8. The number of bags is also determined and assigned to PB4. A test determines if PH8 is greater than zero. When true, indicating more greeter transactions are required, a copy is sent back to DEP13 to represent the next greeter group. The parent transaction has zero assigned to PB14, GRPPF to PB5 and a sequence number, SEQ14, assigned to PH10. The transaction transfers to CTRLI which will affect the transfer to DCARO. The greeter arrival time function DCA2F, is located at DCARO.

The original flight transaction is transferred to DEPL7 and will be used to generate all deplaning passengers for the flight. At DEPL7 the transaction is held until flight arrival time by an ADVANCE block, which delays further processing for 3600 seconds. A GATE block holds back succeeding aircraft until this baggage unloading logic, following location BUNLO, resets logic switch DPL1G. This allows deplaning passenger transactions for this flight to execute the assembler subroutine BAGS without interference from a simultaneously arriving flight. After the GATE block is passed, DPL1G is placed in a set condition by this transaction. The gate number of the flight in MH1 (PH1,9) is assigned to PH5 and also to PH7. The point number of the gate is assigned to PH2.

The maximum passenger unloading time ACUNL is calculated by multiplying the total number of passengers by 3 seconds and adding 90 seconds. A SELECTLR block places the number of the first chain between the limits CHA1B and CHA2B in a reset condition into PH6. Numerical values of these limits were previously set by EQU statements. A LOGICS statement places this chain in a set condition. The chain represents a bag claim device which will hold the passenger transactions in the bag claim logic until all bags are delivered from the flight.

Three split blocks route copy transactions to DEPL4, DEPL6 and DEP10 to initialize the generation of transactions representing terminating, transit and transfer out of system passengers, respectively. The parent transaction proceeds to a TEST block to determine if a flight is available in the MH5 (1,1) element of the transfer flight table for accepting transfer passengers. If no flights are available, the number of transfer passengers on the flight represented by this transaction is added to the halfword save value XFRXH and this transaction to represent transfer passengers is terminated.

When MH5(1,1) indicates a flight is available, the transaction proceeds to DEPL8 where MH1(PH1,11) is tested to determine if the arriving flight has any transfer passengers. If none, the flight transaction is terminated. When transfer passengers are to be simulated for this flight, PB8 is assigned a value 2, as an indicator of a transfer passenger and byte parameters 5 and 13 are assigned the value 1. This assumes all transfer passengers are travelling singly, rather than in a group.

A split block creates the number of transactions required to simulate transfer passengers on the flight. This number is obtained from MH1 (PH1,11). These transactions are tested to determine if the arrival flight is international. When this occurs, the international transfer passenger routing function, TIP1F, is assigned to PB1. A HELPA call to FORTM is used to assign the next flight by random number draw. Transactions not representing international arriving flights are routed directly to this HELPA block. The departing flight MH1 row number is assigned to PH1 at this time. The flight table matrix element MH1 (PH1,7) is tested for a value 3 to determine if the flight is international. When this occurs, the transaction is given a routing function TDP5F. This transaction is then transferred to DEP11.

The non-international transfer passenger is transferred to a SAVE VALUE block, where the gate of the departing flight is retained as the save value SAVXH. The time difference between the departing flight time and the current clock time is tested to determine if the duration exceeds 45 minutes.

Those time differences exceeding this value are categorized as long term waiting times and the transaction is transferred six blocks to the next test. Transactions with a short stay, less than 45 minutes, test the value of the MH9 matrix elements in column 4 for the gate numbers of the arrival and departure gates to determine if both are on the same concourse. When this occurs, the selection function TDSSF is assigned to PB1 and the transaction proceeds to DEP11. Transfer passenger transactions with a short stay and with arrival and departure rates on different concourses are assigned selector function TDSDF and transfer to DEP11.

The passengers with a greater duration than 45 minutes between current absolute clock and departure times are assigned selection functions TDLSF and TDLDF for the same or different concourses, respectively. The selector functions assign routine functions to PB1 based upon input percentages and the current value of a random number generator. All transactions proceed to DEP11 which executes a transfer based upon the address assigned to PH8 by FORTM.

The transit passenger transactions, represent continuing passengers who exit an aircraft, circulate within the terminal building and return to the same flight. These are marked as transit passengers by assigning 3 to PB8 and designated as single passengers with no visitors, by assigning 1 to PB5 and PB13.

Transit passenger transactions are created by a SPLIT block using MH1(PH1,13) to specify the numbers of copies. The parent transaction is terminated after the SPLIT block. An ADVANCE block simulates the deplaning process by delaying the entry of each transactions by an amount ACU2V, the random variable specifying service times. A HELPA block calls FORTM to assign the MH1 row number of the next flight at the same gate. A process function is selected by assigning TDPXF to PB1. The function chosen by this selection function routes the transit passenger through selected landside facilities. If a process function cannot be selected, PB1 equals zero and the transaction is terminated. When PB1 is non-zero, the function executes the assigned routing function and proceeds to CTRLO to begin landside processing.

Transfers out of the system are routed to location DEP10. The number of transactions to be generated is specified by MH1(PH1,16). After departing the aircraft, these are assigned the routing function TOSDF, and transferred to CTRLO to begin processing.

Terminating passengers are routed to DEPL4 where PB8 is assigned the value 1 to designate the transaction as representing the deplaning terminating passenger.

Transaction sequence counter SEQ2H is tested for a value equal to or greater than 32,000. When this condition exists, the value is reset to zero. Otherwise, the transaction is directed to a series of SPLIT blocks. Copies are directed to DEP18, DEP19, DEP22 and DEP26 to generate passenger groups using private vehicles and to assign sequence numbers for passenger greeter matching. A copy is also directed to a section generating transactions for simulation of passengers using public modes of transportation. The flight transaction is transferred to BUNLO to initiate simulation of the baggage unloading process.

The first copy is directed to DEP18 where PB6 is assigned the value 1 to indicate that the curbside will be used after the greeter-passenger meeting and removal of the vehicle from the parking garage is simulated. The number of transactions from this category is calculated by FLT2V and assigned to PH8. The procedure for determining party size is identical to that previously described for greeters. The random number selections are drawn from the table ML1, using the PH1 row number, modulus 125, as the pointer.

Using the same sequence of numbers for the passenger transaction as for the greeter, a transaction is created for each passenger group from the distribution function PPDF. The number of passengers in the group is assigned to PB13 and subtracted from PH8 as before. This is repeated until PH8 becomes zero or negative. When negative, the last party size is adjusted to force a zero value in PH8.

Subroutine BAGS is called by the passenger transaction and uses the argument FN*PB14 to determine the number of bags to be assigned to the passenger group transaction in PB4. This is identical to the usage of the bag function performed by the greeter transaction and the same random number from ML1 is used by both transactions. For each terminating passenger transaction, subroutine BAGS also selects one random number for each bag specified in PB4 using one of the GPSS random number generators. The largest random number generated for the transaction is assigned to PH3. This value will be used later in the baggage unloading logic to release the passenger from the chain representing the bag claim device.

Sequence numbers, SEQ1H and SEQ2H assigned to PH10 in greeter and passenger transactions, respectively, are identical for the pair to be matched later. The numbers of greeter and passenger transactions proceeding to the gate, bag claim or lobby for meeting are determined identically. The major differences in the two paired transactions are; (1) assignment of different processing functions to PB1, (2) assignment of passenger group size number to PB13 of the passenger transaction and zero to the same parameter of the greeter transaction, (3) assignment of the greeter group size to PB5 of the greeeter transaction, and assignment of passenger group size to PB5 of the passenger transaction.

The second copy is directed to DEP19 where PB6 is assigned the value 2 and the number of transactions needed to proceed directly out of the airport from the parking facility after meeting is calculated from VFLT3V and assigned to PH8. The same processing as the transaction processing to DEP18 is executed.

The third copy proceeds to DEP22 where the transactions representing terminating passenger groups greeted at the curb are generated. A value of 1 is assigned to PB6. The number of transactions required is determined from FLT4V and assigned to PH8.

All of the transactions generated from the three above copies are transferred to DEP24 where the deplaning and gate meeting processes are simulated.

The fourth copy, routed to DEP26, assigns a value of 2 to PB6 and calculates the number of transactions required to represent passengers without greeters using private auto. The numbers of passengers per party is drawn from the function PPPDR, using the random number table in ML1 as a source for argument values of the function. Subroutine BAGS is also executed with a number from ML1 as the argument of the bag per passenger function in PB14. If the number of bags is zero, the function DLP1F is assigned to the transaction representing this passenger group. These transactions proceed directly to DEP24 for deplanement.

The final SPLIT block directs a copy transaction to areas creating transactions representing terminating passengers using taxi, limousine or bus for ground transportation. The number of passengers on the flight utilizing ground transportation modes other than private auto is provided by FLT6V and is assigned to PH8. The function PPPDF is also used to determine party size and is assigned to PB13. A loop determining PH8 by each party size until depletion, creates the required number of transactions. Each transaction executes a HELPA block and is assigned a mode by FORTM using random number selection. The subroutine BAGS is also executed. For passengers other than those using car rental, the number of bags, PB4 is tested for zero. When this condition occurs, the function DLP1F is assigned to PB1. All BAGS transactions are transferred to DEP 24.

AT DEP24, all terminating deplaning passenger transactions are delayed by the value ACU2V a random unloading time from a uniform distribution. Byte parameter 12 is tested for a value of 1, to determine if the passenger transaction will attempt to match the corresponding greeter transaction. Those passenger transactions with PB12 not equal to 1 are transferred to CTRL0 to begin landside routine. Those with a value 1 in PB12 are transferred to an UNLINK block to examine this user chain GREGC. The parameter PH10 of the passenger transaction is compared to the same parameter of all the greeter transactions waiting on the chain. When transactions match, the greeter transaction is removed from the chain and transferred to CTRL1. The passenger transaction proceeds to the next block at DEP29. At this location, the transaction is held until the logic switch PAS3L is placed in a set condition by the greeter transaction. This setting is performed in the gate logic section by the greeter transaction after transfer from CTRL1. The number of greeters is assigned to the savevalue PAS32 and the parking lot number to PAS33. These are obtained from PB5 and PB14 of the greeter transaction respectively in the gate logic section. After PAS3L is set, the number of greeters, PAS32, is added to PB5 of the passenger transaction. The parking lot number, PAS33, is placed in a reset condition for use by the next passenger transaction finding a matching greeter. The active transaction proceeds to CTRLO to continue landside routing.

When the terminating passenger transaction marked for gate meeting does not unlink a greeter, the transaction is transferred to DEP28. An ASSIGN block places the program location DEP29 into PH4 and transaction is linked to GREGC. This will transfer the passenger transaction to this location when unlinked later by the greeter transaction in the gate logic.

Enplaning Passenger Logic

The first block in this section is at location ENPLO. Because this is the B operand of the JOBTAPE statement in the program definition statement section, all transactions created by the Auxiliary Program and written on the JOBTAPE are routed to this block. The first transaction does not represent an enplaning passenger group, but was written on the JOBTAPE to provide an initial time coincident with the start of the simulation to reference the entry times of all originating enplaning passenger transactions. This first dummy transaction is transferred to ENPL9, where a CHANGE block referencing CHNGO redefines the operation performed by the block at ENPLO. All succeeding transactions will transfer to program locations in the enplaning passenger logic based upon the selection function ENPlF. The dummy transaction is then terminated.

Function ENPlF will route the JOBTAPE transactions to ENPL1, ENPL2 or ENPL3 based upon respective values of 1, 2 or 3 in PB3. The parameter was assigned these values in the Auxiliary Program tu represent domestic, commuter or international passengers, respectively.

Transactions representing transfers out of the system are arbitrarily assigned a value 1 as the point number designator in PH2 and the routing function TOSEF is assigned to PB1. This transaction is transferred to CTRLO to begin landside processing.

The originating enplaning domestic passenger transaction is assigned the function EDPIF in PB1 and a random number from the variable RND2U in PB10. The latter serves as an argument to DBAGF, the function used in the next block to assign the number of bags to PB4. This transaction then transfers to CTRLO.

Enplaning commuter passenger transactions are assigned their processing function, ECP1F, to PB1 at location ENPL2. The number of bags assigned to PB4 from function CBAGF and the transaction is transferred to CTRLO.

International enplaning passengers transfer to ENPL3 from ENPLO. The routing function assigned to PB1 is EIP1F and PB4 obtains the number of bags from the function IBAGF.

Facility Modules Section

This section describes the program logic of 17 modules used to simulate landside processing facilities. These operate independently, with few direct linkages from one module to another. Transactions are generally routed from one of these modules to another through the use of the routing function and a transfer to CTRLO of the Control Module or the assignment of the address of another module to PH4 and a transfer to CTRLI, also in the control module.

Two modules, Baggage Unloading Logic and Deplaning Curb (Cars) simulate landside activities not performed by passengers. All others involve passenger simulation.

1. Bag Claim

This module begins at location BAGCO, when a test is performed on PB4 of the entering transaction to determine if the number of simulated bags associated with this transaction is zero. If PB4 is zero, the transaction returns immediately to CTRLO. All other transactions proceed to a HELPA block and return from FORTM with the point number of the bag claim facility assigned to PH2; the value 4 to PB11 and the landside facility number to PH7. The transaction is then advanced by TRUXH, the walking time from the last facility.

Byte parameter 13 is tested for a zero value. When this occurs, indicating that a greeter transaction is being processed, the transaction attempts to unlink the corresponding passenger transaction from the chain GREBC by matching PH10 of both transactions. If successful, the greeter transaction proceeds to a GATE LR block at BAG 3. Since logic switch PAS4L is normally in a reset condition, the greeter transaction continues and assigns the number of greeters in PB5 to PAS42 and the parking lot number PAS43. Logic switch PAS4L is set to allow the passenger transaction to obtain these two values. If the vehicle used by the greeter was assigned to proceed to the curb, then the greeter transaction is routed to GRTY3 to simulate removal from the parking facility. Otherwise, the greeter transaction is terminated. A greeter transaction unable to unlink a passenger transaction is transferred to BAGC2 where it places BAGC3 in PH4 and then links this transaction on the user chain GREBC.

Passenger transactions are routed to BAGC1. A savevalue recording the occupancy at the point where the bagclaim facility is located is incremented by PB5, the number of passengers in the party. Matrix element MH1(PH1,1) is tested for a value of zero, to determine if all bags have been delivered to the bagclaim for the flight. When this occurs, no time for bag delivery or pickup is simulated. The passenger transaction parameter byte 12 is tested for a value 2, to determine if the simulated greeter and passenger matching process will occur at bagclaim. For PG12 not equal to 2, the passenger transaction proceeds to BAGC4 for a transfer to CTRLO.

When PB12 is 2, the passenger transaction attempts to unlink a matching greeter transaction from chain GREBC. If successful, the passenger transaction waits for the unlinked greeter transaction to set PAS4L. The values PAS42 and PAS43 are subsequently assigned to PB5 and PB14, respectively, and logic switch PAS4L is reset. The passenger transaction transfers to BAGC4 for immediate transfer to CTRLO.

Passenger transactions unsuccessful at unlinking greeter transactions are sent to BAGC5, and BAGC6 is assigned to PH4 and the passenger transaction is placed on the chain GREDC.

When bag delivery from the aircraft is incomplete, the address BAGC7 is placed in PH4 and the passenger transaction is placed on the chain assigned to PH6 by the deplaning passengers logic. The unlinking from this chain occurs in the Baggage Unloading Logic Module. Upon return from BAGC7, after unlinking from the PH6 baggage chain, the test for greeter matching and subsequent activity described above is performed.

2. Baggage Unloading Logic

This module is accessed by arriving flight transactions generated by the Deplaning Passenger Logic. The first block of this module, at BUNCO, changes the priority of the entering flight transaction from 10 to 5, and, through use of the BUFFER operand, places it below the passenger transactions on the current events chain and restarts the chain. This operation allows the passenger transaction to execute the assemble subroutine BAGS before the baggage unloading logic operates.

The number of halfword and byte parameters of the flight transaction are changed to 9 and 40 respectively. A HELPA block is enacted to assign MH7 row numbers to transaction parameter bytes. Each element of MH7 represents a count of simulated passenger bags placed in the array by subroutine BAGS. Subprogram FORTM extracts each element beginning with the lowest numbered and computes the cumulative sum of the elements. At the same time FORTM examines the cumulative sum after each element is added. Each time the sum exceeds an integral multiple of the C-operand, it places the MH7 row number in a parameter byte, starting with number 40 and decremeeting to number 1, and then increases the C-operand multiplier by one.

An ADVANCE block simulates the aircraft unloading time based upon a random draw from a distribution. Bag delivery is simulated by unlinking all passenger transactions from the PH6 bag claim chain. This process is performed in a loop, with the flight transaction byte counter decremented from 40 to 1. The PH3 value assigned by BAGS to the passenger transaction is compared to the MH7 row number in the byte parameter of the flight transaction. All transactions with PH3 less than or equal to the value in the byte parameter are unlinked. The flight transaction time is advanced 30 seconds between each decrement, simulating a large wait for passenger transaction with high PB3 values than those with low. Passenger transactions are routed to BAGC7 by a transfer to CTRL1.

The PH6 chain is released by a LOGIC R block. Matrix savevalue MH1 (PH1,1) is assigned a value 2, indicating that the delivery of all bags is complete. The flight transaction is then terminated.

3. Ticketing and Check-In

This module is entered by enplaning and some deplaning and transfer passenger transactions. Greeter transactions, routed to simulate the meeting of greeters and deplaning passengers without baggage, also utilize this program section.

At CHEKO, the first program location of this module, the Boolean variable CHK1B is used to test transaction parameters PB7 and PB9 and the value produced by random number generation RN7. The test simultaneously determines if PB7 equals 1, indicating an enplaning passenger transaction; if PB9 is zero, indicating a preticketed status; and, if PB7 is less than the input percentage of preticketed passengers proceeding directly to the gate. When CHK1B is true, the transaction bypasses the ticketing and check-in procedure and is routed directly to CTRLO, for transfer to the Security Module.

Transactions requiring processing by the module execute a HELPA block. They are assigned the following parameter values by FORTM: point number is PH2; program location CHEK2 or CHEK3 is PH4; the GPSS storage or queue number for the check-in facility of the MH1(PH1,3) airline is PH5; the landside facility number is MH9 in PH7; and the process code number 2 or 14 in PB11, to indicate express or full service check-in.

An ADVANCE block simulates the walking time from the previous facility. Byte parameter 13 is tested for a zero value. When PB13 is zero, indicating a greeter transaction, a set of logical operations simulating the greeting process and identical to those performed at Bagclaim, is executed.

Passenger transactions are transferred to CHECK9 and PB12 is tested for a value of 3, the flag indicating a deplaning passenger transaction to be greeted at the enplaning level lobby. These deplaning passenger transactions also simulate the greeting process using program logic identical to that in the Bagclaim Module.

Transactions with PB12 not equal to 3, transfer to CHEK6. At this location another test is performed, this time on PB8, to determine if the transaction represents a terminating passenger. This transaction type is transferred to CTRLO. All other transactions increment the occupancy PH2 savevalue by the value in PB5.

The passenger transaction requiring a simulation of the check-in process joins a queue at the PH5 facility and the time of entry is marked. When the transaction is ready to enter the service process simulation, the waiting time deviation is placed in PH11 and the transaction transfers to CHEK2 for a preticketed passenger check-in, or to CHEK3 for full service. Service time for each facility is drawn from the applicable service time distribution. For the preticketed passenger transaction only one distribution is used. However, for simulation of full service ticketing, the selection function ATK1F will choose the service time from the distribution applicable to the airline number represented in MH1(PH1,3). Halfword savevalue CHKXH is assigned the value of the airline number and is used as the argument for the function ATK1F. Upon completion of the simulated ticketing and check-in process, the occupancy count is decremented by PB5. The flow count in MH13 is incremented, and the transaction transfers to CTRLO.

4. Immigration

This module is entered by deplaning international passenger transactions. At IMMIO, a HELPA call to FORTM results in the following parameter assignments: point number of the facility is PH2; GPSS queue and storage number is PH5; landside facility number is PH7 and PH8; and the value 13 in PB11 to denote immigration processing.

The walking time from the previous facility, TPVXH, is simulated by an ADVANCE block. The PH2 occupancy count is incremented by PB5. The transaction enters a QUEUE block if the storage is full or begins simulated service if it is not full. The simulated service time is obtained from the variable FMM1V. After leaving the PH5 storage, the transaction decrements the occupancy count by PB5 and increments the immigration flow count IMIG by the same value. The module is exited by a transfer to CTRLO.

5. Customs

This module is also entered by international deplaning passenger transactions. The HELPA block at CUSTO returns with the same type of information in PH2, PH5, and PH7 as in the immigration module. PB11 is assigned a value 5 to signify customs processing.

All simulation processing is identical to that performed in the immigration module. The service time is provided by the variable CUS1V.

6. Concourse Exit

This module provides a program location for accumulating the count, in MH11, of simulated passengers and visitors exiting concourses and entering the terminal lobby. At CNCRO the HELPA block assigns the point number of the exit to PH2 and the FACNO numbers of the security station for this concourse to PH5. The walking time from the gate to this exit is simulated. After incrementing the matrix element MH11 (PH1, PH5) by the value PB5, to increase the passenger-visitor concourse count, the transaction is transferred to CTRLO.

7. Deplaning Curb (Cars)

In this module, vehicles proceeding to the curb to greet terminating passengers are simulated. Assignments to curbside or double parking are made by this module, based upon current space availability. A limited queue at each curbside section is established if storage representing curbside or double parking is filled. Congestion of vehicular traffic resulting from lane blockage due to double parking and queueing is simulated. The program logic inspects only the curbside section assigned to the bagclaim area of the terminating passenger for space availability. When no space is available the vehicle transaction simulates a recirculating process.

Vehicle transactions routed to DCARO represent greeters arriving at the airport for simulation of passenger-greeter meetings at the curbside. These are delayed by a random draw of the function DCA1F which represents the distribution of times of arrival of greeters at the airport landside. These vehicle transactions are joined at DCAR1 by those representing vehicles removed from the parking lot by the greeter transaction after simulation of their matching process within the terminal building. Vehicular counts on the entry road and deplaning curbside approach section are incremented by 1 at DCAR1 and the next block, respectively.

A HELPA call to FORTM assigns a GPSS storage number to the transaction in PH6 and the value 1, 2, or 3 is assigned to PH10 to indicate curbside parking, double or queuing, respectively. If no available space is located, a simulated recirculation of the vehicle will be performed. PH6 is then assigned a value zero and PB10 is 4.

The byte parameter 4, as operand in the HELPA block, determines if the vehicle will proceed to the enplaning or deplaning curb. When PB4 is zero, indicating that the passenger and greeter transactions are without simulated bags and the greeting process will take place at the enplaning level lobby, FORTM assigns a storage number to PH6 corresponding to enplaning curbside parking, double parking or queuing. When PB4 is non-zero, the assignment is made to a deplaning curb storage. Recirculation is provided for either PB4 condition when required.

Upon return to GPSS, a test on PB4 for a non-zero value is performed. Transactions with PB4 zero values are transferred to DCARO. Because the simulated vehicle may pass other curbside sections to arrive at the assigned section, the program must determine if lane blockage occurs at this section to be bypassed. The first deplaning curb storage number minus 1 is placed on PH8 and PH4 and the first deplaning double parking storage number minus one is PH5.

A variable DEPLS is established and assigned in PH8 for the purpose of comparison with PH6. This will determine if the GPSS storage number of the curbside parking, double parking or queue, contained within the curb section where the transaction is currently located, matches the destination storage of the transaction. Because the value of DEPLS depends upon PB10, comparison of PH8 with PH6 will only be between storage numbers representing identical facility types. Recirculating vehicle transactions do not calculate the value of DEPLS for assignment in PH8. These transactions transfer to DCARB and are joined by the transactions with DEPLS assigned to PH8.

Vehicle transactions transferred to DCARD are assigned a value of 1 less than numbers of the RPSS storages representing the curbside at the first enplaning curb section in PH4 and the double parking area at the first enplaning curbside section in PH5. The variable ENPLS performs the same function as DEPLS but operates on enplaning curb storage and is assigned to PH8. All transactions proceed to DCARB.

The loop for simulating lane blockage and delay is begun at DCARB. PH8 is incremented by 1 each time this block is passed. Non-recirculating vehicle transactions test PH8 against PH6 to determine if the current curbside facility is the destination. When PH8 equals PH6, the transaction proceeds to DCARA to simulate parking. Otherwise, PH4 and PH5 are each incremented by 1. A Boolean variable LNFLD determines if all lanes are blocked in the current curb section, and holds the vehicles at the TEST block if this occurs. Otherwise, the PH5 storage is tested to determine if any vehicles are double parked. When no double parking occurs, the land delay, LNDLY, is zero and the transaction performs a test to determine if the current curb section transfers to DCARA for the last section or DLARB for passing the next curb section.

When lane blockage occurs, the maximum number of lanes available is decremented by one and assigned to NOLAN. The delay for each vehicle is calculated as 2x3/NOLAN and placed in the variable LNDLY. Each transaction passing the current curb section is delayed by an ADVANCE V8LNDLY block. The same test and routing to DLARA ar DCARB is performed.

At DCARA, the count of vehicles as the deplaning curb is incremented by 1 and the entry time into this section is marked in PH8. Byte parameter 10 is tested for values 1, 2, or 3. The value 1 routes the transaction to DCAR5; 2 to DCAR4; and 3 enters the vehicle in the storage representing queuing at the destination curb section. Recirculating vehicle transactions have CIRCO assigned to PH4 and transfer to CTRL1.

Queuing vehicle transactions enter storage PH6, change their priority level to 12, then link on a chain numbered PH6. These will be released when corresponding greeter vehicle and passenger transactions are matched in the deplaning curb passenger logic. A transaction is released from DPL1C and routed to DCAR2, when the simulated matching takes place between the passenger and a vehicle at curbside. At the section following DCAR2, the transaction representing a previously double parked vehicle will move into the storage representing the curbside, reset its priority level to 10, relink itself on chain DPL1C and remove one transaction from the queue chain, and route the transaction to DCAR3. Vehicle transactions released from DPC1C by the deplaning curb passenger logic, representing double parked vehicles, release one transaction from the queue and transfer it to DCAR3.

At the program location DCAR3, PB7 is tested for a value zero. A non-zero value indicates that the transaction is a greeter vehicle using the enplaning curb and transfers it to ENPC8. Deplaning curb vehicle transactions continue to the next block and leave the PH6 storage entered above by queueing vehicle transaction. The number of the double parking storage associated with the queue previously departed is placed in PH6. Byte parameter 10 is assigned a value 2, signifying double parking. The transaction proceeds to the next block at DCAR4.

All double parking vehicle transactions proceed through DCAR4, where the priority level is set at 11. These transactions, and the transactions representing vehicles to be parked at curbside, enter the PH6 curb or double parking storage at DCAR5.

An attempt to unlink waiting passengers from the chain DPL2C by matching equal values of PH10 in the vehicle and greeter transactions results, if successful, in a transfer to DPLC9. If unsuccessful, the vehicle transaction is linked on chain DPL10.

The model generates a transaction to perform a policing function. This transaction will remove vehicle transactions remaining in double parking or queuing longer than a prescribed time limit. The length of time allowed in these two facilities is specified in an ADVANCE block in DLAR6. After passing DCAR6, the transaction unlinks all transactions representing double parked vehicles on the chain DPL1C if the absolute clock time is greater than the entry time into these two facilities plus 300 seconds. These vehicle transactions are transferred to DCAR8.

The policing transaction places the total number of all curb sections, enplaning and deplaning, in PH1. The APSS storage number, DPQCS, of the first deplaning curb queue is placed in PH2. At DCAR7, the program unlinks the vehicles from the PH2 chain if they have been in the queue longer than 30 seconds. This chain has the same number as the storage representing a queue at a curb section. The program increments PH2 by one and loops back to DCAR7. Because the storage and chains representing queues at curb sections were numbered continuously for deplaning and enplaning curbs by SYN statements, executing this loop PH1 times will remove all queued vehicle transactions and transfer them to DCAR9. The policing transaction is transferred back to DCAR6 to await the next enforcement time.

Double parked vehicle transactions forced to leave the chain DPL1C and transferred to DCAR8, unlink one vehicle transaction from the queue chain corresponding to the storage representing double parking. The unlinked transaction is routed to DCAR3. The previously double parked vehicle transaction proceeds to the next block DCAR9.

At DCAR9, previously queued and double parked vehicle transactions leave the PH6 storage. The priority level is dropped to 10. For vehicle transactions assigned to enplaning curb sections, the waiting time spent queued or at double parking is added to PH11. This addition is performed for this type of vehicle transaction because only queued vehicle transactions are forced to leave the enplaning curb. Double parked vehicle transactions remain at this curbside without a simulated enforcement process. Furthermore, the vehicle using the deplaning curb has a waiting time dependent upon time of arrival at the landside. Simulated waiting time for this transaction type is not entirely dependent upon landside service processes and is not entered into wating-time tables.

Vehicle transactions leaving the deplaning curb area are assigned to program location CIRCO to simulate a recirculation process. Vehicles directed away from the enplaning curb area begin recirculation at CIRC1.

8. Deplaning Curb (Pax)

The terminating deplaning passenger transactions are routed to DPLCO for subsequent matching with greeter vehicle transactions, loading into taxis or loading into buses or limos. A HELPA call to FORTM by the transaction returns with the point number of the deplaning curbside section in PH2, the landside facility number in PH7, and the value 12 in PB11. The transaction is advanced by TRUXH, the walking time from the last facility utilized. Using the selection function DPL1F, a branch to DPLC3, DPLC4 or DPLC5 is executed for transactions assigned to private vehicle, taxi, or bus/limo modes, respectively.

Transactions transferred to DPLC3 attempt to unlink waiting vehicle transactions from the chain DPL1C, even if previous matching between greeter and passenger transactions has taken place within the terminal building. This is because simulated vehicles removed from the parking facility after the terminal building greeting process and subsequent routing to the curbside, perform the same logical operations as greeter vehicle transactions proceeding directly to curbside from the airport boundary. The unlinked vehicle transaction is transferred to DPLC9 and the passenger transaction transfers to DPLC1. Unsuccessful passenger transactions are transferred to a second UNLINK block to attempt unlinking transactions representing greeters before vehicles have recirculated and parked: these are located on chain DPL3C.

Greeter transactions are transferred to DPCG2 and passenger transactions go to DPCG1. Passenger transactions unsuccessful in removing transactions from DPL3C are transferred to DPLC2.

The passenger transactions unlinking a greeter vehicle from DPL1C are held by a GATE block at DPLC1 until the vehicle transaction places the storage number of the curb section half-word savevalue PASS1, the PB10 flag in byte savevalue PASS2 and sets the logic switch PASSL. The passenger then proceeds and places PASS1 in PH6 and PASS2 in PB10. The passenger transaction resets PASSL and simulates the vehicle loading time by advancing the value DPL1V. Byte parameter 10 is tested for a value 1 to determine if the greeter vehicle will leave from the curbside. When this occurs, the storage number of the curbside is placed in TMPXF for calculating a random pull-out time using variable DPL2V. This time value is retained in PH11 for entry in the table of waiting times. The passenger transaction leaves storage PH6 and unlinks one vehicle transaction from DPL1C when the double parking storage adjacent to the curbside area just departed, matches PH6. The unlinked vehicle transaction is routed to DCAR2. The passenger transaction transfers to DPLC8.

Greeter transactions with PB10 not equal to 1, transfer to DPLC7 and leave the PH6 double parking storage. They unlink one vehicle transaction from the queue chain for vehicles awaiting entry to the double parking storage just departed.

These greeter transactions proceed to DPLC8 where they join the passenger transactions from the curbside and increment the count of vehicles on the departing road. All transactions transfer to CTRLO.

At DPLC2, those passenger transactions unsuccessful at unlinking greeter transactions from DPLC1 and DPL3C are assigned the address DPLC1 in PH4 and are linked on the chain DPL2C to await unlinking by which transactions in the deplaning curb (cars) logic.

Greeter vehicle transactions unlinked from DPL1C are transferred to DPLC9. These are held at a GATE block if PASSL has not been placed in a reset condition by a passenger transaction. When able to proceed, the vehicle transaction places the value of PH6 in PASS1 and PB10 in PASS2 to provide information to the matching passenger transaction. The transaction sets logic switch PASSL, to allow the passenger transaction at DPLC1 to proceed and is then terminated.

Transactions using the bus/limo or taxi are transferred to blocks DPLC4 or DPLC5, respectively by the function DPLC1F. At DPLC4, the count of deplaning passengers waiting for a bus or limo, DPCXH, is incremented by 1. At DPLC5, the loading time of the taxi, PDL1V, is simulated by an ADVANCE block. The taxi increments the count of vehicles on the roadway departing the airport by 1. All transactions entering these two sections are transferred to CTRLO.

A transaction representing a bus or limo is generated in the Control Section and transferred to DPLC6. At this and subsequent blocks, the vehicle transaction increments the count of vehicles on both the arriving and departing roadways. It also removes the passengers waiting for a bus or limo by assigning the halfword savevalue DPCXH a value of zero. This transaction is then terminated.

The last three areas of the Deplaning Curb Section (Pax) simulate activities of greeters intending to meet passenger at curbside but forced to recirculate and park because of congestion at the curb. At location DPCG0 the greeter transaction excutes a HELPA block. The E-operand is used as a flag to indicate that this transaction represents a greeter acting as a pedestrian and has a value 1.

The transaction assigns the point number of the curb to PH2. The landside facility number of the curbside is placed in PH7 and the value 12 is assigned to PB11. The transaction is advanced by TRVXH, the walking time from garage to curb. The greeter transaction attempts to unlink a passenger transaction from the chain DPL2C. If successful, the passenger transaction transfers to DPLG1 and the greeter transfers to DPCG2. Unsuccessful greeter transactions are placed on chain DPL3C for subsequent unlinking by passenger transactions.

At DPCG1, the passenger transactions are held by a GATE block until switch PAS5L is placed in a set condition by the greeter transaction. The greeter transaction passes through the GATE block at DPCG2; then places the number of greeters in byte savevalue PAS52 and the parking lot number in PAS53. The greeter transaction places PAS5L in a set condition and is then terminated.

After the passenger transaction proceeds past the GATE block, PAS52 is added to PB5 and PAS53 is assigned PB14. The passenger transaction resets PAS5L. A value 2 is assigned to PB6 to indicate a parking mode. The process function pointer, PB2, is reset to 1 and routing function GRCPF is assigned to PB1. This routes the passenger transaction to the parking facility. The transaction transfers to CTRLO.

9. Enplaning Curb

This program section represents the activities of the originating passenger from the vehicular approach to the enplaning curbside, through the parking and unloading process to entry into the terminal building. Private vehicles and taxis departing the curbside after separation from the passenger groups are simulated. Arrivals of buses or limos at the enplaning curb are also represented. This module contains a section to simulate recirculation of vehicles from either enplaning or deplaning curbsides.

Transactions representing originating passengers are routed to ENPCO if the simulated mode of arrival at the airport is private car or taxi, or to ENPC2 if the mode is bus or limo. At ENPCO and the next block, the count of vehicles, ARDXH, on the airport entrance road and the count of vehicles, ENPXH, proceeding to the enplaning curbside, are both incremented by 1. A HELPA call to FORTM is executed, using the airline number and the PB6 mode operates.

For the private car or taxi mode, subprogram FORTM searches for a curbside or double parking space. The curbside associated with the airline specified by the B operand is examined first, and, if necessary, all other curbside sections are inspected in a prespecified order with adjacent sections first and remote sections last. If a space is located, the GPSS storage number is assigned to PH6. The value 1 or 2 is assigned to PB10 to designate curbside or double parking, respectively. When no space is available at the two above facility types, the queue associated with the B-operand airline curb section is inspected for a space. An available space results in the assignment of the storage number representing the queue in PH6 and 3 in PB10.

Transactions representing vehicles parked at curbside or in a double parking location or in the queue are assigned the point number of the facility to PH2 and the landside facility number to PH7. Vehicle transactions unable to locate any space are forced to recirculate. These are assigned a value zero to PH5 and PH6 and a value 4 is assigned to PB10.

After returning from FORTM, the model performs logical operations to model congestion due to lane blockage. The operations performed are identical to those in the deplaning curb logic with two exceptions. First, the number of bags in PB4 is not inspected because only the enplaning curb congestion is modeled by this section. Secondly, the time spent passing through the congestion is recorded in PH11 for enplaning curb vehicle transactions but not those using the deplaning curb.

When the vehicle transaction is at the section containing the storage designated by PH6, the transaction is transferred to ENPC1. Recirculating vehicle transactions completing simulated passage of all curbside sections are also directed to ENPC1.

The subsequent logical operations of this curbside section closely parallel the Deplaning Curb (Cars) Section. Transactions representing double parked vehicles are routed to ENPC4, where their priority level is made 11, as before, and directly enter the PH6 storage curbside vehicle transaction. Queuing vehicles enter their PH6 storage, change their priority level to 12, as before, and link on the PH6 chain. Recirculating vehicle transactions are transferred to CIRC1.

Vehicles transactions in curbside or double parking storage simulate the unloading time by an ADVANCE block which holds the transaction by the amount ENP1V. A SPLIT block separates the vehicle transaction from the originating passenger transactions. The passenger transaction is tested for a preticketed status. If PB9 is zero, indicating this condition, a percentage of these transactions, defined by halfword savevalue CRBXH, enter an ADVANCE block to simulate curbside check-in. All transactions, including those representing non-preticketed and non-curbside check-in passengers are transferred to CTRLO.

The vehicle transactions are transferred to an ADVANCE block to delay the transaction for simulating empty-car parking time. The vehicle transaction with PB10 equal to 1, simulates a random pull-out time from curbside and then leaves the PH6 storage. This transaction unlinks a vehicle transaction from the queue corresponding to the departed curbside, and transfers it to ENPC8. The parked vehicle transaction is transferred to ENPC9.

Double parked vehicle transactions transfer to ENPC7 and immediately leave the PH6 storage. These also unlink one vehicle transaction from the queue adjacent to the double parking storage, transfer it to ENPC8 and proceed to ENCP9.

At ENPC9, the count of vehicles on the departing road is incremented by 1. If no well-wishers are simulated to enter the terminal building, PB5 equals PB13 and the vehicle transaction is terminated. When PB5 and PB13 are unequal, the vehicle transaction executes a HELPA call to FORTM to determine the parking facility number to be entered. It then increments the entry count of the parking facility and is terminated.

Queued vehicle transactions released from the PH6 chain by previously parked vehicle transactions and routed to ENPC8, test PB7 for a value of 1 to determine if they represent vehicles using the deplaning curbside. Deplaning curb vehicle transactions are transferred to DCAR3. The enplaning curbside vehicle transactions proceed to the next block and add the waiting time spent in queueing to PH11. These transactions leave the PH6 storage which represented the queue, and assign the storage number of the curbside in the same curb section to PH6 and assign one to PB10. The PH6 curbside storage is tested for occupancy and, if not full, the vehicle transaction enters it at ENPC5. When the curbside storage is full, the corresponding double parking storage number is assigned to PH6.

Byte parameter 10 is assigned the value 2, the priority level of the transaction is changed to 11 and the transaction enters the PH6 storage at ENPC5. The vehicle transaction is then terminated. Enplaning passenger transactions assigned to the bus/limo mode are routed to ENPC2. A HELPA call to FORTM returns with the loadside facility number of the bus stop for the airline designated by the B-operand placed in PH6 and the point number of the facility is PH2. The transaction has the program address ENPC3 assigned to PH4 and is linked on the chain EBUSC to await arrival of the simulated bus/limo at the enplaning curbside. After unlinking by the bus/limo transaction, the enplaning passenger transaction is transferred to CTRLO.

The transaction generated by the Control Section representing the bus/limo is routed to ENPC6. At this location the transaction increments the halfword savevalue ARDXH, the count of vehicles on the approach roadway. An ADVANCE block simulates the unloading time of the passengers from the vehicle. This transaction unlinks all passenger transactions placed on EBUSC by the instructions following ENPC2 and transfers them to CTRL1. The count of vehicles on the departing road, DRDXH, is incremented by 1 and the bus/limo transaction is terminated.

The recirculation roadway logic is also contained in this section. Vehicle transactions assigned to the Deplaning Curb (Cars) logic are transferred to CIRCO for recirculation.

An ADVANCE block simulates the vehicle recirculation time and the count of vehicles proceeding from the deplaning curb to the recirculation roadway, RCDXH, is incremented by 1. A test of PB12 for zero, determines if the transaction was routed to perform the matching process with the terminating passenger transaction only at the curbside. A percentage of these transactions specified by halfword savevalue CPKXH are selected to enter the parking facility. The PB6 mode is assigned a value 2 and the routing function GRECF is assigned to this transaction. The transaction is then transferred to CTRLO.

All other deplaning curb recirculating transactions, those with PB12 non-zero and those not parking, are routed to DCAR1 to repeat the deplaning curb logic instructions. Recirculating vehicle transactions transferred from the enplaning curb logic to CIRC1 are advanced by the value CIRCV to simulate recirculation time. The count of these transactions, RCEXH, is incremented by 1 and the transactions then transferred to ENPCR to return to the enplaning road logic.

10. Entrance

Originating enplaning transactions and greeter transactions with non-zero PB12 values are routed to ENTRO to simulate entry into the terminal building. A HELPA call to FORTM returns with the point number in P42, of the entrance nearest to the facility immediately departed by the transaction. The transaction advances TRVXH, the amount of walking time from the last point to the entrance. The Boolean variable DPDIN is tested for a value of 1 to determine if the transaction will increment the terminal entry count on the deplaning level. When this occurs, the transaction represents a greeter proceeding to bag claim. The halfword savevalue DPLIN is incremented by the value in PB5. The transaction is transferred to increment the total count of all entrances in halfword savevalue ENDUR and is then transferred to CTRLO.

All other transactions are assumed to enter on the enplaning level. These increment the halfword savevalue ENDIN by the amount of PB5 and also increment ENDOR. These transactions also transfer to CTRLO.

11. Exit

This module begins at program location EXITO. A HELPA call to FORTM at this location returns with the point number in PH2, to the exit nearest the facility immediately departed from. The walking time from the last facility to the exit is simulated by an ADVANCE block.

The Boolean variable DDOUT is used to determine if this transaction represents a domestic, international or commuter passenger group with baggage. These transactions increment the halfword savevalue DPOUT, the count of passengers and visitors exiting the terminal from the deplaning level, by the value in PB5. This transaction type also increments EXDOR by the value in PB5.

All other transactions are assumed to represent passengers or visitors exiting through the enplaning level. These increment the count EPOUT, for this level, and increment the total exit count, EXDOR, both by the value in PB5. All transactions entering this section transfer to CTRLO.

12. Gate (Enplaning pax)

This module simulates the processing of enplaning passengers at gate counters, matching of greeters with passengers and separation of well-wishers from passengers. A subsection simulates the aircraft boarding process.

Passenger and greeter transactions first enter this section at GATEO. At this location, a HELPA block is executed. This results in assignment of the point number of the gate facility to PH2 and the GPSS storage number of the gate to PH5. The landside facility number, identical to the gate number, is assigned to PH7 and PB11 is given a value of 1.

The walking time from the last facility utilized is simulated by an ADVANCE block. Byte parameter 13 is tested for a value zero, to determine if the transaction represents a greeter. Non-greeters are transferred to GATE7. Greeter transactions attempt to unlink a passenger transaction from the chain GREGC. The transaction removed is routed to DEP29 in the Deplaning Passenger Logic Section. The greeter transaction proceeds to GATE8, and, after passing a CATE block, assigns the number of greeters in PB5 to byte savevalue PAS32 and the parking facility number is PB14 to byte switch PAS3L in a set condition. PB6 is tested for a value 1, to determine if the greeter transaction is required to simulate removal of the vehicle from the parking facility and perform a subsequent pickup of the terminating passenger transaction at curbside. A value of 1 in PB6, indicating the routing of passengers and greeters to the parking garage, requires only the use of the passenger group transaction, and the greeter transaction is terminated.

Greeter transactions with PB6 unequal to 1 have zero assigned to PB5 and GRTO3 assigned to PH4. These transfers to CTRL1 to proceed to the parking facility. Greeter transactions unable to unlink a passenger transaction from GREGC are transferred to GAT10. At this location, they are assigned the address GATE8 in PH4 and link on the chain GREGC for later release by a deplaning passenger transaction in the Deplaning Passenger Logic Section.

At GATE7, enplaning passenger transactions enter a TEST block to determine if well-wishers are included in the simulated passenger group. Transactions without well-wishers represent transfer to GATE3. Those transactions including a representation of well-wishers enter a SPLIT block assign the value is PB13 to PB5 and transfer to GATE3. The copy transaction simulates the well-wishers. The number of passengers in PB13 of this transaction is simulated from PB5 and PB13 is made equal to zero. A well-wisher routing function, WWG1F, is assigned to PB1. The pointer PB2 is reset to 1 and the transaction transfers to CTRLO to begin the routing process out of terminal area.

Passenger transactions performing the gate check-in operations proceed from GATE3 to GATE1 when they enter a QUEUE block. The total number of passengers to reach gate counters, represented by halfword savevalue GATXH, is incremented by PB13. The simulated waiting time in a gate queue is recorded in PH11. The transaction enters the PH5 storage, simulates the service time by an ADVANCE block then leaves the PH5 storage.

A GATE block directs transactions to GATE2 unless the flight transaction has placed the logic switch PH5 in a SET position. This setting operation is performed at the start of the simulated boarding time. Transactions transferred to GATE2 increment MH1(PH1,12) by PB13, then transfer to CTRLO for subsequent termination. Those transactions arriving at the GATE block after boarding time decrement the occupancy count at the point number and then transfer to CTRLO for termination.

Flight transactions for departing flights generated in the Deplaning Passenger Logic Section are transferred to GATE9 to simulate the boarding process. Each flight transaction is generated when the absolute clock time is one hour before the scheduled departure time. At GATE9, the transaction is held until absolute clock time equals the flight time minus a boarding interval, BDTXH. The GPSS storage number of the gate is assigned to PH5 and the PH5 logic switch is placed in a set position, allowing passenger transactions executing the LEAVE block at GATE6 to proceed directly to CTRLO for termination.

The flight transaction is held at an ADVANCE block until the absolute clock equals flight time. The transaction then resets the PH5 logic switch and resets MH1(PH1,12) to zero, allowing this element to record the numbers of simulated passengers missing the flight. The flight transaction is then terminated.

13. Ground Transportation (Misc.)

This module is accessed by passenger and visitor transactions. Self-driver, deplaning passenger transactions are routed to GRTOO; enplaning passengers proceed to GRTO1. Greeter transactions simulating initial entry to the terminal area proceed to GRTO2. The greeter transaction designated to proceed from parking to curbside after matching respective passenger transactions at gate, bag claim, security or lobby, transfer to GRTO3. Well-wishers, separated from the passenger group at the gate or security, are routed to GRTOO.

Deplaning passenger transactions routed to GRTOO execute a HELPA call to FORTM. For this transaction type, the parking facility number assigned in GPSS to PB14 is used to obtain other parameter values. The values assigned by FORTM are the point number of the parking facility in PH2, the storage number in PH5, the landside facility number in PH7, and the value 10 in PB11. The transaction is advanced by the amount TRUXH, the walking time from the last facility to the parking facility. Following this delay, the transaction transfers to CTRL8 for direct routing to the parking garage exit.

Enplaning passenger transactions entering the terminal area for parking are routed to GRTO1. At this program location, a HELPA call to FORTM is executed and returns with the assignments made to the same parameters as deplaning passenger transactions at GRTO.

Greeter transactions entering the terminal area for entrance to parking facilities are routed to GRTO2. At this block, and the three following blocks, the assignment to parameters and incrementing of counts are exactly the same as those of the enplaning passenger at the subsection starting at GRTO1.

Greeter transactions assigned to remove vehicles from the parking facility after passenger matching for proceeding to curbside are routed from gate, bagclaim, security or lobby to GRTO3. At this program location these transactions execute a HELPA call to FORTM. All parameter assignments are identical to those described for transactions entering GRTOO.

After return from FORTM, an ADVANCE block simulates the walking time from the last facility to the parking facility. The greeter transaction then simulates a vehicle and enters the PH5 queue, if necessary, at the parking lot exit. The service time at the parking facility payment booth is simulated. The parking lot inventory is decremented by 1 and the count of vehicles proceeding from parking lot to curbside is incremented by 1. The car is routed to the Deplaning Curb Logic Section by assigning DCAR1 to PH4 and transferring the transaction to CTRL1.

14. Parking (Dept. Pax-Cars)

The deplaning passengers transactions previously routed to GRTOO are routed to PARKO, the first program location of this section. At this point, the PH5 queue is entered and the time of entrance is recorded. After departing the queue to enter service at the vehicle parking payment booth, the time spent queuing is recorded in PH11. The service time at the booth is simulated, then the vehicle inventory in the facility is decremented by 1 and the count of vehicles on the departing road is incremented by 1. The transaction is transferred to CTRLO.

15. Rent-A-Car

Deplaning passengers transactions selected for the car rental mode are transferred to RCARO to begin the simulation of processing performed at the car rental counter. Byte parameter 6 is tested for a value 3, a flag designating usage of this mode. Transactions not utilizing car rental are immediately transferred to CTRLO. Transactions proceeding to the next block select a car rental agency from the selection function RCA2F and assign the agency number to PB10. A HELPA call to FORTM determines the agency counter nearest to the facility immediately departed by the passenger transaction. FORTM returns with the point number of this counter in PH2, the storage number in PH5, the landside facility number in PH7 and the value 11 in PB11. The walking time from the last point to the car rental counter is simulated by TRVXH. The transaction enters a QUEUE block, and subsequently the PH5 storage. The waiting time is recorded in PH11 and the processing time is simulated by RCA1V. This transaction proceeds to CTRLO for further processing.

Deplaning passenger transactions using the car rental mode are assigned to RCAR9 by the ground transportation selection function CTR1F. These simulate passengers who have completed processing inside the terminal building and are proceeding to a location to obtain the vehicle. This logic assumes that the passenger picks up the vehicle at an agency parking facility.

At RCAR9, a HELPA block is executed. FORTM determines the number of the agency facility and returns with the point number of this facility in PH2, the storage number of the facility in PH5, the landside facility number in PH7 and the value 10 in PB11. The travel time to this facility is advanced. The transaction then increments the count of vehicles on the departing road by 1 and transfers to CTRLO.

Enplaning passenger transactions arriving at the airport landside who are proceeding to return rental vehicles are routed to RCAR8. The HELPA call to FORTM returns with the same assignments as for transactions using RCAR9. The arriving roadway count, ARDXH, is incremented by 1, then the transaction is transferred to CTRLO.

16. Security

All transactions simulating entry to concourses and subsequent movement to gate facilities pass through the Security Module. At SECUO, a HELPA call is made to determine the security facility for the destination gate. FORTM returns with the security point number assigned to PH2, the storage number in PH5, the landside facility number in PH7 and the value three in PB12. The transactions are advanced by the travel time between security and the last facility used. A test comparing PB5, the number in the party, with PB13, the number of passengers in the party, determines if the transaction represents any simulated well-wishers. When PB5 equals PB13 there are no well-wishers, and the transaction is transferred to SECU3. When PB5 does not equal PB13 well-wishers are present, and the transaction continues to the next sequential statement. A statistical transfer is made, using halfword savevalue WWXGH. If the result of the statistical transfer requires separation of well-wishers and passengers at security, the transaction goes to the next sequential block. Otherwise the transaction is transferred to SECU3. At the next sequential block, the transaction is split once. The copy transaction, representing well-wishers, is transferred to the third sequential block. For the parent transaction, PB5, the number in the party, is made equal to PB13, the number of passengers in the group, thereby removing well-wishers from the party count. The passenger transaction is then transferred to SECU3.

The well-wisher transaction removes the number of passengers in PB5 by subtracting the number in PB13 from it, then assigns the value zero to PB13. PB1 is assigned WWS1F, the well-wisher leaving from security process function. The process function pointing PB2 is reset to 1, and the well-wisher copy transaction is transferred to CTRLO.

At SECU3, the PH2 halfword savevalue, representing occupancy at the security point, is increased by PB5, the number in the party. At SECU1 the PH5 queue is entered with an entry count of PB5, then the transaction is marked to record waiting time. The PH5 storage representing security service is entered and the waiting time is added to PH11. At SECU2 the passenger transaction is advanced by V$SEC1V, the security check time.

The occupancy count at the point, which is kept in the halfword savevalue whose number is PH2, is decremented by PB5, the number in the party. The matrix savevalue, MH12(SECNN,1) is incremented by PB5. This matrix is the count of simulated passengers and visitors processed through the security facility. SECNN is the number of the security facility.

17. Concessions

A number of simulated transfer and transit passengers, selected by random draw in the Deplaning Passenger Logic, are sent to their section to spend part of the waiting time for their departing flight.

Concessions are located in the lobby and on the concourses, and the routing to either facility is also specified during the selection process. Transactions are routed to LOBCO for lobby concessions and to CONCO for concourse concessions. Program Logic for both facilities is identical.

At LOBCO or CONCO a HELPA FORTM call is made, with a flag set for lobby or concourse concession, respectively. The FORTM subprogram calculates the travel time to the concession and assigns the time spent at concession to PH5. The transaction is advanced by TRUXH, the travel time from the last facility. The savevalue whose number is the same as the point number of the concession is increased by PB5, the number in the party. The transaction is next advanced by PH5, the simulated time spent at the concession. The savevalue, which keeps a count of the congestion at their point, is decremented by PB5 to indicate the transaction has left the concession, and the transaction is then transferred to CTRLO.

Transfer Flights

This module establishes and maintains a table in MH5 containing the MH1 row number of simulated departing flights able to accept transfer passenger transactions. The variable TPAX of the input flight schedule specifies the initial number of transfer passenger transactions accepted by the flight. During the operation of the simulation, the number of transactions is decremented by the transfer passenger logic in FORTM. Flights are added to or deleted from the MH5 table as simulation time progresses. If an insufficient number of transfer passenger transactions were assigned to the flight prior to deletion from the transfer flight table, this module generates transactions to complete the assigned number of transfers.

A single transaction is split from the initial transaction generated by the Deplaning Passenger Logic Section and transferred to XFLTO. At XFLTO the domestic transfer process function, TDP1F, is assigned to PB1. Byte parameter 2, the process function pointer; is assigned a value 2 to place these transfer passengers at security; Byte parameters 5, the total number in this passenger groups; and 13, the number of passengers in the group, are assigned the value 1. A HELPA FORTM call is made, with a zero value D-operand used as a flag, to initialize the transfer flight table in MH5. At this initial pass, FORTM arranges the MH1 row numbers of simulated departing flights accepting transfer passenger transactions in MH5, in chronological order. The simulated departure times of these flights must be between 30 minutes and 2 hours, or between the FORTRAN input limits DELETE and ADD respectively, after the simulation start time. FORTM returns with the MH1 row number of the first flight departing after the ADD or two hour time limit in PH1. If the last row of the flight table matrix was reached during this initialization process, the number of the next succeeding row of MH1 is assigned to PH1.

The next block tests MH1(PH1, 1) for a value less than zero. If the end of the flight table matrix was reached in FORTM, the MH1 (PH1,1) value is minus 1. When this occurs, the transaction proceeds to the next block to determine if any flight accepting transfers was located by FORTM with subsequent placement of the MH1 row numbers in MH5. The MH5 (1,1) element is tested for a zero value to determine if an MH1 row number is present. When MH5 (1,1) is zero, the transaction is terminated If MH5(1,1) is non-zero, the transaction transfers to XFLT3 to delete flight numbers from the transfer flight table as simulated time progresses.

When the returning value of PH1 does not indicate that the last row of MH1 was reached, the transfer flight transaction transfers to a SPLIT block. The parent transaction proceeds to the next succeeding block to attempt to add the PH1 value to the MH5 flight table. The copy transaction transfers to

XFLT3 to attempt to delete the MH5(1,1) element.

When adding the current PH1 and subsequent MH1 row number values to MH5, an initial test is made of MH1 (PH1,11) for a value greater than zero to determine if simulated transfer passengers were input for the flight. The transaction proceeds to the next block, at location XFLT1, when MH1(PH1,11) is greater than zero, otherwise it is transferred to XFLT2. An ADVANCE block at XFLT1 delays movement of the transaction until absolute clock time equals flight time minus the ADD time. Following this delay, HELPA FORTM call is executed with a flag value of 2 to implement addition to PH1 to the MH5 table. Upon return from FORTM, PH1 is incremented by 1 at XFLT2 and the test of MH1(PH1,1) for the end of the table is again performed. If the test indicates the end of the table, the transaction is terminated, otherwise it is transferred to XFLT5. At this location, the Boolean variable XFL1B is tested to determine if the MH1 row number represents a departing flight and has been assigned to accept transfer passenger transactions. When XFL1B is true the transaction proceeds to the next block for transfer back to XFLT1. If it is false, the transaction transfers immediately to XFLT2.

The deletion process begins at program location XFLT3. The MH5(1,1) element is tested for a value greater than zero. If transfer flights are available, as indicated by a non-zero value of MH5(1,1) the value of this element is assigned to PH1, otherwise the transaction is transferred to XFLT9. After the above assignment to PH1 is performed, an ADVANCE block at XFLT7 delays the transaction until the absolute clock time equals flight time minus DELETE time. The gate number of the flight is assigned from MH1(PH1,9) to PH5 and flight type from MH1(PH1,7) to PB3. A HELPA call to FORTM with a flag value 3 is executed to determine the point number of the ticket counter corresponding to the airline number in MH1(PH1,3) of the flight. Transactions are then generated to create remaining transfer passenger assignments if the flight was not filled by the Deplaning Passenger Logic Section. The number of transactions required to fill the flight is contained in MH1(PH1,11). A SPLIT block creates these transfer passenger transactions and transfers them to XFLT8 immediately before deletion of the row number from MH5.

A HELPA call with a flag value of 1, deletes the MH1 row number contained in PH1 from MH5. Upon return from FORTM, the PH1 value is incremented by one at XFLT4. The next block, located at XFLT9, tests MH1(PH1,1) for a negative value to determine if the end of the flight schedule matrix has been reached. The negative value terminates the transaction, otherwise the transaction transfers to XFLT6.

At XFLT6 the Boolean variable XFL1B is again tested. If XFL1B is true the transaction proceeds to the next block for transfers to XFLT7. If false, the transaction transfers to XFLT4 to increment PH1.

Transfer passenger transactions routed to XFLT8 are advanced a random delay time between 0 and 15 minutes before proceeding to the next block. After this delay, they transfer to CTRLO to proceed from ticket counter point to security.

Change Card Reader

A single transaction, used to perform storage capacity changes is generated, with a priority level of 120. At CHGOO, this transaction performs a HELPA call to FORTM. The initial call by this transaction reads the first change card and assigns the time difference between the current clock and time of the first service change to halfword savevalue CHGXF.

The transaction tests the halfword savevalue NSCXH, the number of storage changes designated by the CHANGE data card, for a value greater than zero. For the initial pass, NSCXH is zero, and the transaction transfers to an ADVANCE block. When the clock time equals the change time, the transaction proceeds to the next block and tests NSCXH for zero. Again, at the initial pass NSCXH is zero and the transaction transfers to CHGOO.

The second and subsequent HELPA calls to FORTM at CHGOO assign GPSS storage numbers to MH7(I+30,1). Subscript I refers to the Ith storage designated for a capacity change on the CHANGE input data card. The total number of storage to be changed is assigned to NSCXH. This information is transferred from the data card read on the previous HELPA call. Thus, the data from the initial FORTM call is assigned on the second call. FORTM then reads the next input CHANGE card, assigns the time interval between the current time and the next change to CHGXF and returns to the GPSS main program.

The TEST block examines NSCXH for a value greater than zero and the transaction proceeds to the next block for this condition. PH1 is assigned a value of minus 1. The transaction is then split NSCXH times and each copy transaction is sent to block CHG01. Sequence numbers of these transactions are assigned to PH1. When the transaction is split, the value of PH1 is incremented by 1, producing the value zero in PH1 of the parent transaction and sequence numbers beginning with 1 in PH1 of the copy transactions. These sequence numbers will be used to address elements in MH7.

The parent transaction proceeds to the next block where it is delayed by the value CHGXF. This block is also the destination of the transaction if NSCXH is zero before the splitting process. After leaving the ADVANCE block, a TEST block holds the transaction until NSCXH is decremented to zero by the copy transactions. Thus, if NSCXH is greater than zero, matrix MH7 is still used by transactions generated by the previous change, and the parent transaction is held at the TEST block until MH7 is available. The parent transaction then returns to CHG00 to process the next change.

At CHG01, the destination of the sequenced copy transactions, PH2 is assigned the value in PH1. The halfword savevalue SAVXH is assigned the value of the sequence number in PH1 in order to save the pointer to the MH7 row. Halfword parameter 2 has the value 30 added to its current contents. The number of the storage to be changed and the storage capacity information are next assigned to PH1 and PH2 respectively, from MH7(PH1,1) and MH7(PH1+30,1). Using the halfword savevalue SAVXH, the pointer to the storage number in MH7, the MH7(SAVXH,1) element, is reset to zero. The capacity information element at MH7(SAVXH+30,1) is also assigned the value zero. Halfword savevalue NSCXH is next decremented by one to indicate that the number of transactions using MH7 has decreased by 1. When the new capacity of the PH1 storage equals or exceeds the current contents, PH2 contains the value zero. PH2 is tested for zero and the transaction continues to the next block for this condition. Because the remaining capacity of the PH1 storage has been reset in FORTM, to the difference between the new capacity and current contents, the storage may change status from full to not full. Transactions waiting to enter a storage which was full before the FORTM reset are on a delay chain not scanned by GPSS until a transaction in that storage executes a LEAVE block. In order to activate these waiting transactions immediately, the transaction performing this change will attempt to enter and leave the storage. A GATE SNF block allows the transaction to enter the storage if it is not full. It will then execute a LEAVE block for the PH1 storage and be terminated. If the storage is full, transactions will undergo the normal wait, and the changing transaction will be terminated immediately. If the storage is entered, FORTM adjusts the count of transactions to discount the changing transaction. When PH2 is not zero, the new storage capacity is below the current contents and must be lowered. The TEST block transfers the transaction to CHGO2.

At the block labeled CHGO2, the location that the change transaction is transferred to if the storage capacity is to be lowered below the current contents, the change transaction is held at a GATE SNF block until another transaction has left the PH1 storage, and therefore the storage is no longer full. After passing the GATE block, a HELPA FORTM, call is made to lower the storage capacity to the current contents. If the new capacity specified on the data card equals the current contents, the FORTRAN program sets a halfword savevalue SLCXH, used as a flag, to 1. The value one is SLCXH is next tested for in the GPSS program. If SLCXH is 1 the storage lowering is complete. The flag SLCXH will be reset to zero and the transaction is terminated. If the current contents exceed the new capacity, the number of available units of storage is set to zero and SLCXH remains zero. Because the flag SLCXH is not set to 1, more lowering of the storage capacity is needed. The change transaction is transferred back to CHGO2 to wait until another transaction leaves the storage.

Timer Section

A single transaction with priority level 127 and one halfword parameter is generated to initiate and, subsequently, to terminate the activity of the Main Program. An advance of XH$CLKXH is done to define CLKXH as a halfword savevalue. XH$CLKXH has the value zero at simulation start, and consequently no actual advance of the simulation time is done. The clock increment savevalue, INCXH, is set to 60 to represent 60 seconds of simulation time. A HELPC call to subroutine CLINK initiates linking of this program and the supporting FORTRAN subroutine FORTM. The next block is a HELPA call to FORTM which completes the linking. After execution of these statements, any HELPA call to FORTM appearing in the Main Program will operate with two way communication between the two programs. FORTM also reads the input data and places the information in GPSS main program matrices. Any errors detected by FORTM during the initalization and input phase are indicated by setting PH1 of this transaction to a value greater than zero. Upon return from FORTM, PH1 is tested for a value greater than zero. For this condition, the transaction splits repeatedly and goes to a TERMINATE PH1 block in order to guarantee termination.

When PH1 is zero, the transaction is transferred to a SPLIT block. The parent transaction is held at a GATE LS block to provide a method of terminating activity when errors accumulated by FORTM exceed a specified number. If this should occur, logic switch JOBLS is placed in a set condition and this transaction proceeds to a SPLIT block at STOP1. Repeated copies activate a TERMINATE 100 block to force simulation termination.

The copy generated before the GATE LS block is again split. This parent is advanced by the variable V$INC1V, which is one less than XH$INCXH, the clock increment time. This compensates for the GPSS simulation starting time which is 1 instead of zero. A HELPA call to FORTM is performed to update the clock time in the FORTRAN subprogram by the amount INCXH. The clock time, CLKXH, is kept in a 24-hour format showing hours and minutes. After an advance by the clock increment XH$INCXH, the transaction is transferred back to the HELPA FORTM, 21 block to again update CLKXH. This procedure of advancing by XH$INCXH and then making a HELPA FORTM, 21 call is continued until the simulation ends.

Halfword PH1 of the copy transaction is assigned the value of 127 which will be used as a loop counter. The function RANDF is used to place a random number between zero and 1 in the matrix element ML1(1,PH1). PH1 is then decremented by 1 by the LOOP block and the next location in the ML1 matrix is filled with a random number. This process is continued until all 127 locations in ML1 are filled with random numbers.

The priority of this timer transaction is set to 126, which is 1 less than the priority of the transaction updating the clock. This will allow the clock to be updated at the last time increment before the run is terminated. The timer transaction is then advanced to the end of run time, and an HELPA FORTM,20 call provides the printing of FORTRAN formatted summary reports. A selected list of the regular GPSS output is also printed out. A TERMINATE block with an A-operand of 1 causes the simulation run to be terminated.

Another single transaction is generated 11,700 seconds after the simulation has started. This transaction entry block is HELPA FORTM,22 to print-out flow and queue length information. The transaction is then advanced by 300 seconds and transfers back to the HELPA FORTM,22 block. This activity is continued for the rest of the simulation so that information is printed out every five minutes of simulated time.

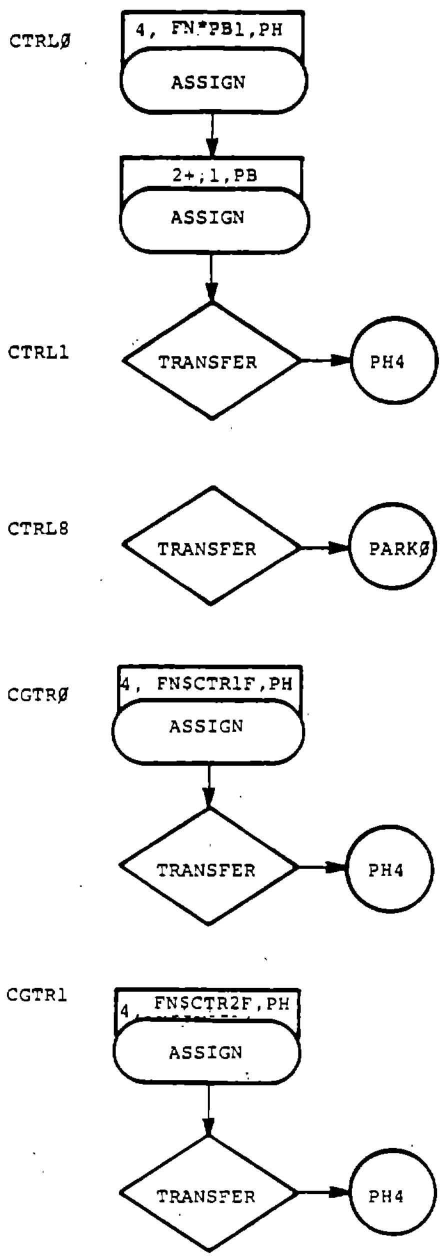
Another GENERATE block generates one transaction 3 hours and 25 minutes after the start of the simulation. This transaction then encounters a TERMINATE 1 block and stops the simulation. This procedure is used in conjunction with the RESET command in order to clear out the simulation statistics accumulated during the start up time. If this feature is not wanted then both the GENERATE block and the RESET block should be commented out.

The final GENERATE block produces a single transaction which increments the value of byte savevalue PXTBN by 1. PXTBH will be used as the number of the table providing hourly frequency distributions of simulated landside waiting time. The time is advanced one hour by an ADVANCE block, and the transaction is transferred back to the block that increments byte savevalue PXTBN by 1. This procedure is continued until the end of the simulation.

APPENDIX A-2

FLOW CHARTS

CONTROL



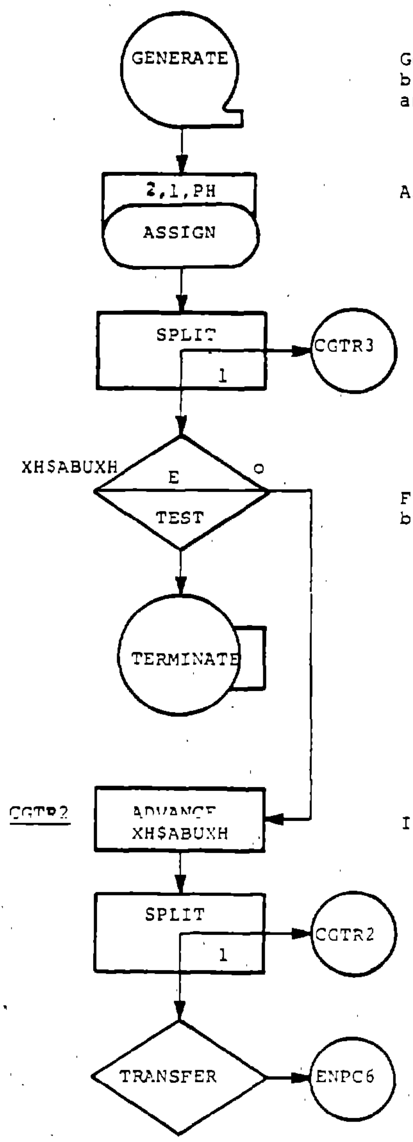
Address of next operation

Increment process function pointer

Handles routing of deplaning pax to parking lots

Route deplaning pax by ground transportation mode

Route enplaning pax by ground transportation mode

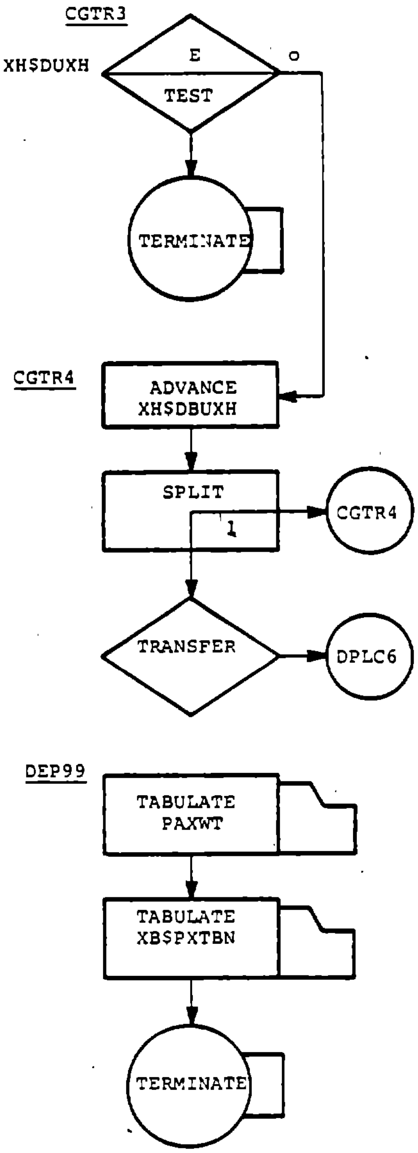


Generate Transaction for bus/limo service to enplaning and deplaning curb

Assign a location

Fall through if no arriving bus simulated

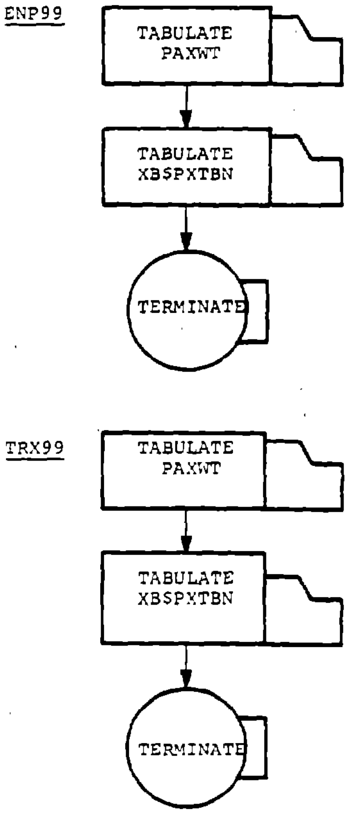
Interval between arriving buses



Fall through if no deplaning bus simulated

Interval between departing buses

Deplaning pax waiting time



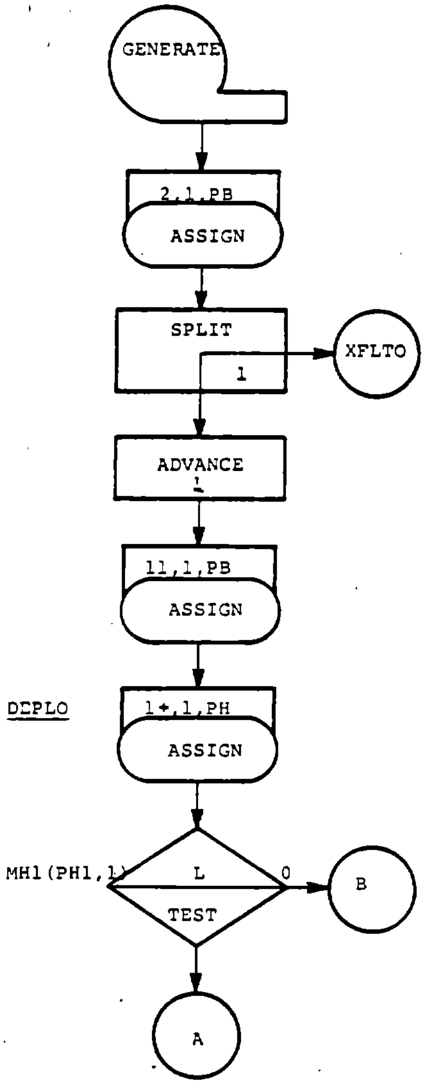
Enplaning pax waiting time

Enplaning pax waiting time by hour

Transfer pax waiting time

Transfer pax waiting time by hour

DEPLANING PASSENGER LOGIC



Generate one transaction for deplaning pax logic

Initialize Process Function Pointer

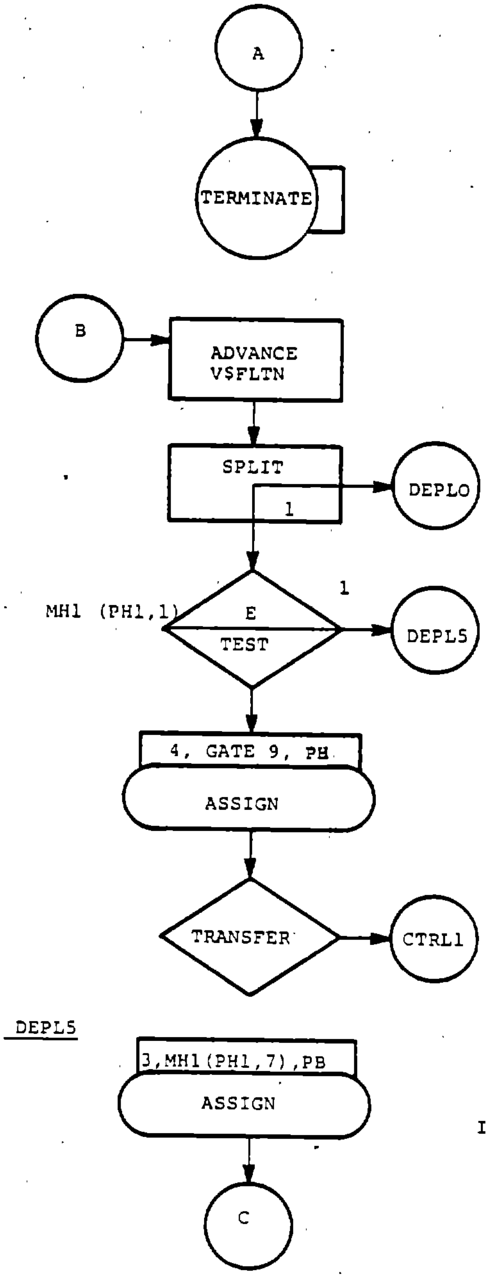
Copy to transfer flight logic

Assure that transfer flight table gets built

Deplaning pax at 'GATE'

Flight Table (MH1) row subscript

A value of -1 in MH1 (PH1,1) indicates end of flight table



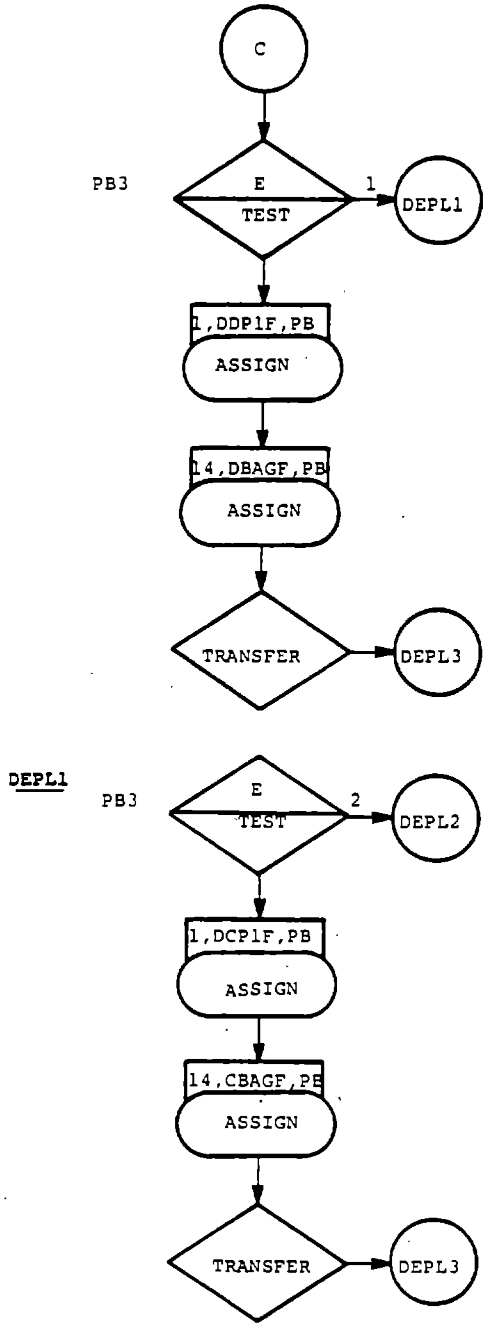
Delay to 1 hour before
scheduled arrival

Create next transaction
for flight table

Branch for arriving
flights

For gate logic to start
boarding

Type of flight;
Domestic = 1
Commuter = 2
International= 3



Tests for domestic flight

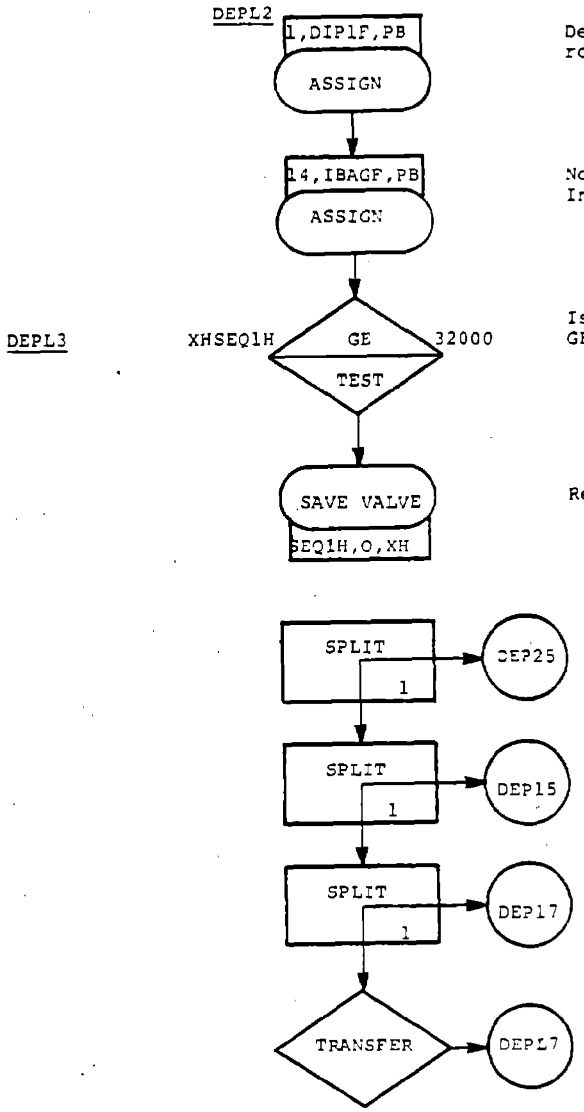
Deplaning domestic route function

No. of BAGS function-Domestic flight

Test for commuter flight

Deplaning commuter route function

No. of Bags Function-Commuter flight



Deplaning international route function

No. of Bags Function-International Flight

Is transaction Sequence No. GE 32000

Reset transaction sequence no.

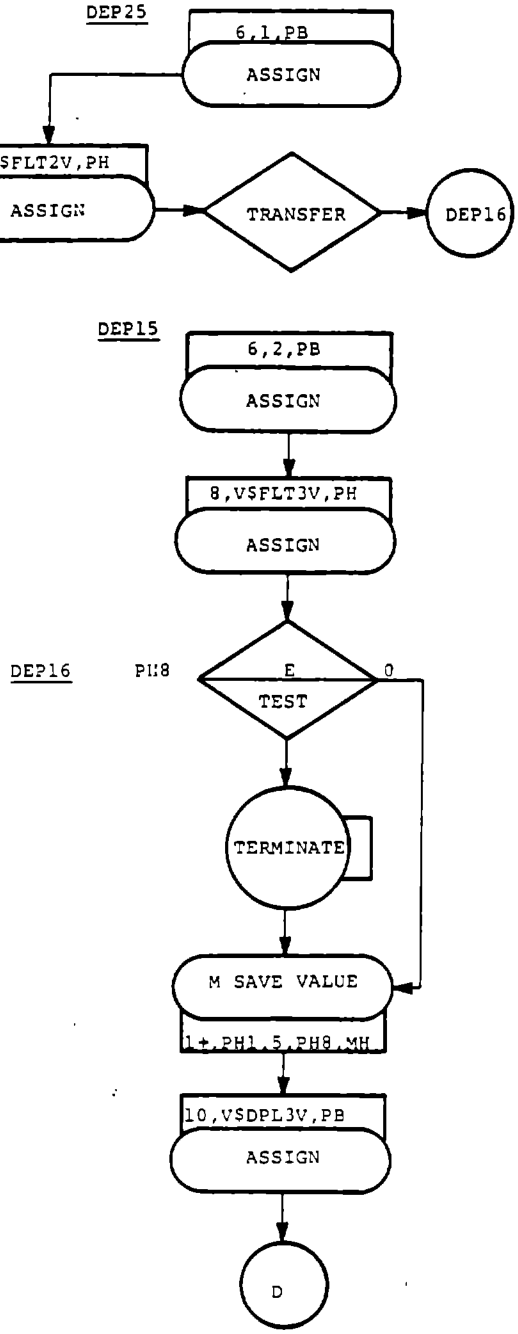
% of greeters meeting at curb

% of parking greeters proceeding to curb

To curb with greeters logic

To park with greeters logic

To curb without greeters logic



Mode = Park + curb

No. of parking greeters proceeding to curb

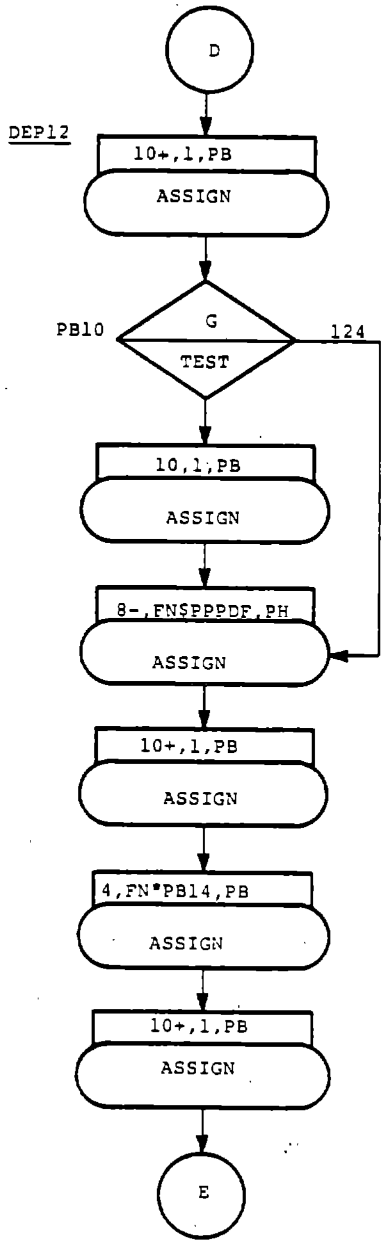
Mode = Park

No. of pax to be greeted

Fall through if no greeters

Add to total met

Starting point in Random number table



Increment random number
table pointer

Is pointer at end of table

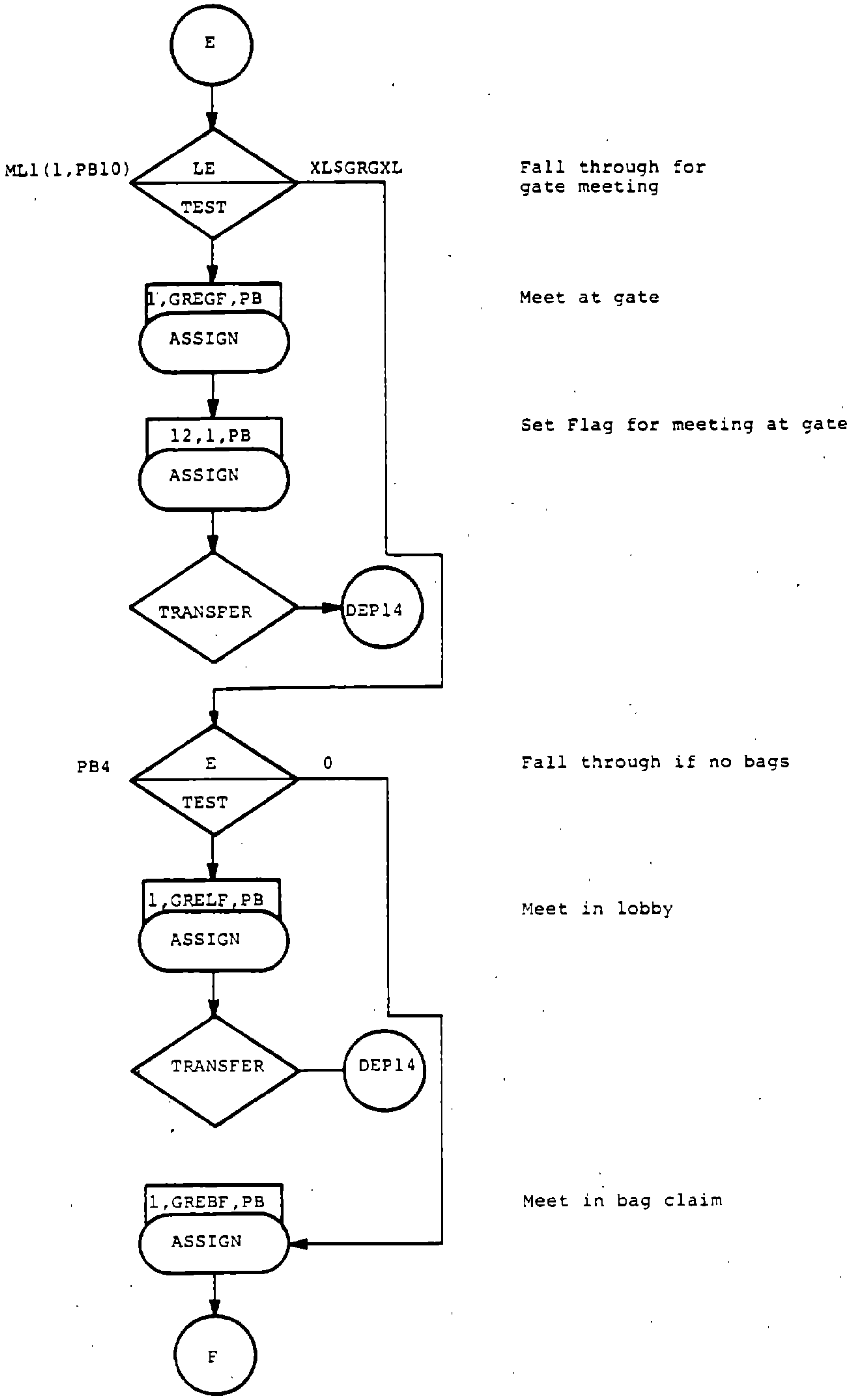
Reset pointer to one

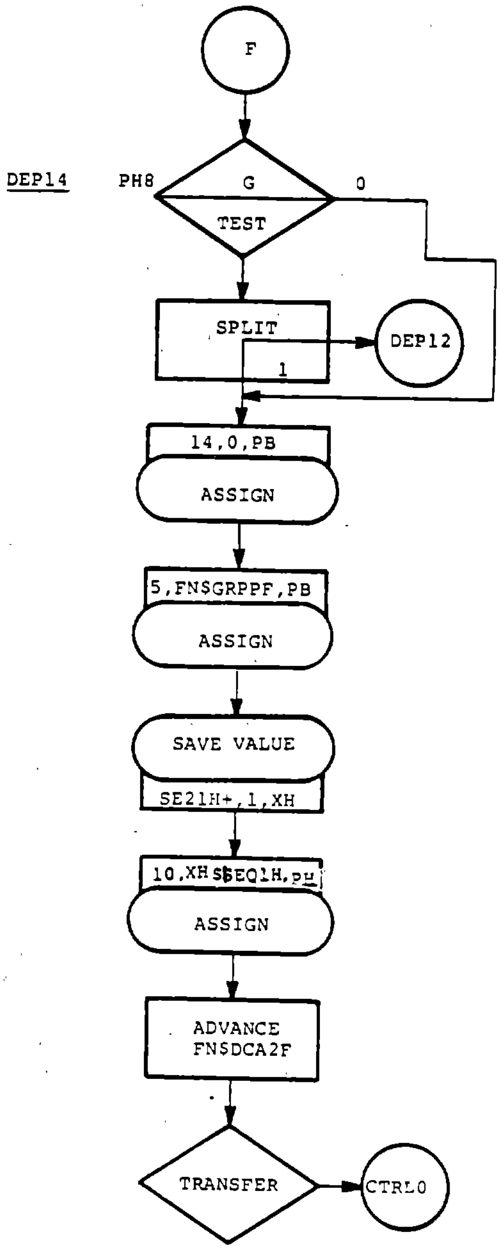
Subtract no. of pax from total

Increment pointer

Compute no. of bags

Increment pointer


E
ML1(1,PB10)
LE
TEST
XL$GRGXL
Fall through for gate meeting
1,GREGF,PB
ASSIGN
Meet at gate
12,1,PB
ASSIGN
Set Flag for meeting at gate
TRANSFER
DEP14
PB4
E
TEST
0
Fall through if no bags
1,GRELF,PB
ASSIGN
Meet in lobby
TRANSFER
DEP14
1,GREBF,PB
ASSIGN
Meet in bag claim
F



Fall through if
need more parties

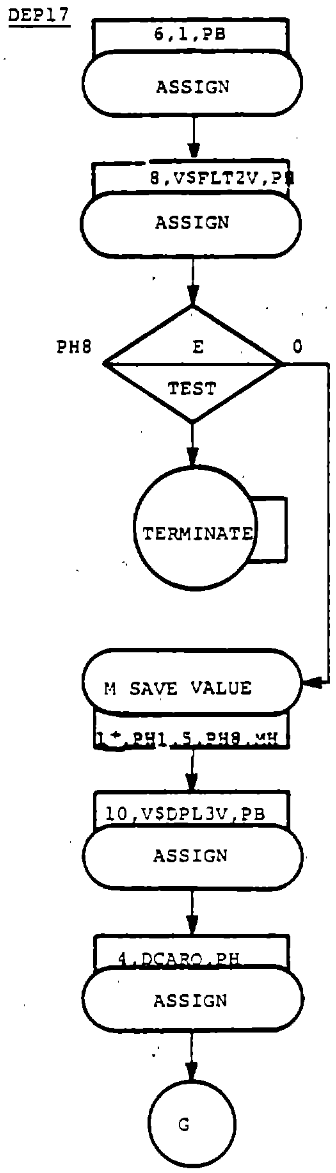
Reset PB14 for later use

No. of greeters

Increment transaction
Counter

Transaction sequence no.

Advance to airport



Mode = Curb

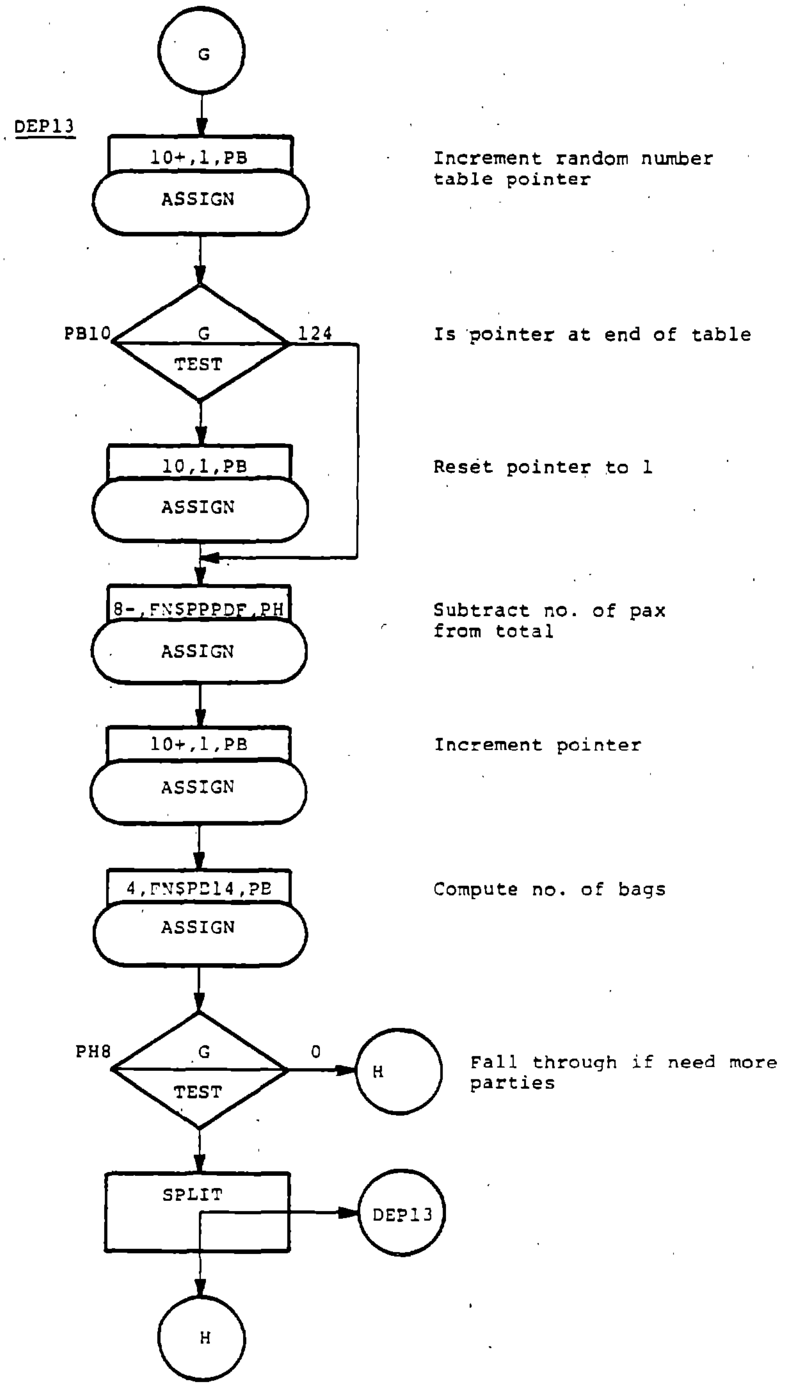
No. of pax to be met

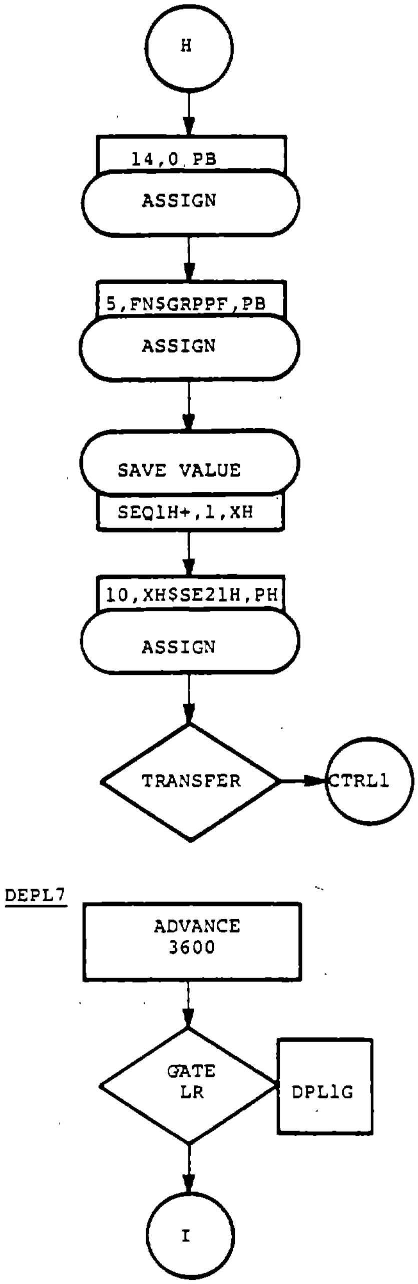
Fall through if none

Add to total met

Starting point in random number table

Will go to curb


G
DEP13
10+,1,PB
ASSIGN
Increment random number table pointer
PB10
G
124
TEST
Is pointer at end of table
10,1,PB
ASSIGN
Reset pointer to 1
8-,FNSPPPDF,PH
ASSIGN
Subtract no. of pax from total
10+,1,PB
ASSIGN
Increment pointer
4,FNSPB14,PB
ASSIGN
Compute no. of bags
PH8
G
0
TEST
H
Fall through if need more parties
SPLIT
DEP13
H



Reset PB14 for later use

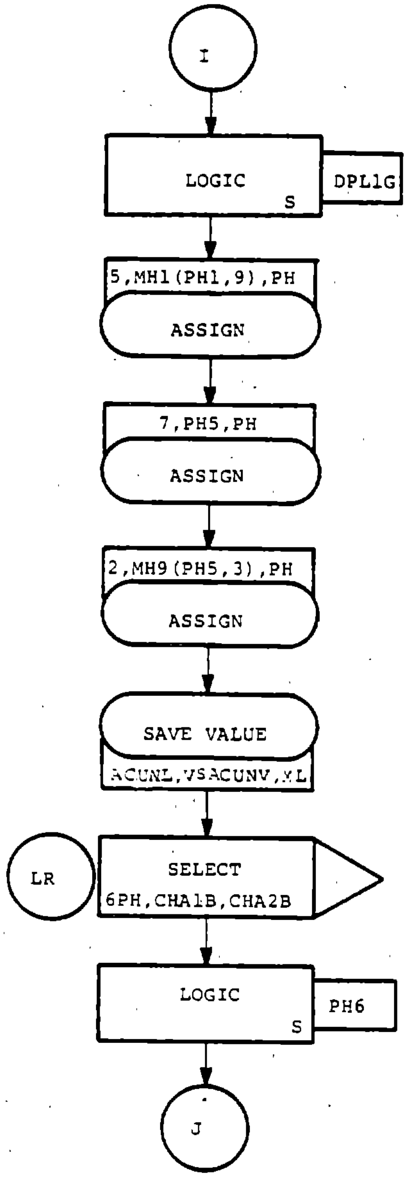
Greeters per party;
not used unless recirculating
and parking

Increment transaction
Counter

Transaction sequence no.

Advance deplaning pax to
time of arrival

Guard 'BAGS' from the
simultaneous arrival
of aircraft



set switch that one flight is being processed

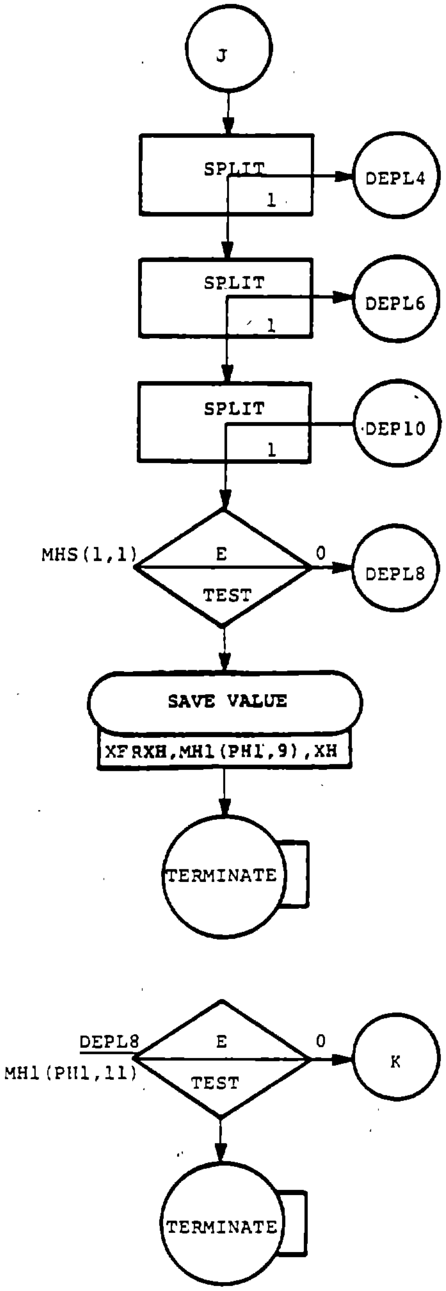
Gate number

Last MH9 row

Point no. of gate

Maximum pax unloading time

Pick free bag claim chain and switch



Copy to terminating pax

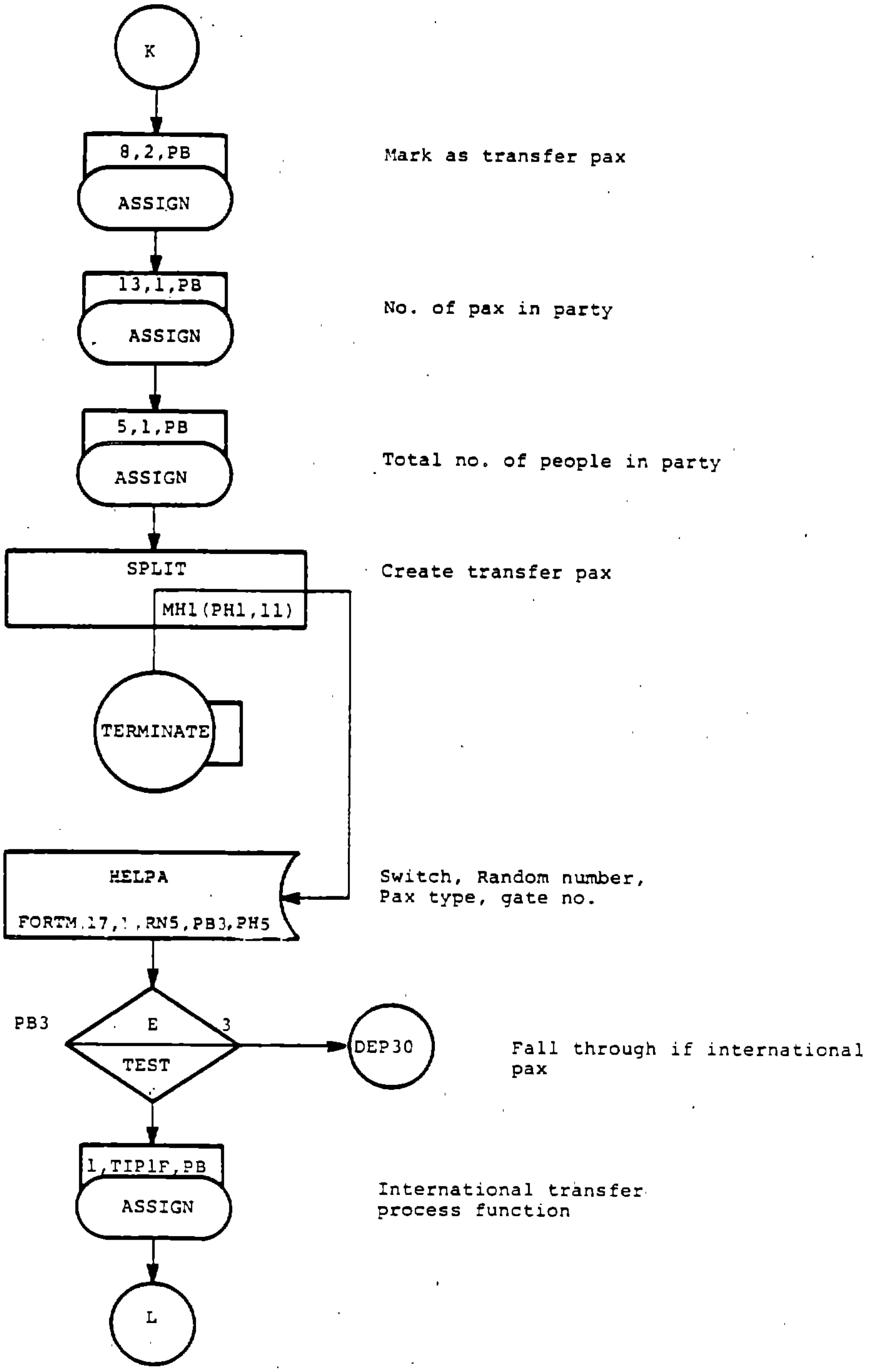
Copy to transit pax

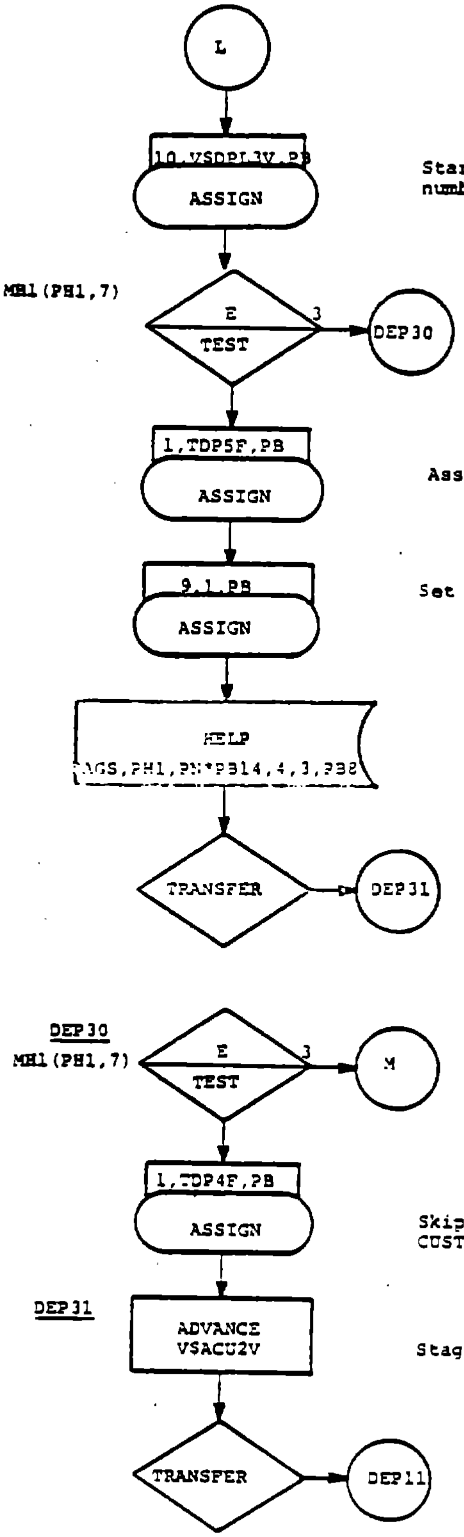
Copy to transfer out-of-system

Fall through if no departing flights in transfer table

Add transfer pax to holding save value

Fall through if no transfer pax


K
8,2,PB
ASSIGN
Mark as transfer pax
13,1,PB
ASSIGN
No. of pax in party
5,1,PB
ASSIGN
Total no. of people in party
SPLIT
MH1(PH1,11)
Create transfer pax
TERMINATE
HELPA
FORTM,17,1,RN5,PB3,PH5
Switch, Random number,
Pax type, gate no.
PB3
E
3
TEST
DEP30
Fall through if international
pax
1,TIP1F,PB
ASSIGN
International transfer
process function
L



Starting point in random
number table

Fall through if international
flight

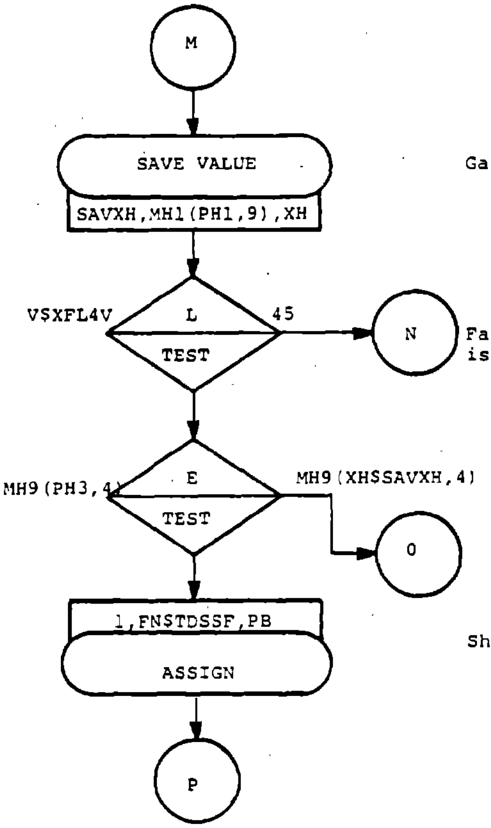
Assign Dom.Com transfer function

Set pax as not ticketed

Is outgoing flight
International

Skip IMMIØ,BAGCØ,
CUSTØ

Stagger pax off aircraft

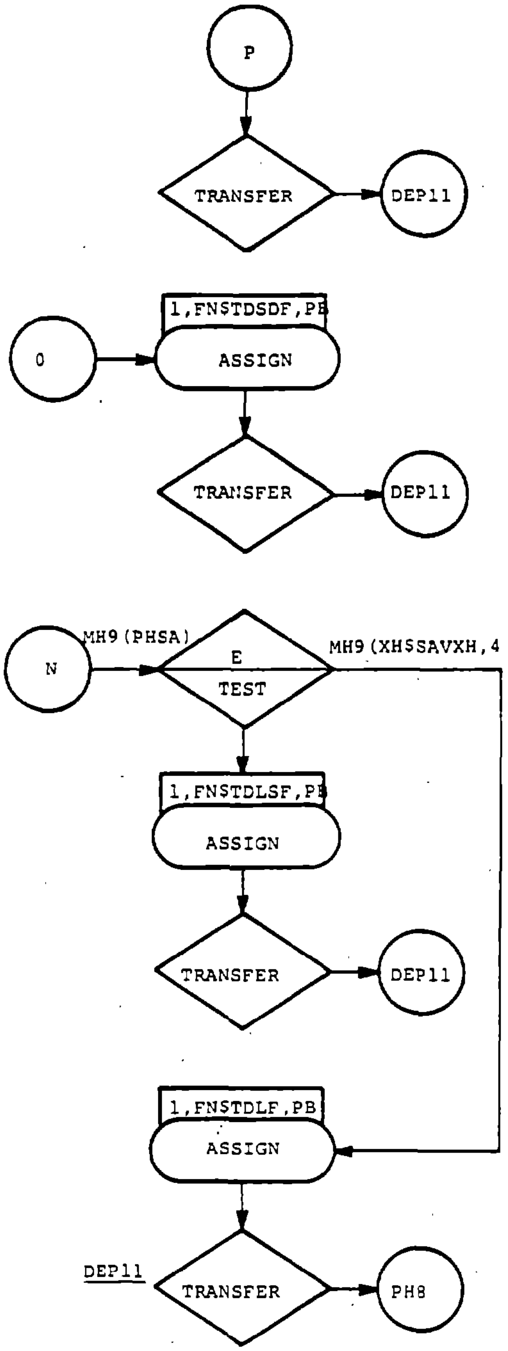


Gate no. of departing flight

Fall through if departure
is less than 45 minutes

Fall through if
same section

Short-stay/same section

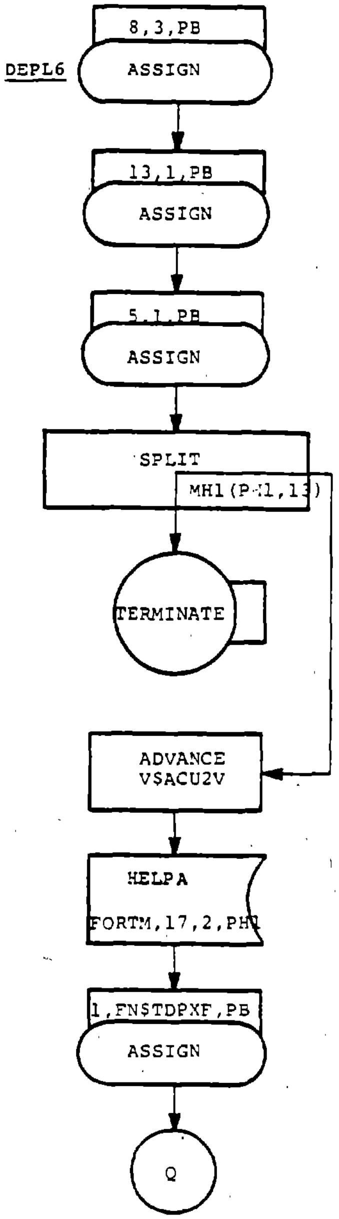


Short-stay/different section

Fall through if
same section

Long-stay/same section

Long-stay/different section



Mark as transit

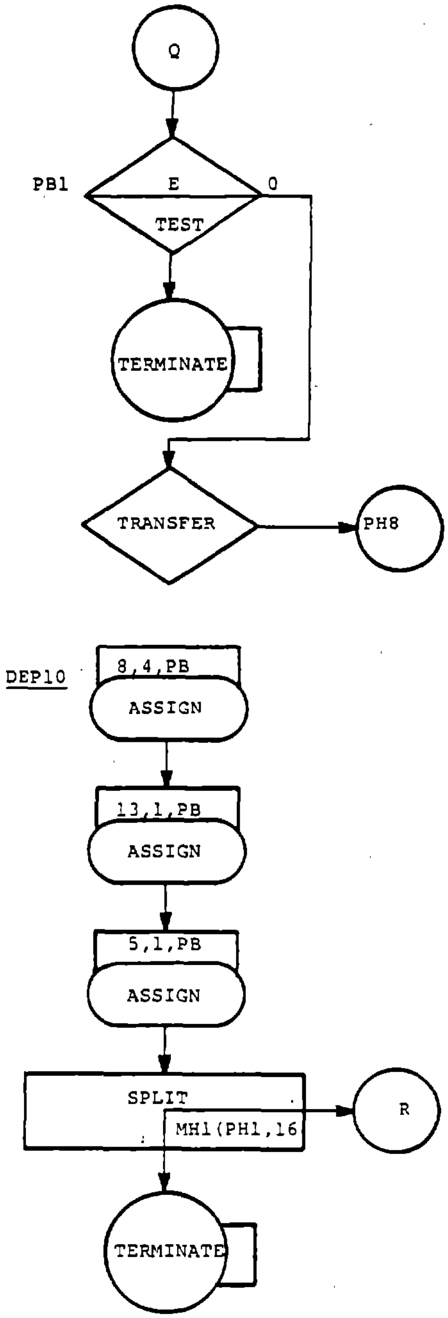
No. of pax in party

Total no. of people in party

Stagger pax off aircraft

Transfer pax section
switch, arriving flight

Select a process fucntion



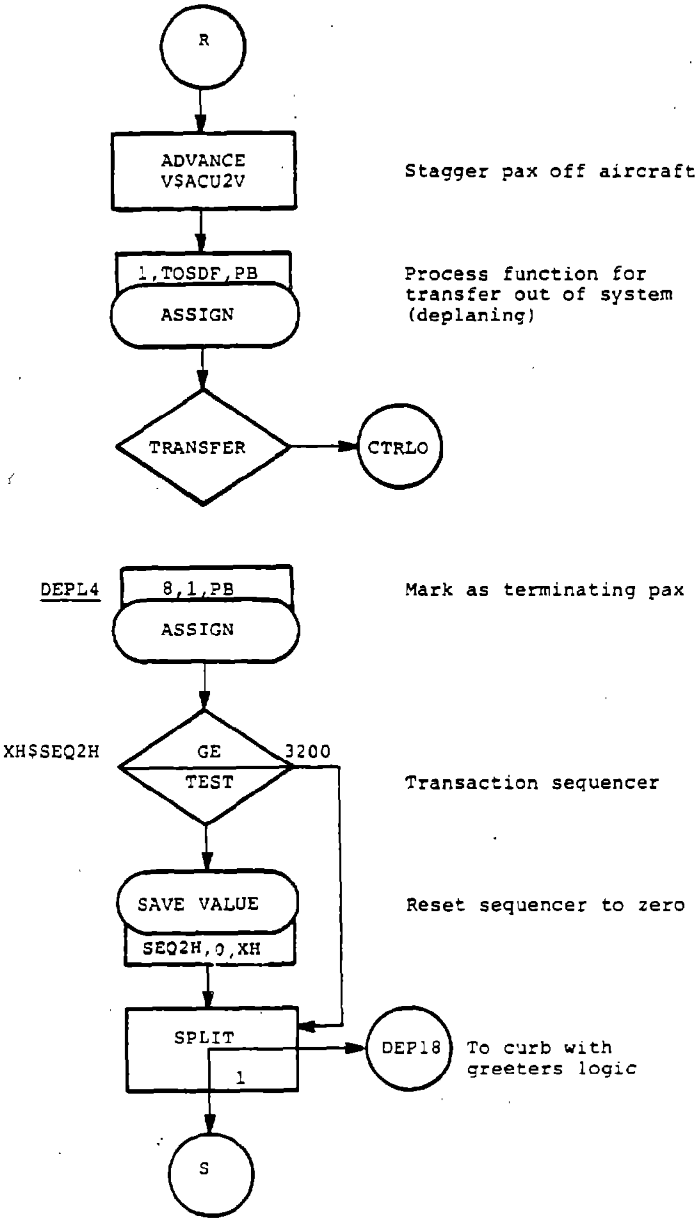
fall through if
never deplaned

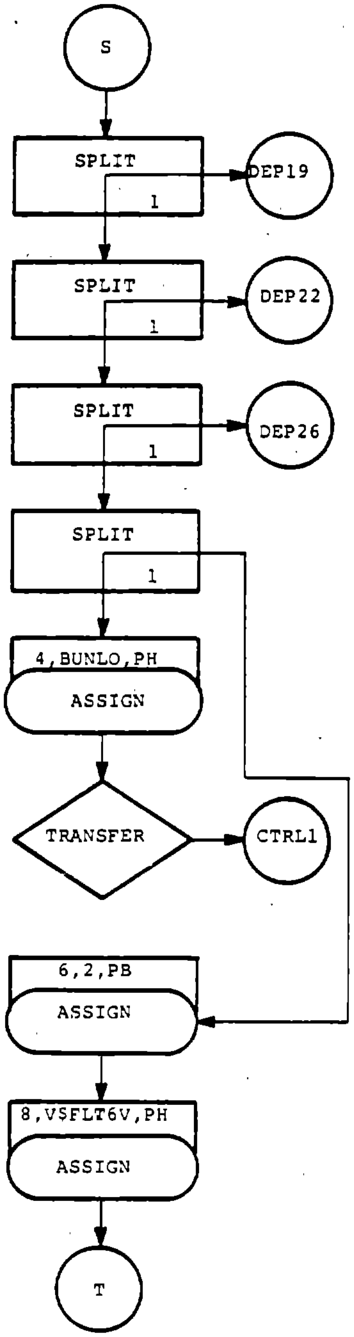
Mark as transfer-out-of-system

No. of pax in party

Total no. of people in party

Create transfers
out-of-system


R
ADVANCE
V$ACU2V
Stagger pax off aircraft
1,TOSDF,PB
ASSIGN
Process function for
transfer out of system
(deplaning)
TRANSFER
CTRLO
DEPL4
8,1,PB
ASSIGN
Mark as terminating pax
XH$SEQ2H
GE
3200
TEST
Transaction sequencer
SAVE VALUE
SEQ2H,0,XH
Reset sequencer to zero
SPLIT
1
DEP18
To curb with
greeters logic
S



To park with
greeters logic

To curb without
greeters logic

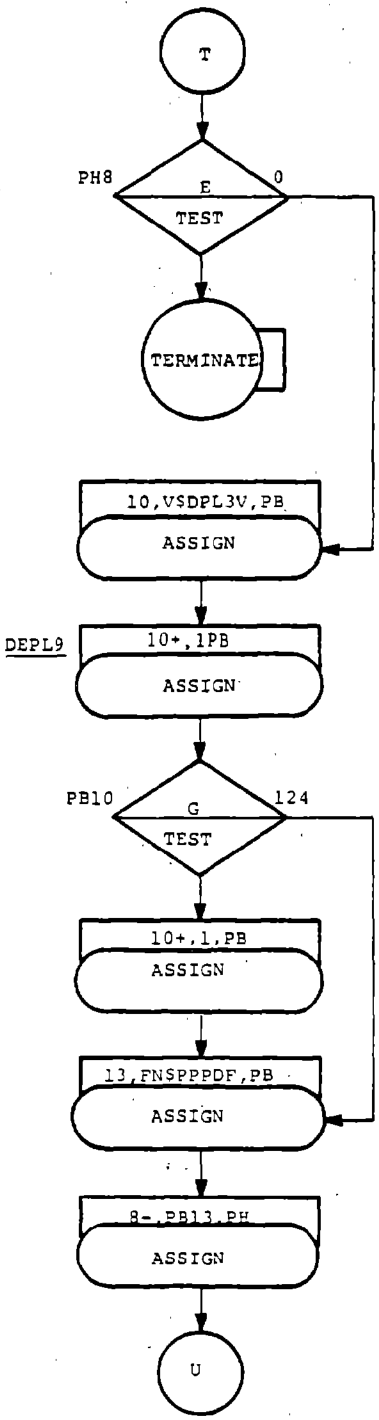
To park without
greeters logic

To all other modes

Assign address parameter

For following assign only

No. of pax in modes
other than curb/park



Fall through if none

Starting point in random number table

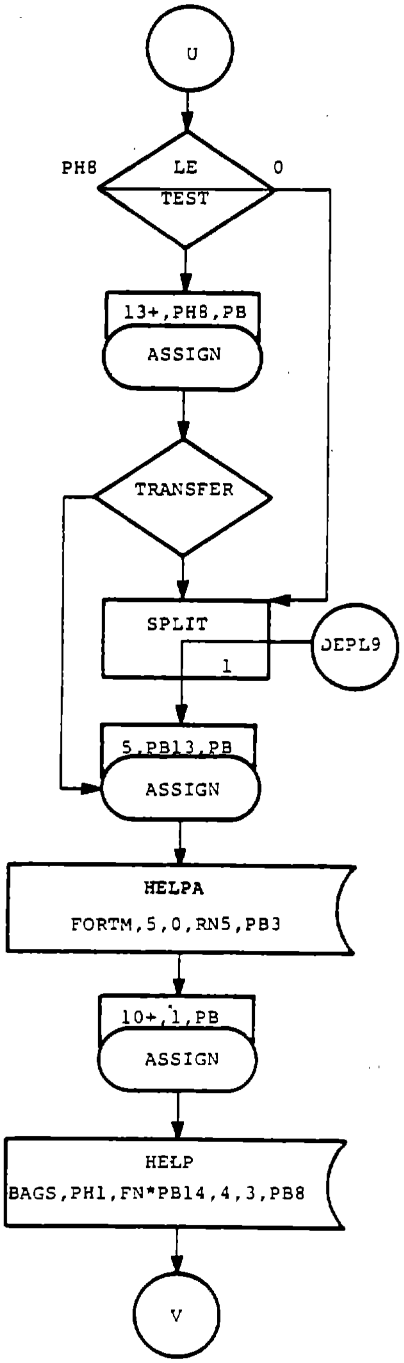
Increment random number table pointer

Is pointer at end of table

Reset pointer to one

No. of Pax in party

Subtract from total



Fall through if total less than or equal to zero

Adjust last party size

Transfer out of loop

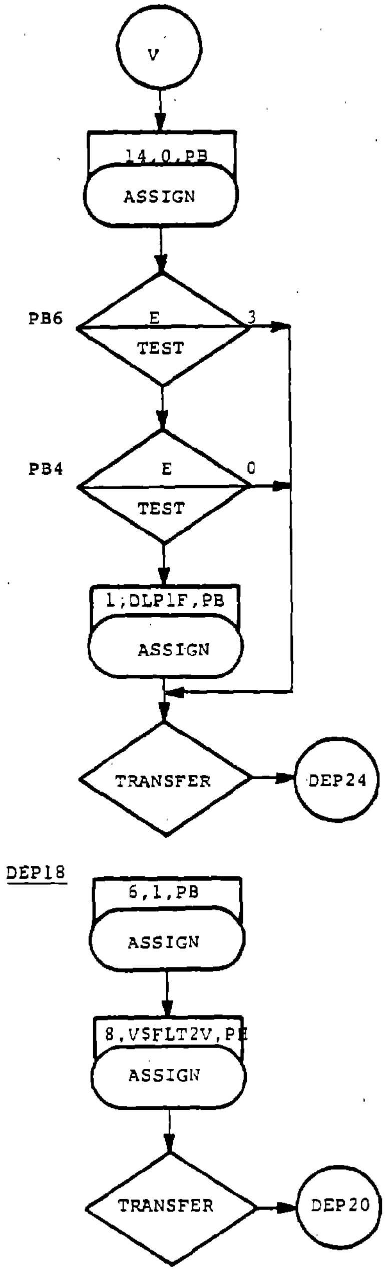
Go back to create another

Total no. of people in party

Park ground transportation mode

Increment pointer

Generate no. of bags, maximum random no.



Reset PB14 for later use

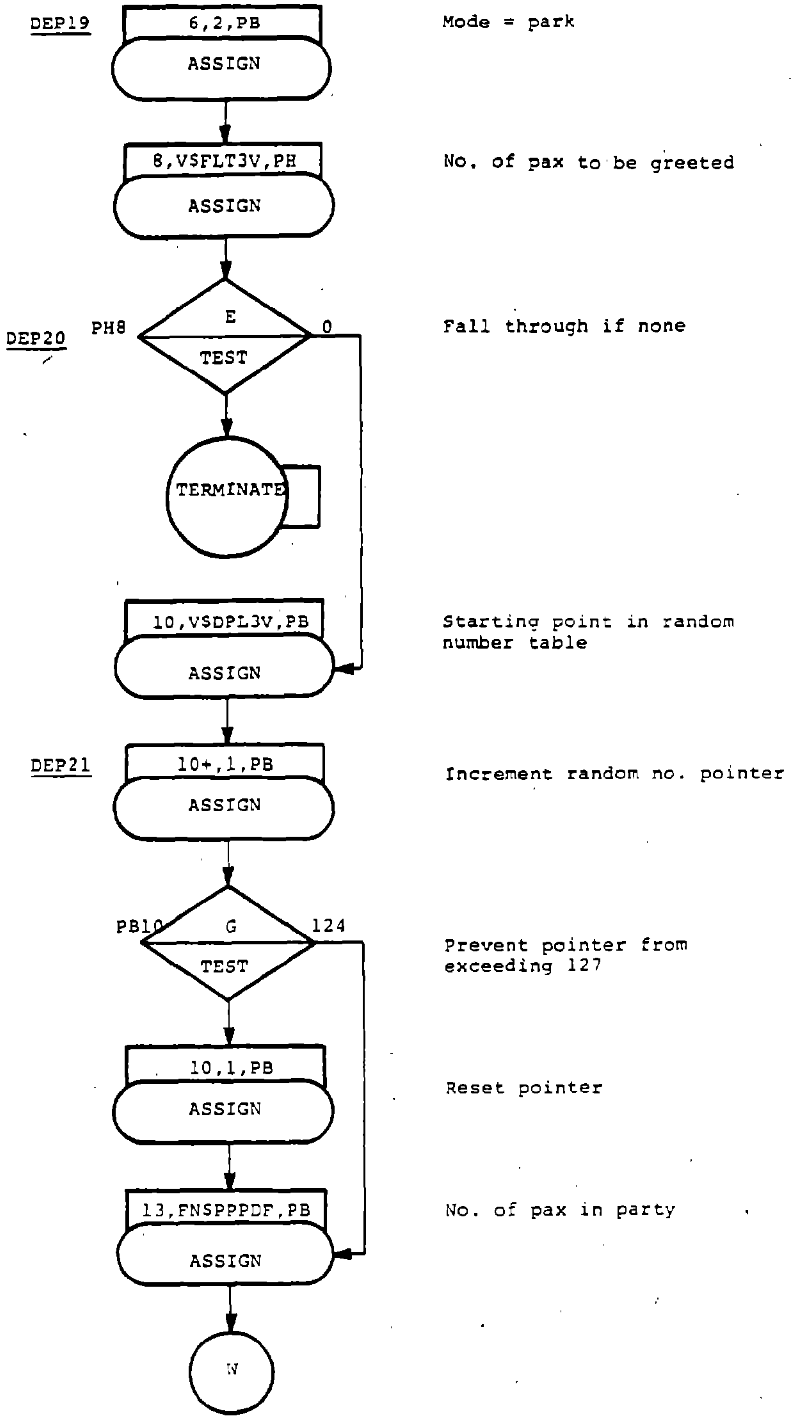
If rent-a-car, use DDP1F

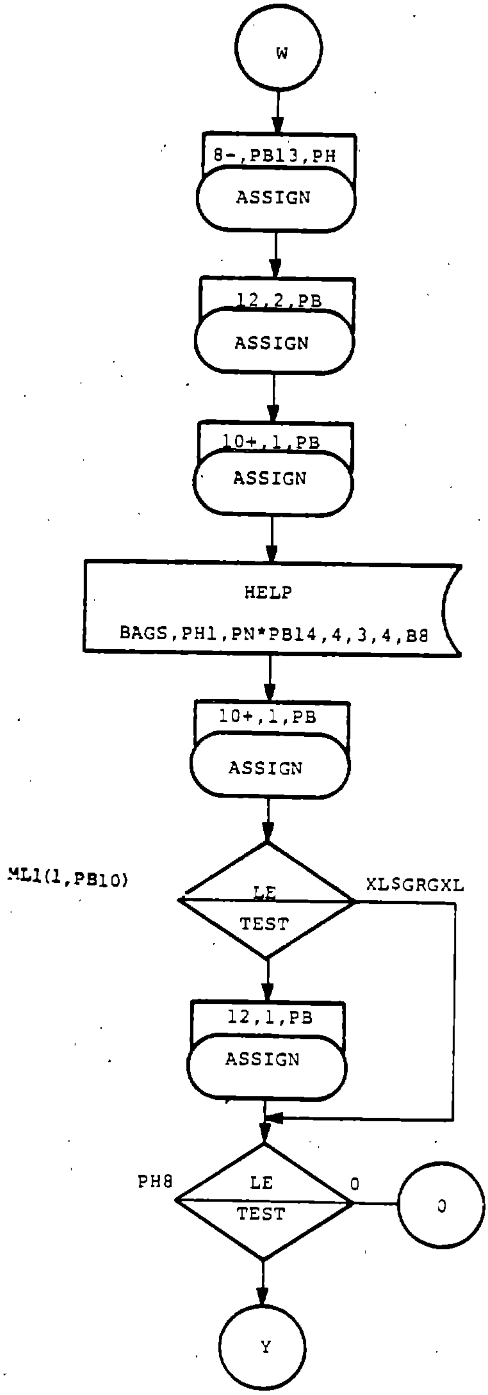
Fall through if no bags

Change to lobby process function

Mode = curb

No. of parking greeters proceeding to curb


DEP19
6,2,PB
ASSIGN
Mode = park
8,VSFLT3V,PH
ASSIGN
No. of pax to be greeted
DEP20
PH8
E
TEST
0
Fall through if none
TERMINATE
10,VSDPL3V,PB
ASSIGN
Starting point in random number table
DEP21
10+,1,PB
ASSIGN
Increment random no. pointer
PB10
G
TEST
124
Prevent pointer from exceeding 127
10,1,PB
ASSIGN
Reset pointer
13,FNSPPPDF,PB
ASSIGN
No. of pax in party
W



Subtract group from total

Meet at bag claim

Increment random number pointer

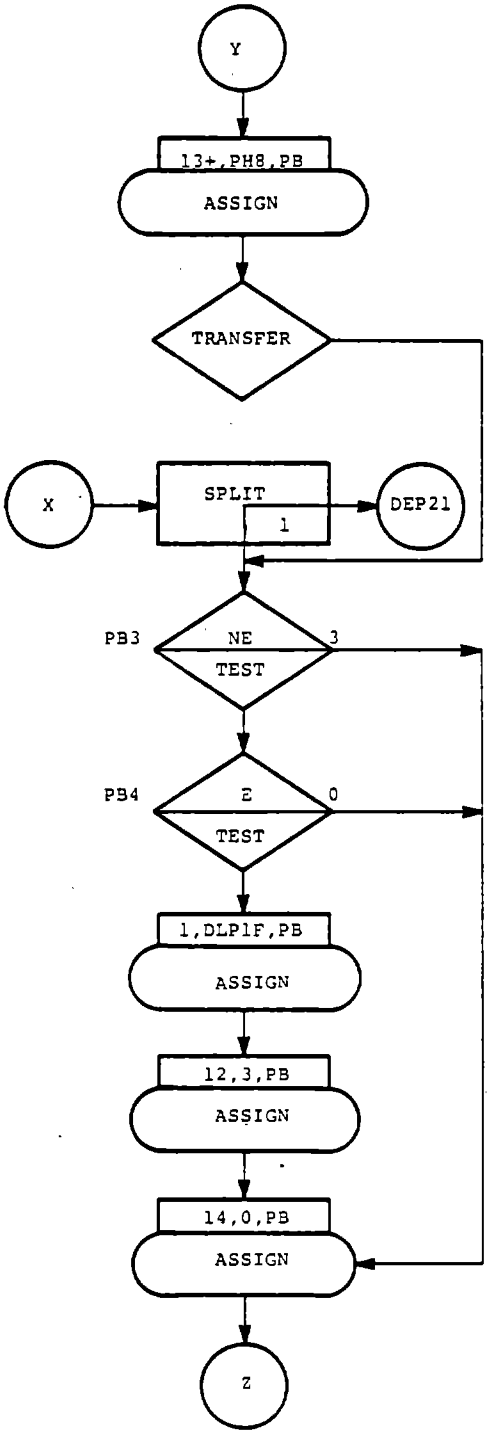
Generate no. of bags,
maximum random no.

Increment random number pointer

Fall through for gate
meeting

Meet at gate

Is last party size
too large



Adjust last party size

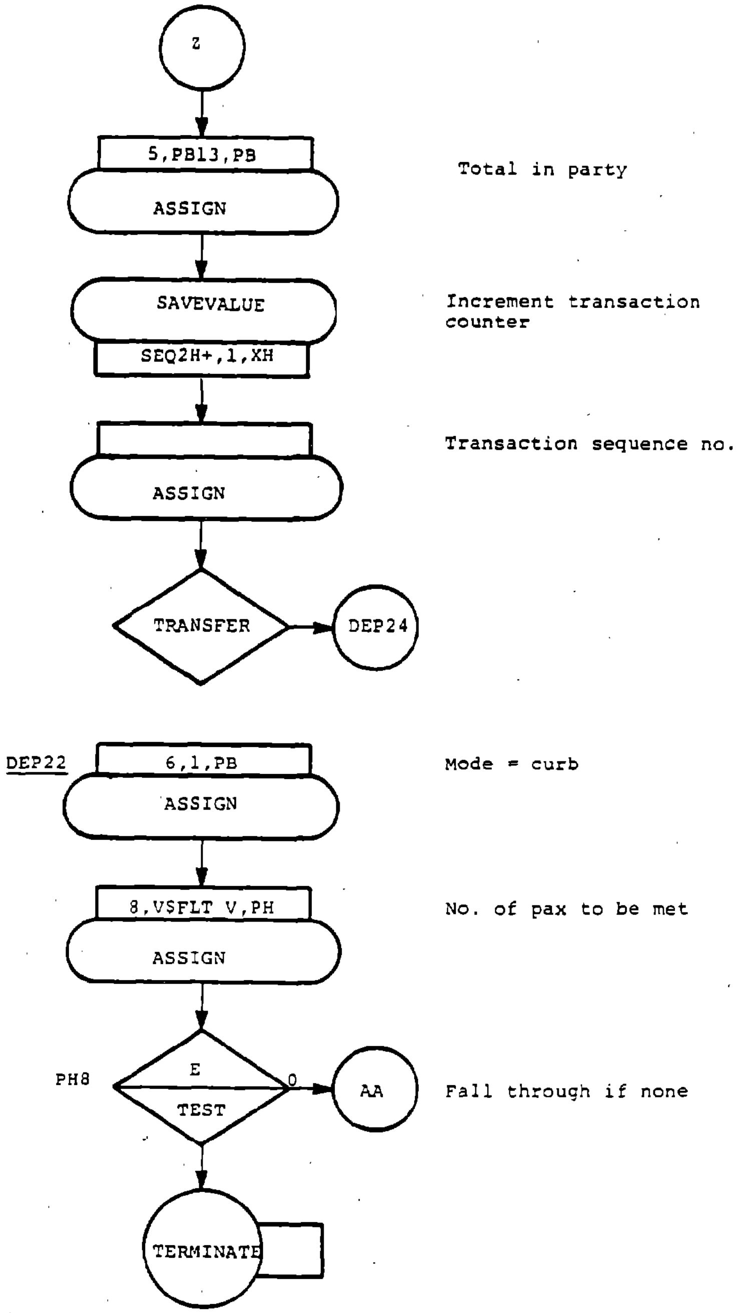
Fall through if not international

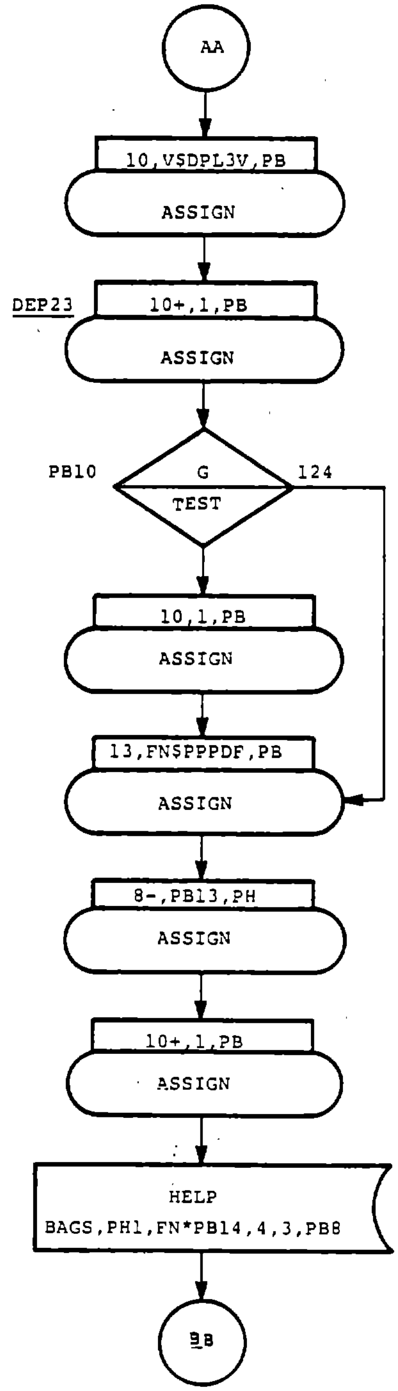
Fall through if no bags

Change to lobby process function

Meet at ticketing

Reset PB14 for later use


Z
5,PB13,PB
ASSIGN
Total in party
SAVEVALUE
SEQ2H+,1,XH
Increment transaction counter
ASSIGN
Transaction sequence no.
TRANSFER
DEP24
DEP22
6,1,PB
ASSIGN
Mode = curb
8,V$FLT V,PH
ASSIGN
No. of pax to be met
PH8
E
TEST
0
AA
Fall through if none
TERMINATE



Starting point in random number table

Increment random number pointer

Prevent pointer from exceeding 127

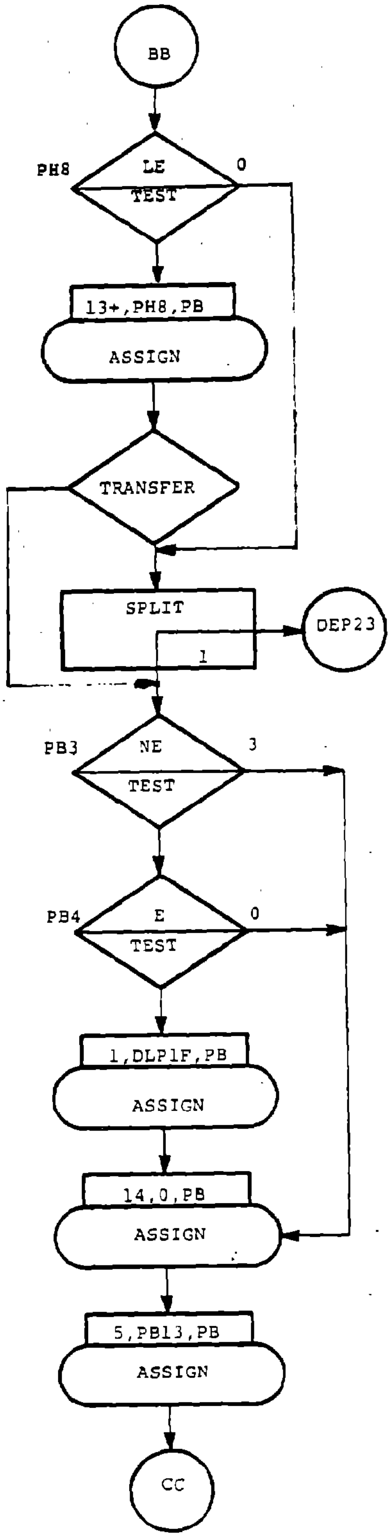
Reset pointer

Pax in party

Subtract group from total

Increment random number pointer

Generate no. of bags, maximum random no.



Adjust last party size.

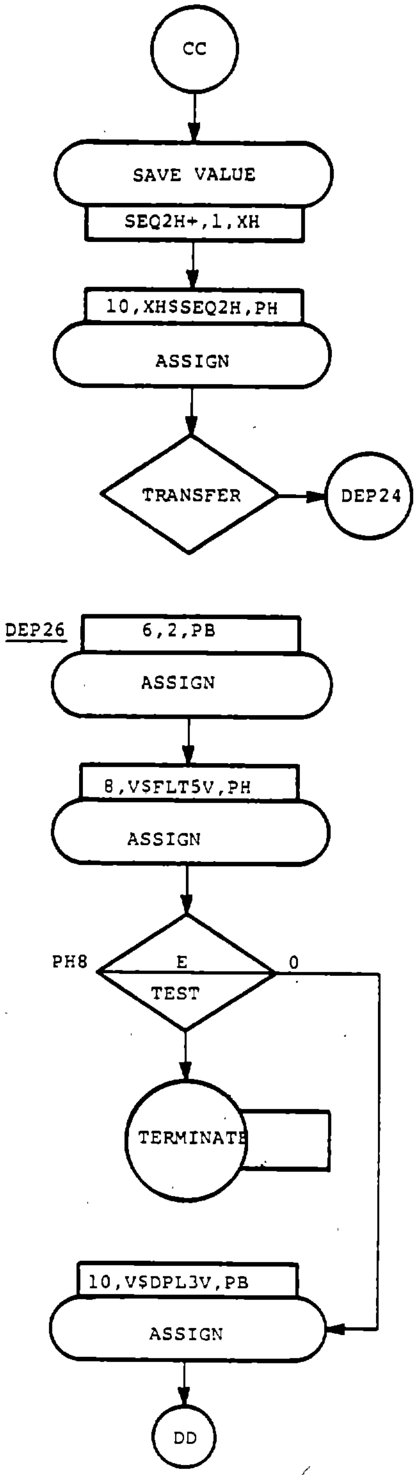
Fall through if not international

Fall through if no bags

Change to lobby process function

Reset PB14 for later use

Total in party



Increment transaction counter

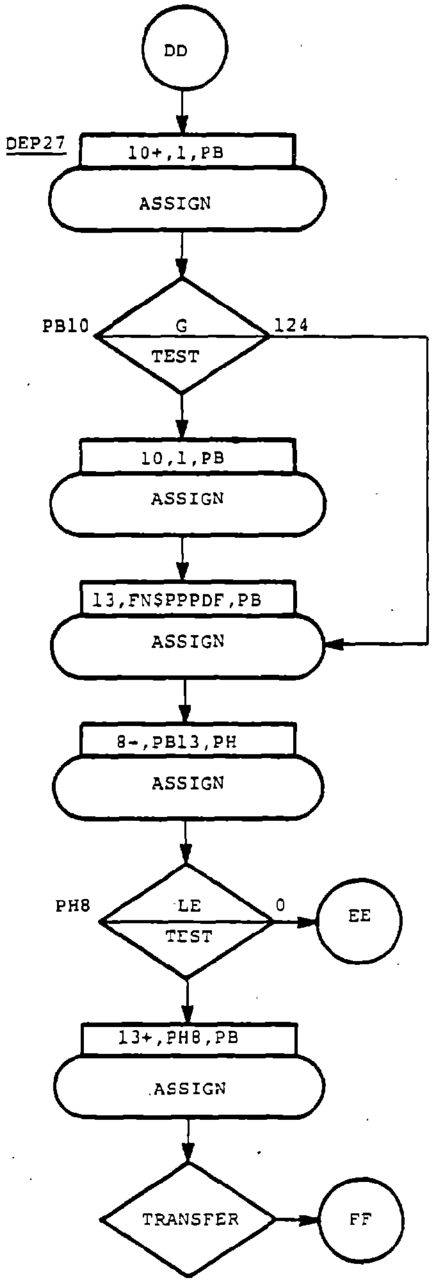
Transaction sequence no.

Mode = park

No. of pax to be created

Fall through if none

Starting point in random number table



Increment random number pointer

Prevent pointer from exceeding 127

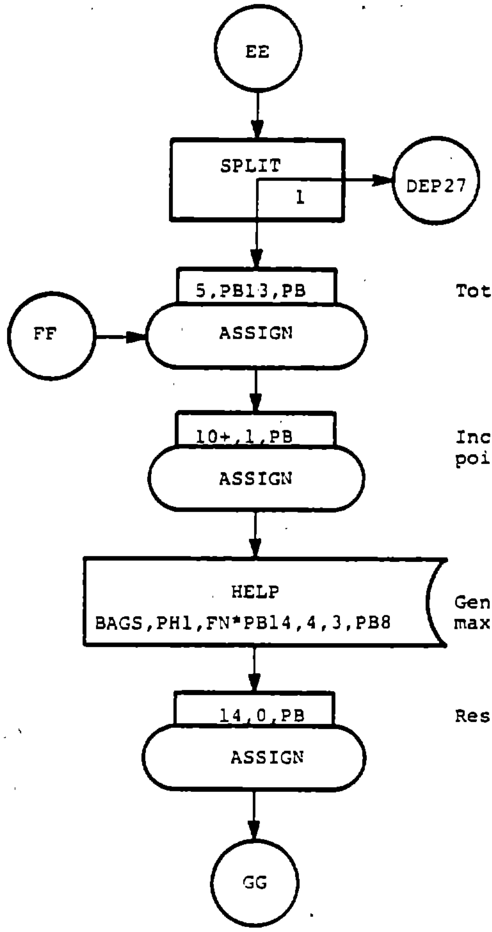
Reset pointer

Pax in Party

Subtract group from total

Is last party size too large

Adjust last party size

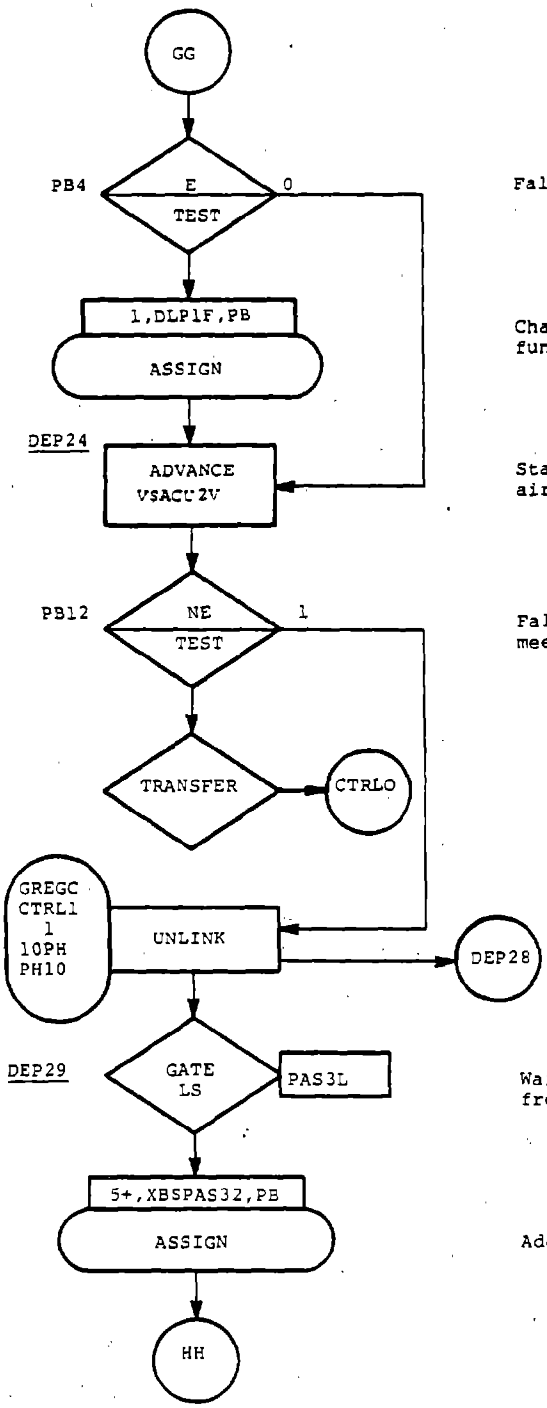


Total in party

Increment random number pointer

Generate no. of bags, maximum random no.

Reset PB14 for later use



Fall through if no bags

Change to lobby process function

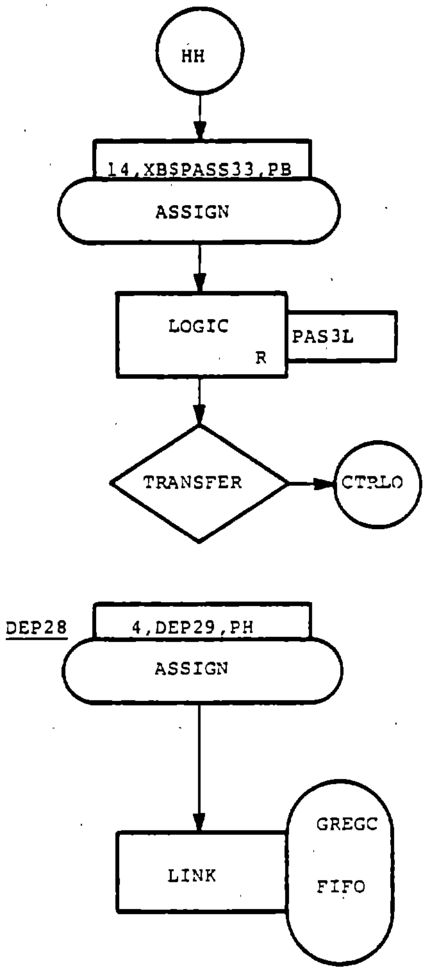
Stagger pax off aircraft

Fall through if not meeting at gate

Try to unlink greeter

Wait for information from greeter

Add greeters to party



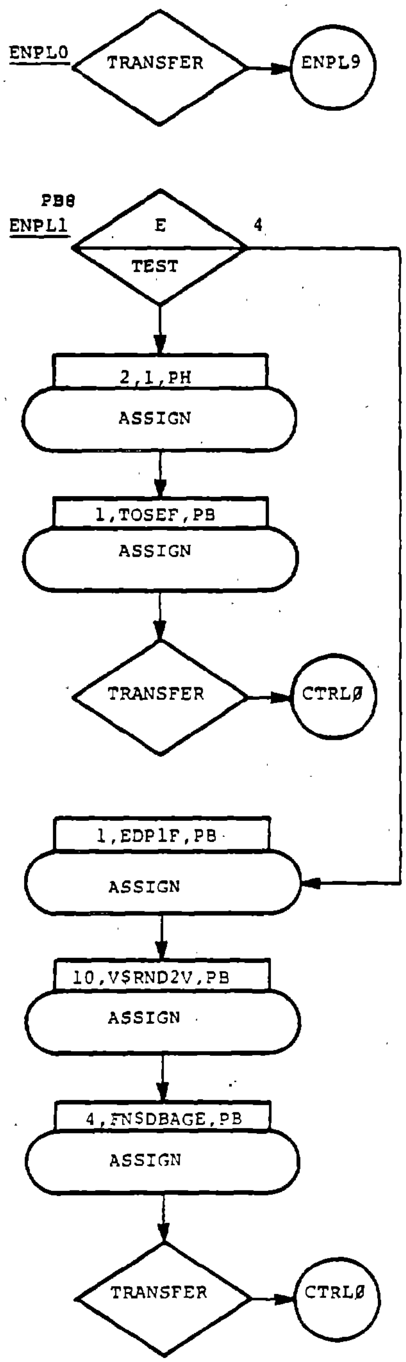
Pick up parking lot no.

Permit next
of PASS save values

Will go to DEP29
when unlinked

Wait for greeters

ENPLANING PASSENGER LOGIC



First transaction on JOBTAPE which is a dummy, is sent here

Fall through if transfer from out of system

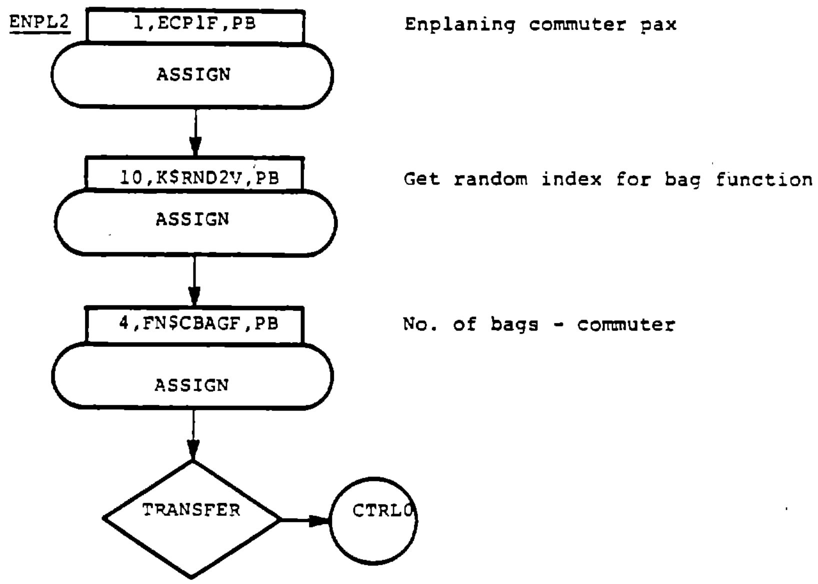
Entrance facility for out of system transfers

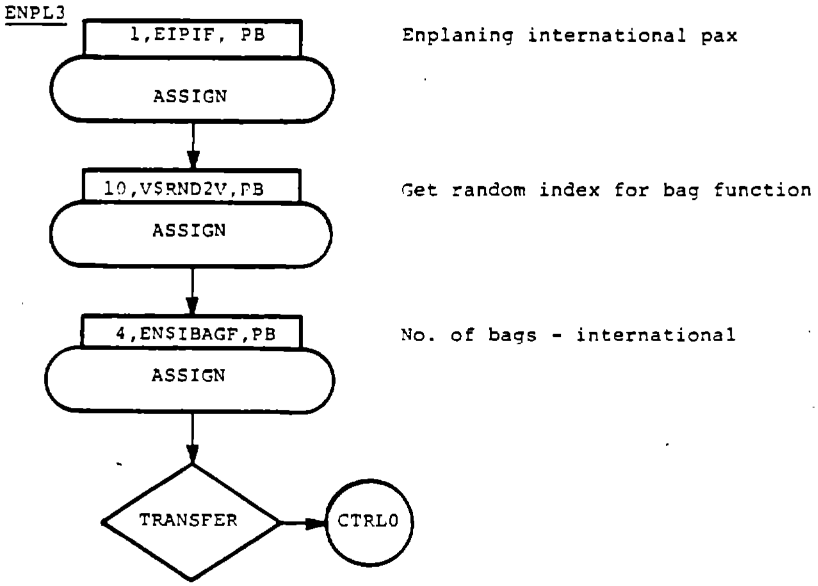
Process function number

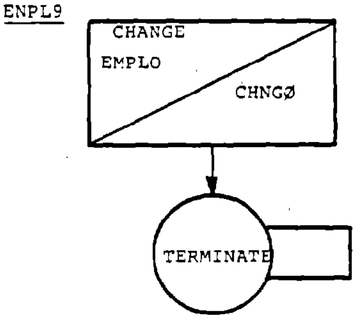
Enplaning domestic pax

Get random index for bag function

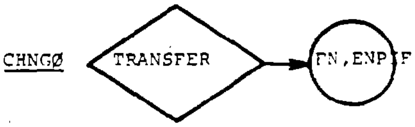
No. of bags - domestic


ENPL2
1,ECP1F,PB
ASSIGN
Enplaning commuter pax
10,K$RND2V,PB
ASSIGN
Get random index for bag function
4,FN$CBAGF,PB
ASSIGN
No. of bags - commuter
TRANSFER
CTRLO

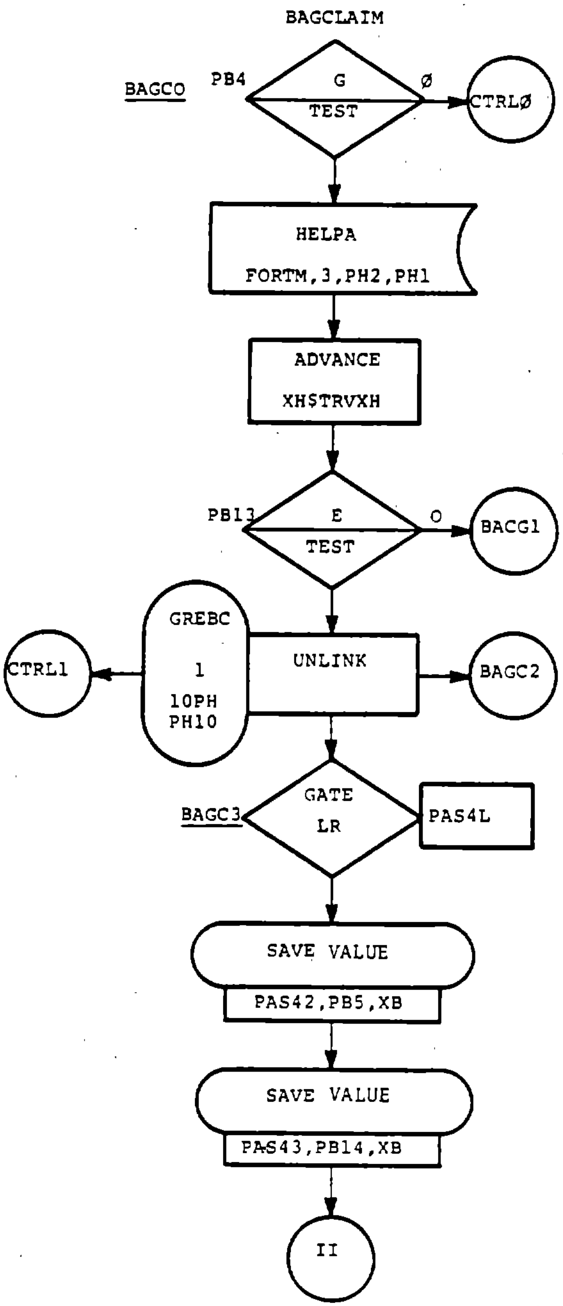

ENPL3
1,EIPIF, PB
ASSIGN
Enplaning international pax
10,V$RND2V,PB
ASSIGN
Get random index for bag function
4,EN$IBAGF,PB
ASSIGN
No. of bags - international
TRANSFER
CTRLO



Redirect actual enplaning pax transactions to CHNGØ



Route pax according to whether domestic, commuter, or international flight



Return to control if no bags

Location MH1 Row

Travel time to bag claim area

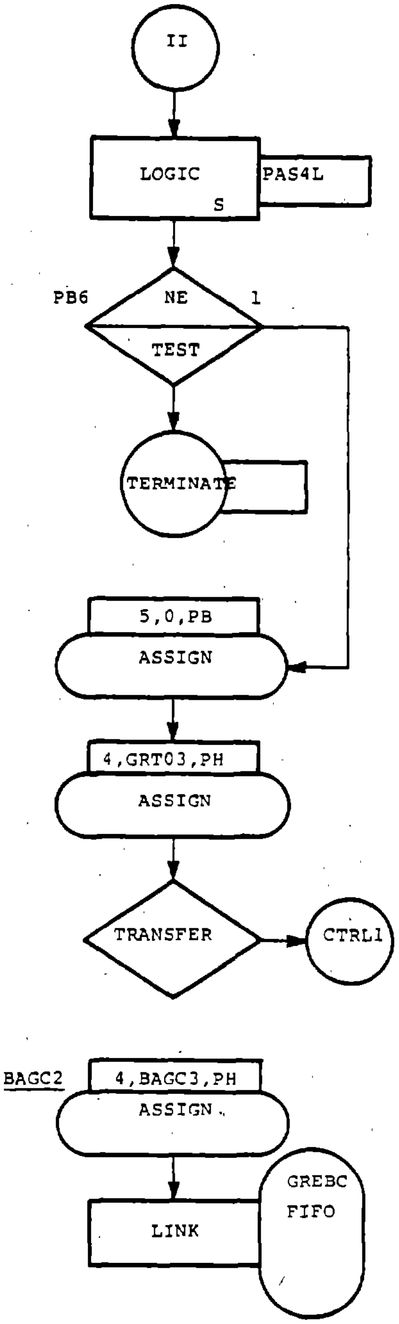
Fall through if greeter

Try to unlink pax

Only one at a time

Number of greeters

Parking lot number



Let pax pick up information

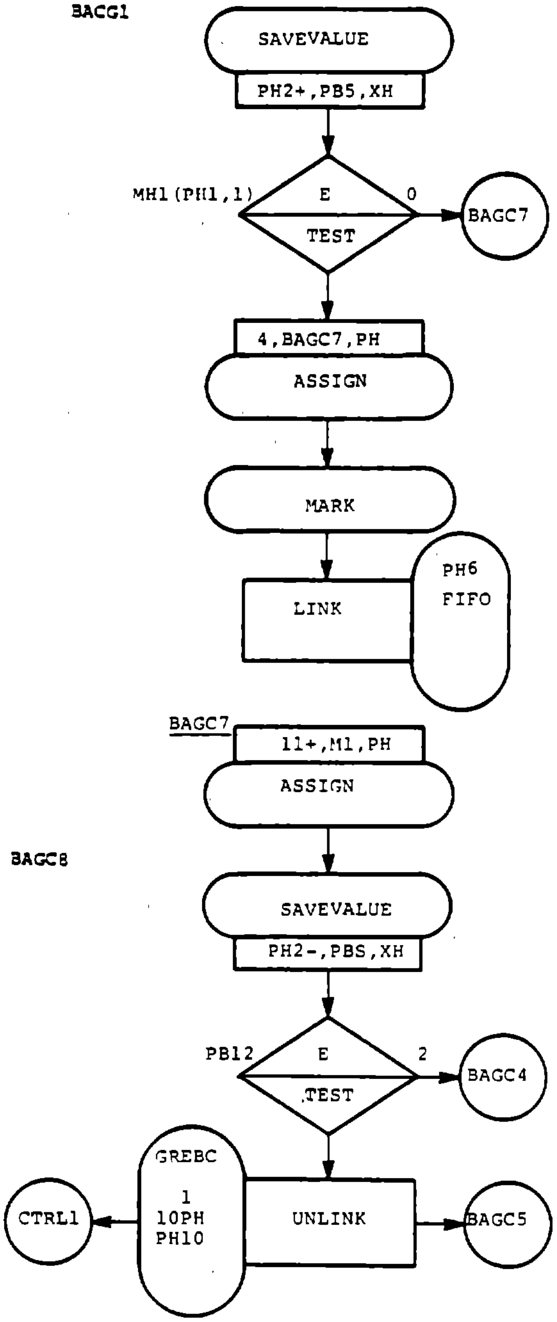
Branch if mode is curb

No one in party

Route to greeter leaving parking

Will go to BAGC3 when unlinked

Make greeter wait for passenger



Congestion

Branch if all bags
at bag claim

Save address parameter

Mark pax for waiting time

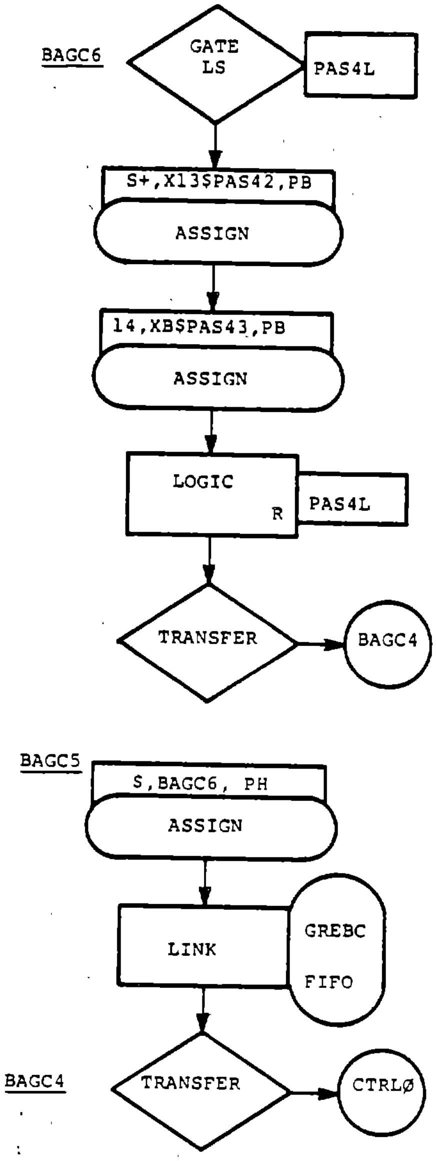
Wait for bags
(unlink in Baggage
unloading section)

Add waiting time to PH11

Congestion

Fall through if to be
greeted at bag claim

Try to unlink greeter



Wait for information from greeter

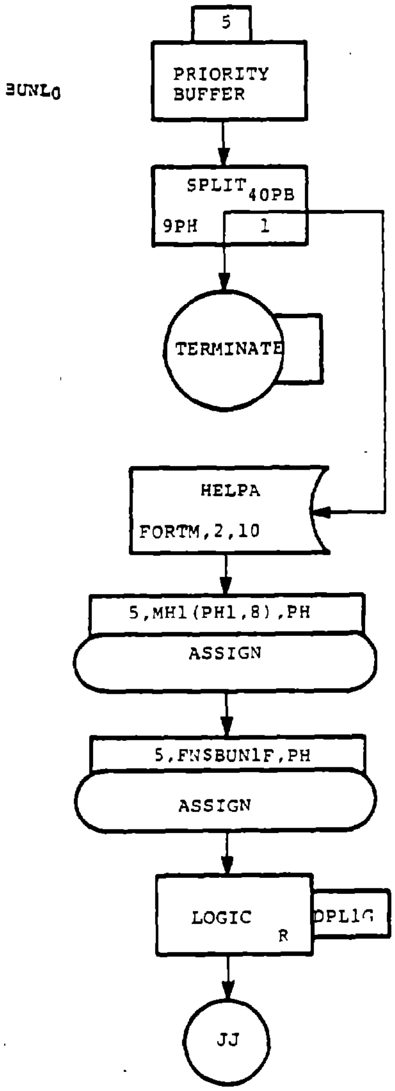
Add greeters to party

Pick up parking lot number

Permit next use of PAS4 savevalues

Will go to BAGC6 when unlinked

Make pax wait for greeter



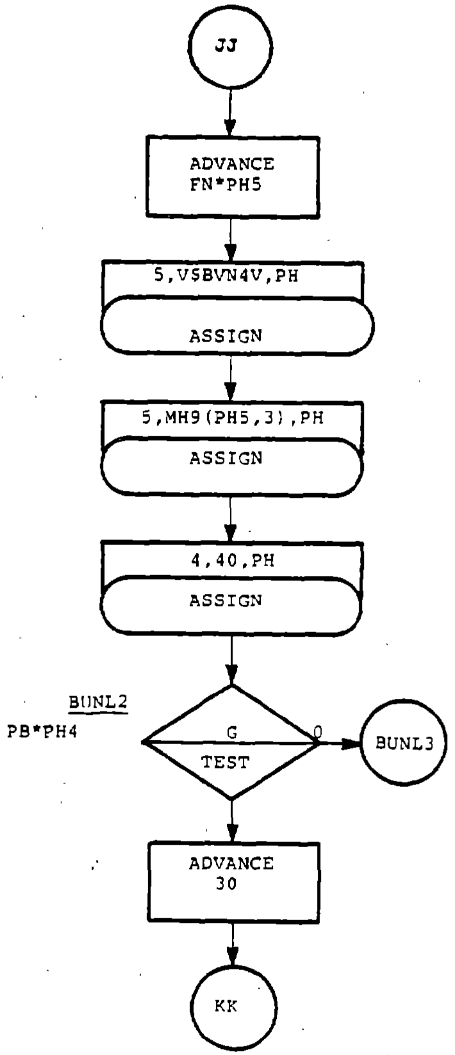
Allow pax to execute "BAGS"

Baggage unloading logic

Aircraft type

Unload time friction (by aircraft type)

Allow next aircraft transaction to move in deplaning section



Baggage unloading time

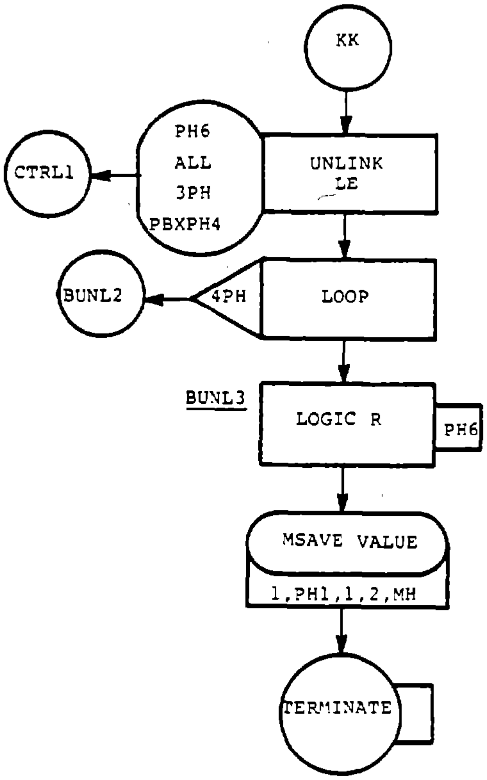
Baggage claim index
and area number

Point no. of baggage claim

Loop counter

Branch if all bags
unloaded

Bag unloading time at
baggage claim



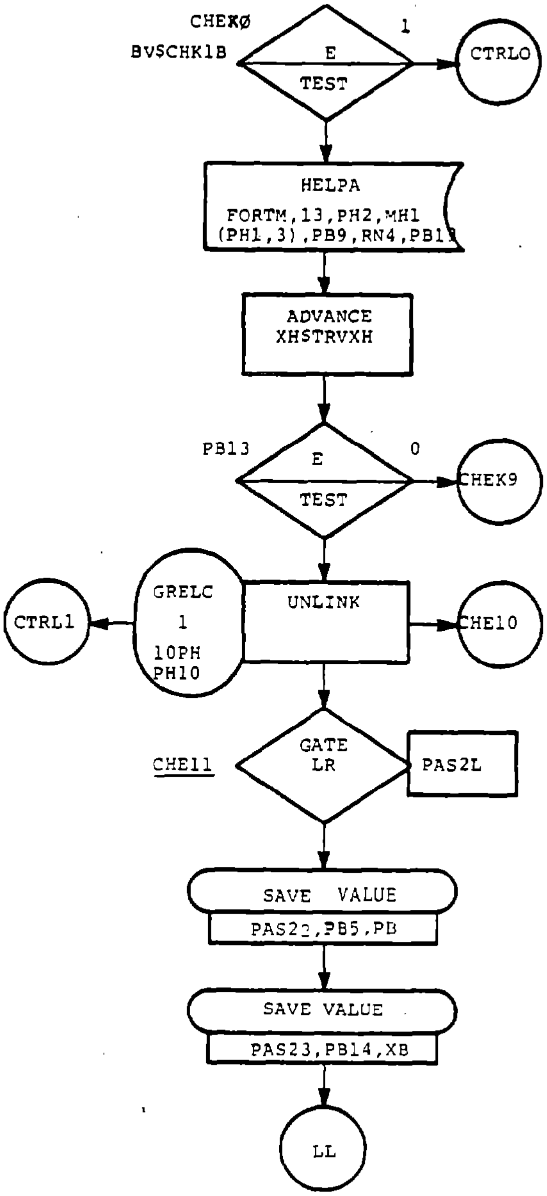
Bags to waiting pax

Loop for next group of bags

Free chain/switch pair

Mark all bags at claim area

TICKETING AND CHECKIN



Branch if enplaning ticketed, and direct

Location, airline, ticketed, express check random no. of pax

Walking time to ticketing and checkin

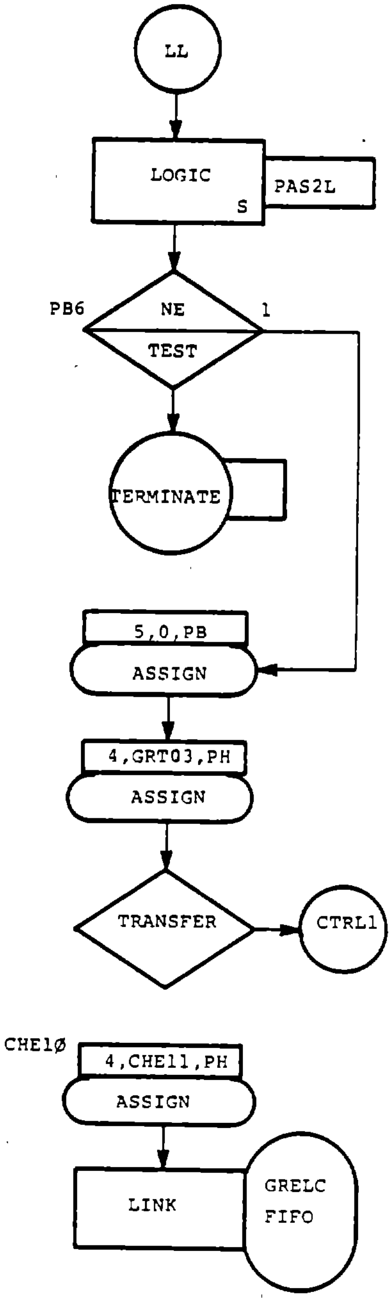
Fall through if greeter

Try to unlink pax

Only one at a time

Number of greeters

Parking lot number



Let pax pick up information

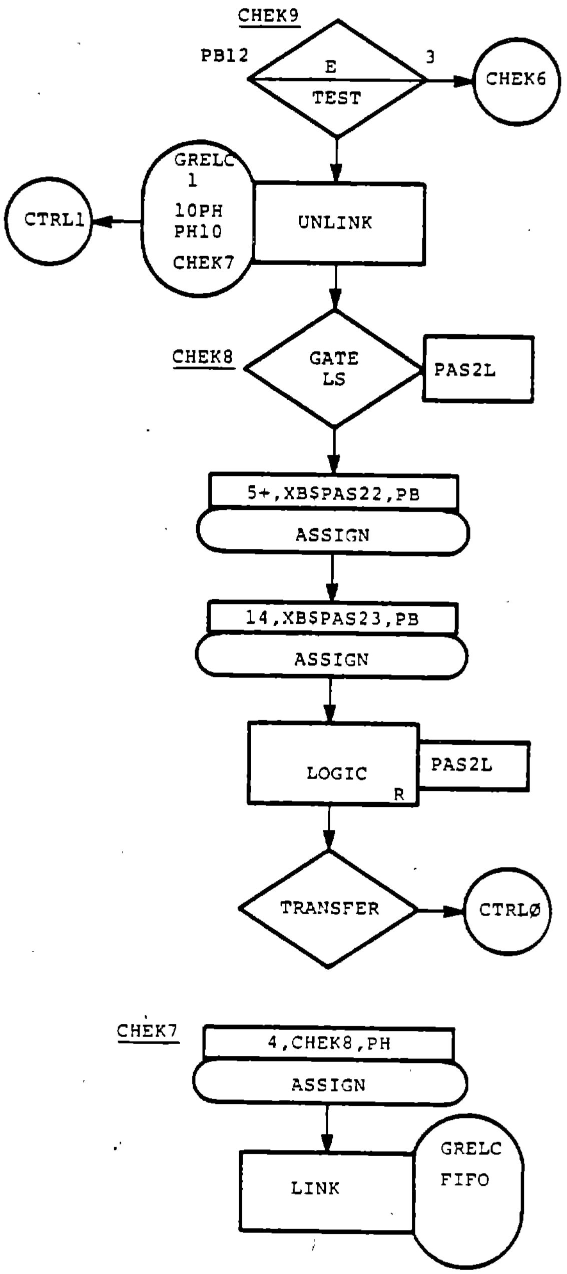
Branch if mode is curb

No one in party

Route to greeter leaving parking

Will go to CHE11 when unlinked

Wait for greeter



Fall through if to be greeted at ticketing

Try to unlink greeter

Wait for information from greeter

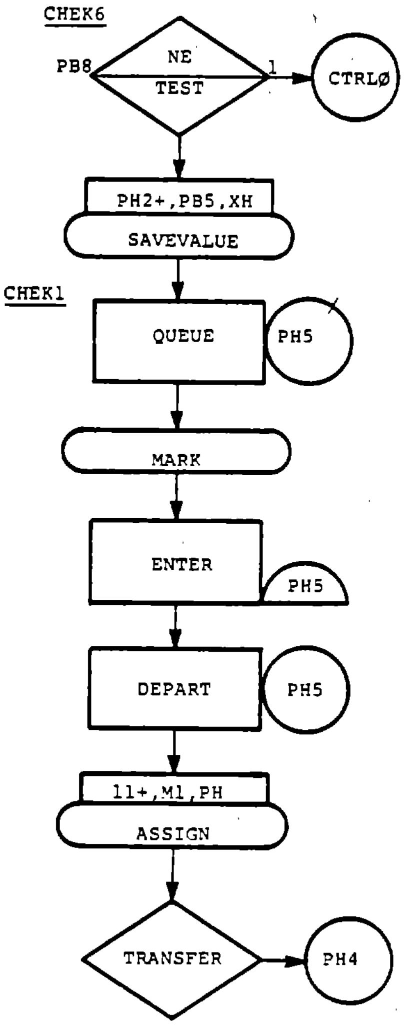
Add greeters to party

Pick up parking lot no.

Permit next use of PAS2 savevalues

Will go to CHEK8 when unlinked

Wait for greeter



Branch if terminating

Congestion

Wait for free server

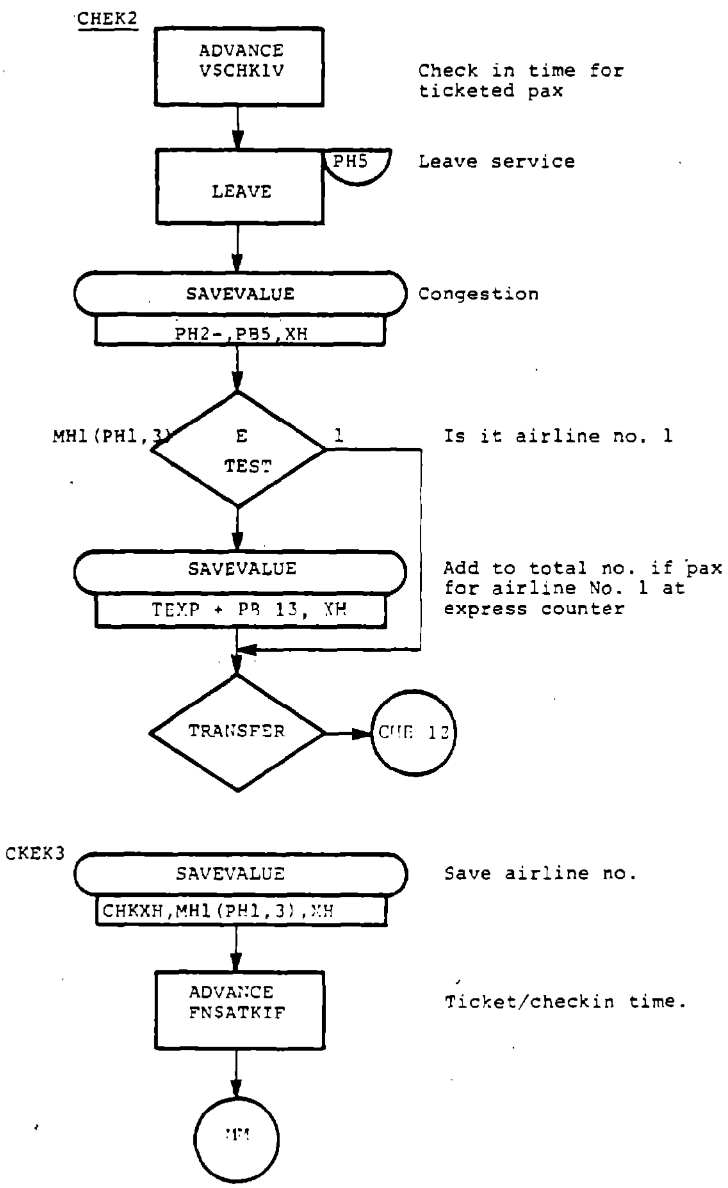
Mark pax for waiting time

Service

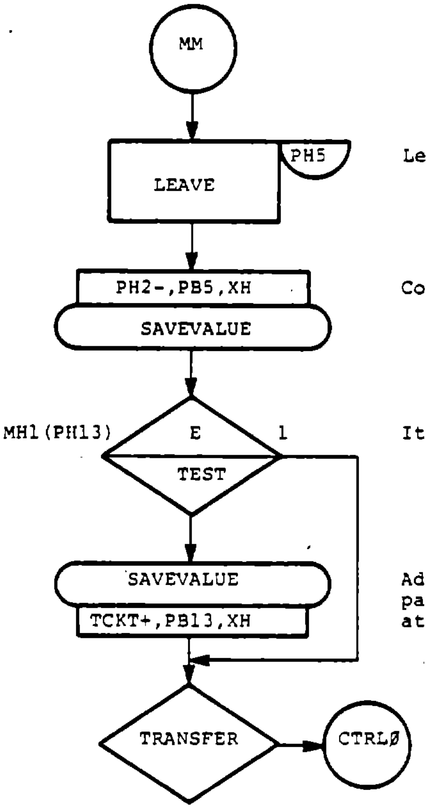
Add waiting time to PH11

Branch to CHEK2 or
CHEA3. Set in FORTM

Note: Halfword SAVEVALUEs may be inserted here to record flow through Express Check-In



Note: Airline full service counter flow values may be recorded in MH13



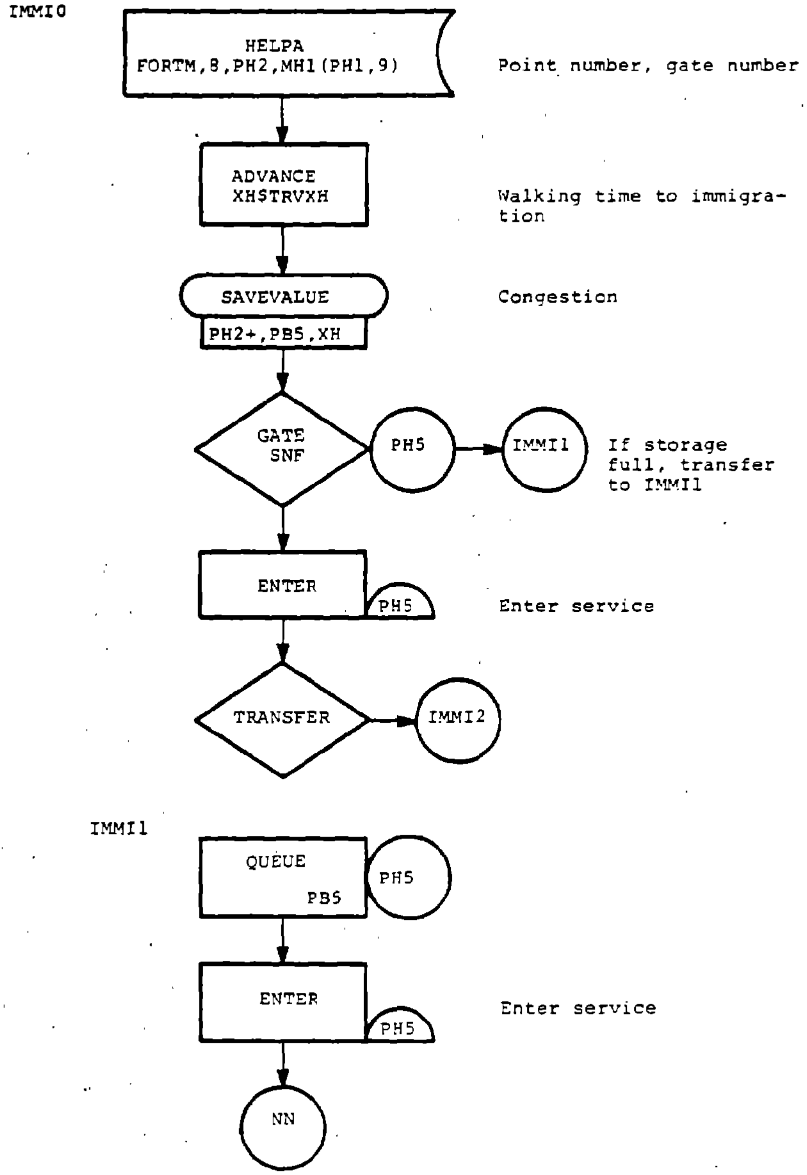
Leave service

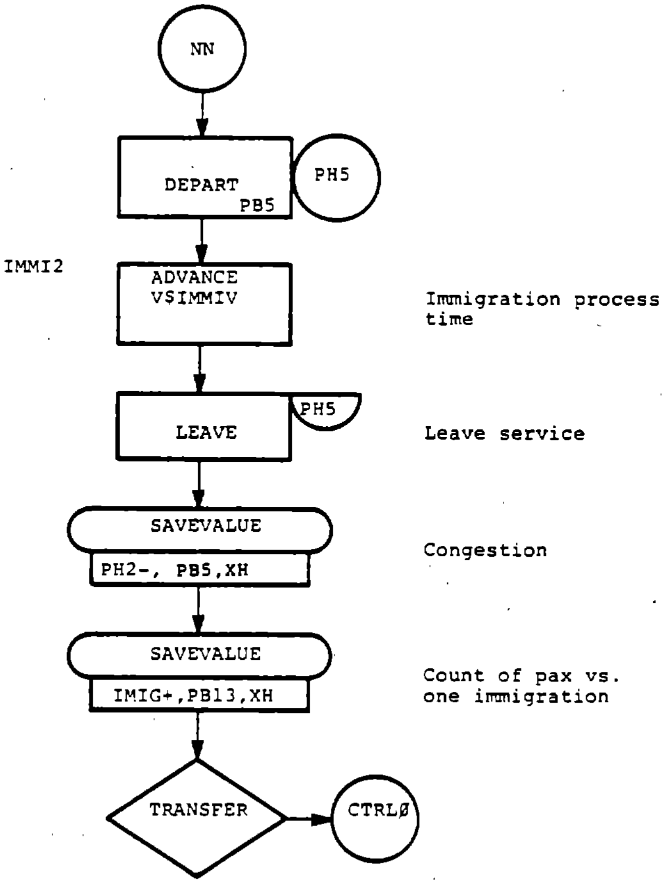
Congestion

It is airline no. 1

Add to total no. of pax for airline no. 1 at ticket counter

IMMIGRATION





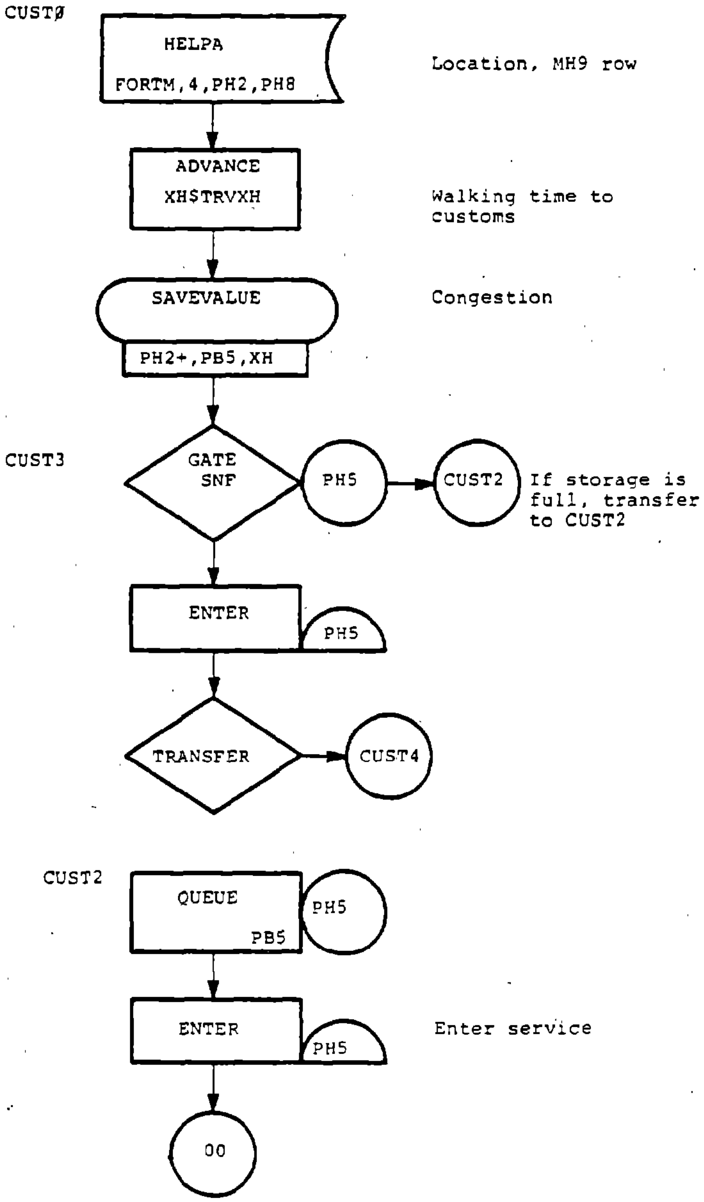
Immigration process time

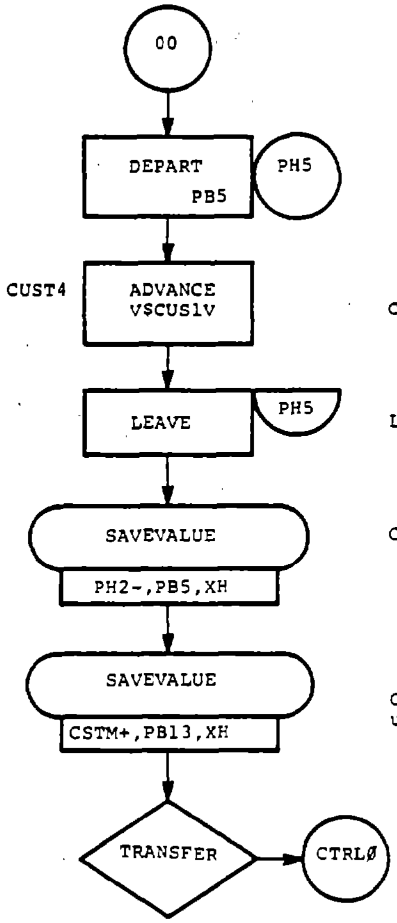
Leave service

Congestion

Count of pax vs. one immigration

CUSTOMS


CUSTØ
HELPA
FORTM,4,PH2,PH8
Location, MH9 row
ADVANCE
XH$TRVXH
Walking time to customs
SAVEVALUE
PH2+,PB5,XH
Congestion
CUST3
GATE
SNF
PH5
CUST2
If storage is full, transfer to CUST2
ENTER
PH5
TRANSFER
CUST4
CUST2
QUEUE
PB5
PH5
ENTER
PH5
Enter service
00



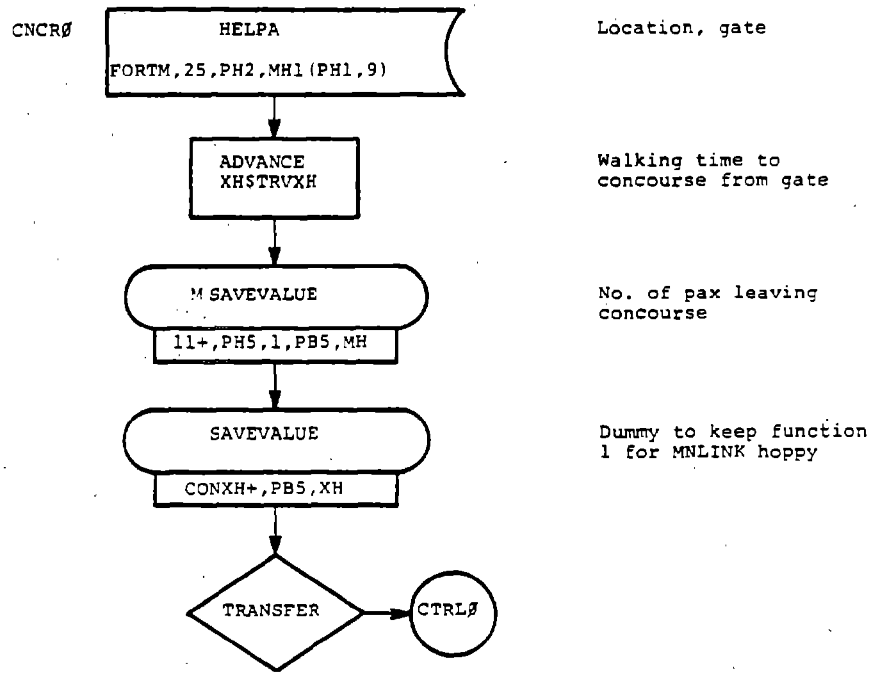
Customs process time

Leave service

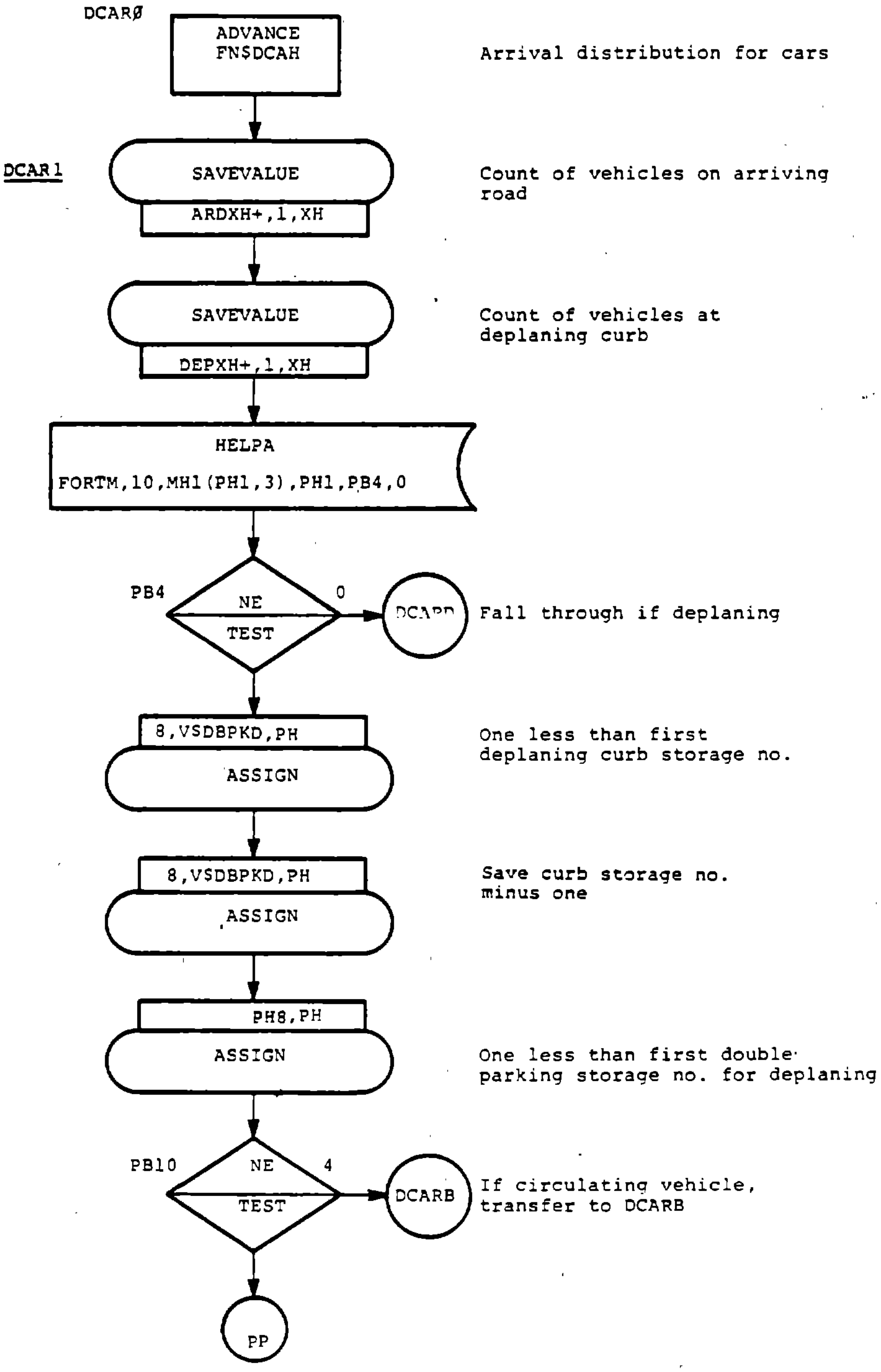
Congestion

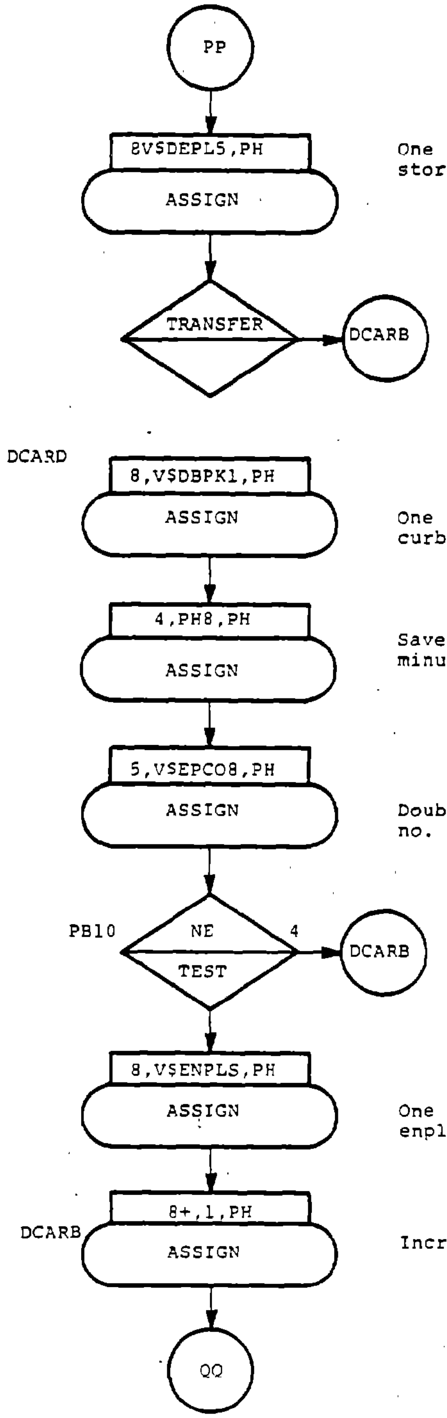
Count of pax
using customs

CONCOURSE EXIT - TERMINATING PAX



DEPLANING CURB (CARS)


DCARØ
ADVANCE
FN$DCAH
Arrival distribution for cars
DCAR1
SAVEVALUE
ARDXH+,1,XH
Count of vehicles on arriving road
SAVEVALUE
DEPXH+,1,XH
Count of vehicles at deplaning curb
HELPA
FORTM,10,MH1(PH1,3),PH1,PB4,0
PB4
NE
TEST
0
DCARD
Fall through if deplaning
8,VSDBPKD,PH
ASSIGN
One less than first deplaning curb storage no.
8,VSDBPKD,PH
ASSIGN
Save curb storage no. minus one
PH8,PH
ASSIGN
One less than first double parking storage no. for deplaning
PB10
NE
TEST
4
DCARB
If circulating vehicle, transfer to DCARB
PP



One less than corresponding storage no.

One less than enplaning curb storage no.

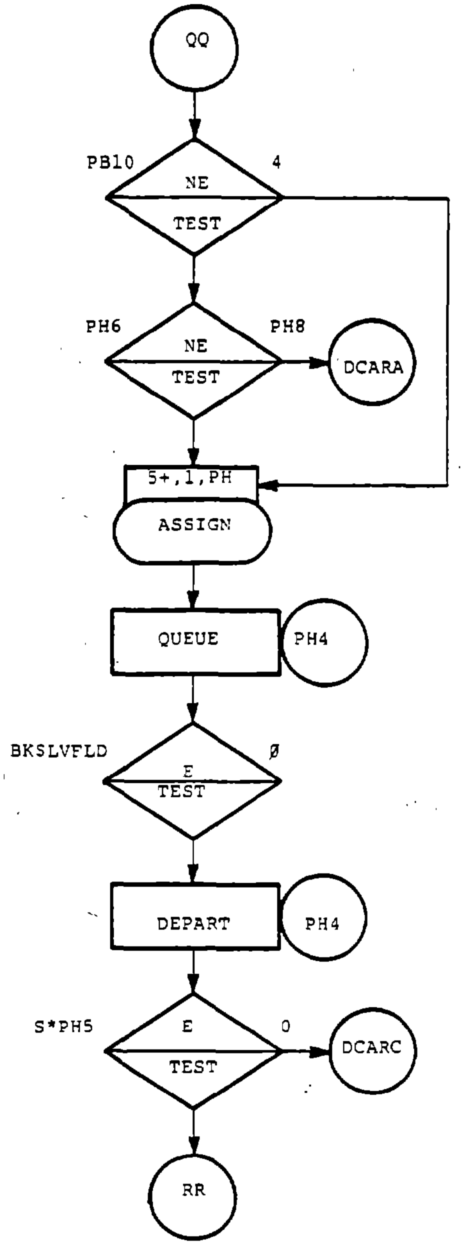
Save curb storage no. minus one

Double parking storage no. for enplaning

If circulating vehicle, transfer to DCARB

One less than corresponding enplaning storage no.

Increment storage no.



If circulating vehicle skip next test

If storage number same as destination storage number, go to DCARA

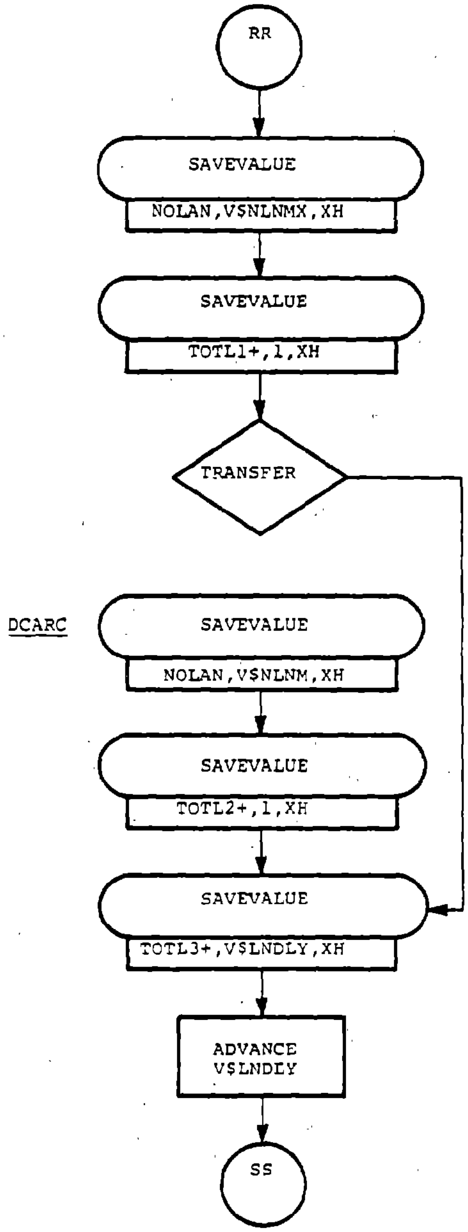
Increment double parking storage number

Enter queue for blocked lane

Stop traffic if double parking and no. of passing lanes available is one

Depart queue for blocked lane

Fall through if current contents of double park storage is zero



Set no. of passing lanes to
maximum available

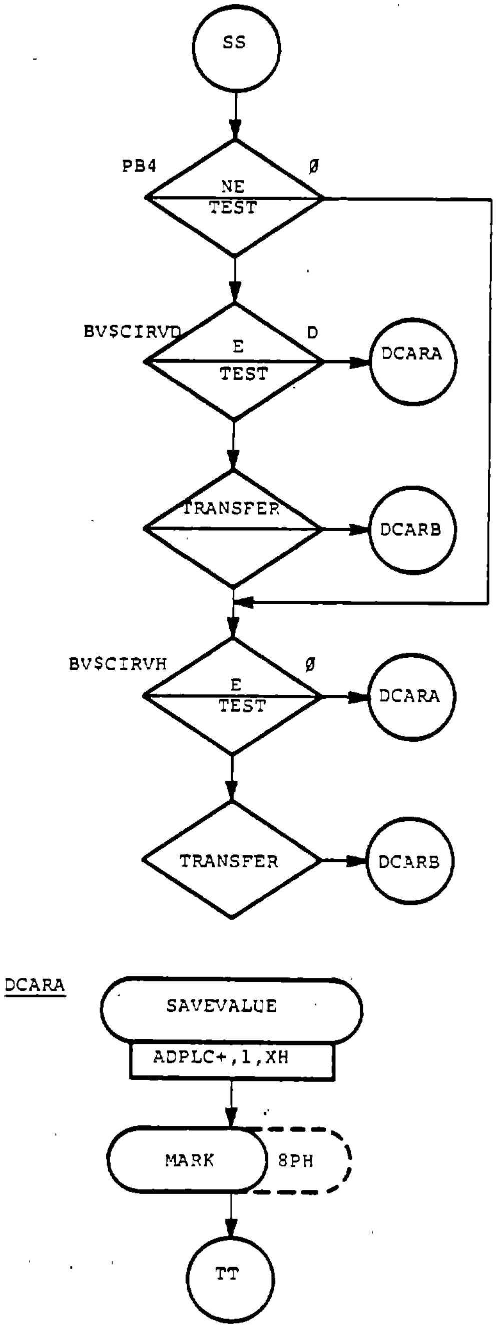
Save count

Set no. of passing lanes to
one less than maximum available

Save count

Save time

Delay vehicle according to
no. of lanes available



Fall through if deplaning

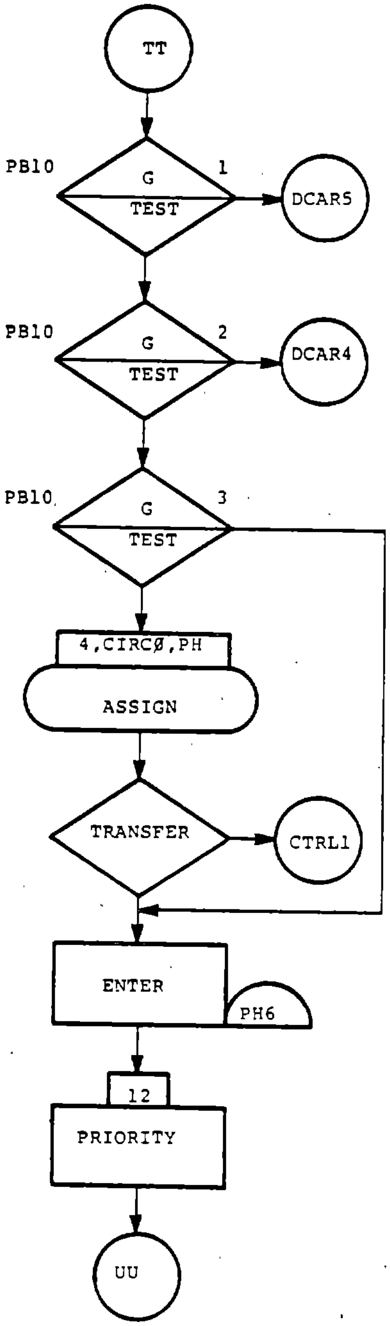
If circulating vehicle is at last deplaning storage transfer to DCARA

Move to next deplaning storage

If circulating vehicle is at last enplaning storage, transfer to DCARA

Count of vehicles on deplaning curb

For time limit enforcement



Branch to enter curb slot

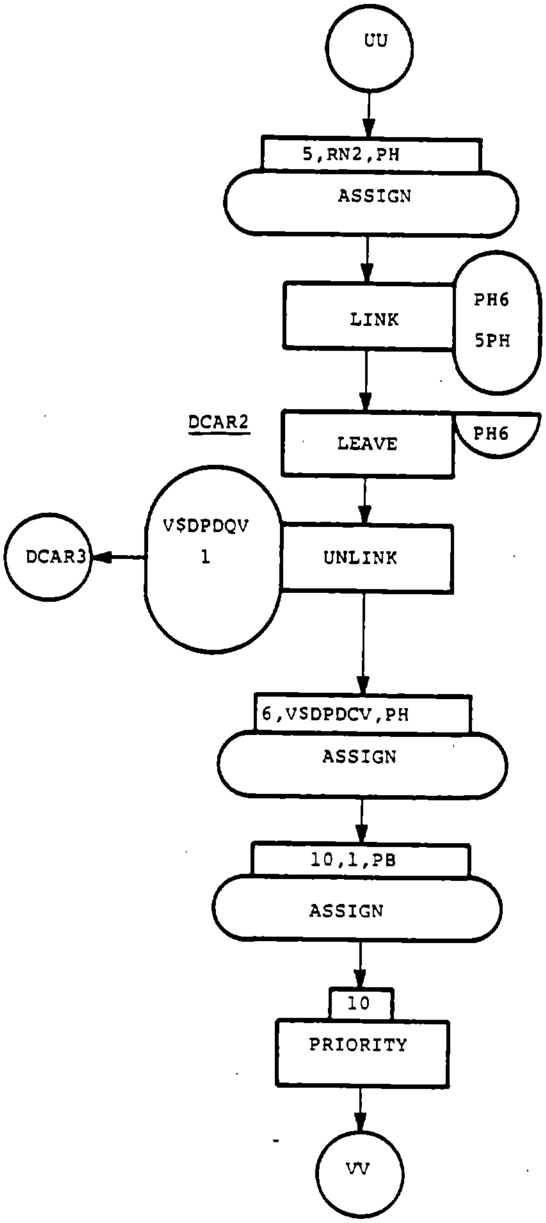
Branch to enter double parking slot

Branch to enter queue slot

Send to recirculation and

Enter Queue slot

To assure priority for DP slot



Get random no.

Link on queue in random order

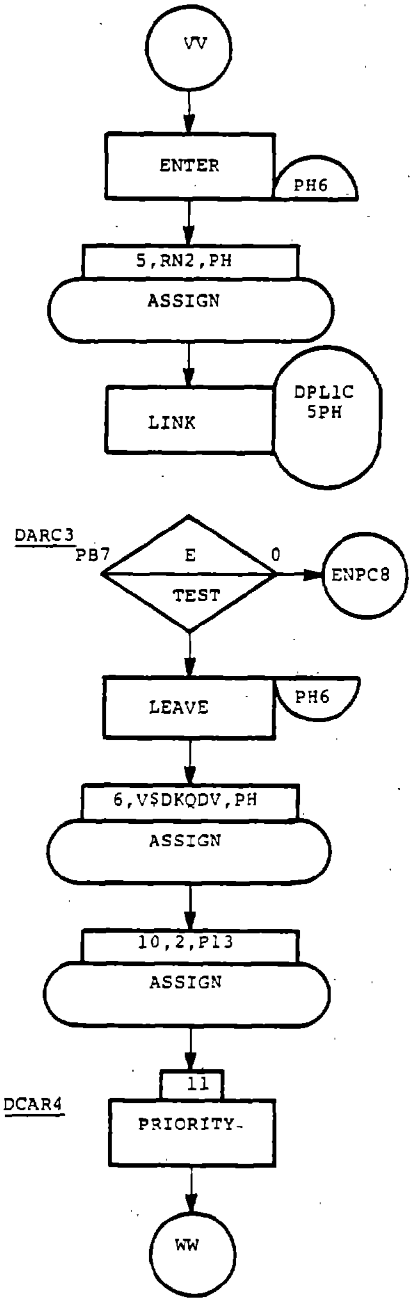
Leave double parking storage

Unlink one from corresponding queue chain

Change PH6 to curb storage no.

Now at curb

Drop to normal priority



Enter curb storage

Get random no.

Relink in random order

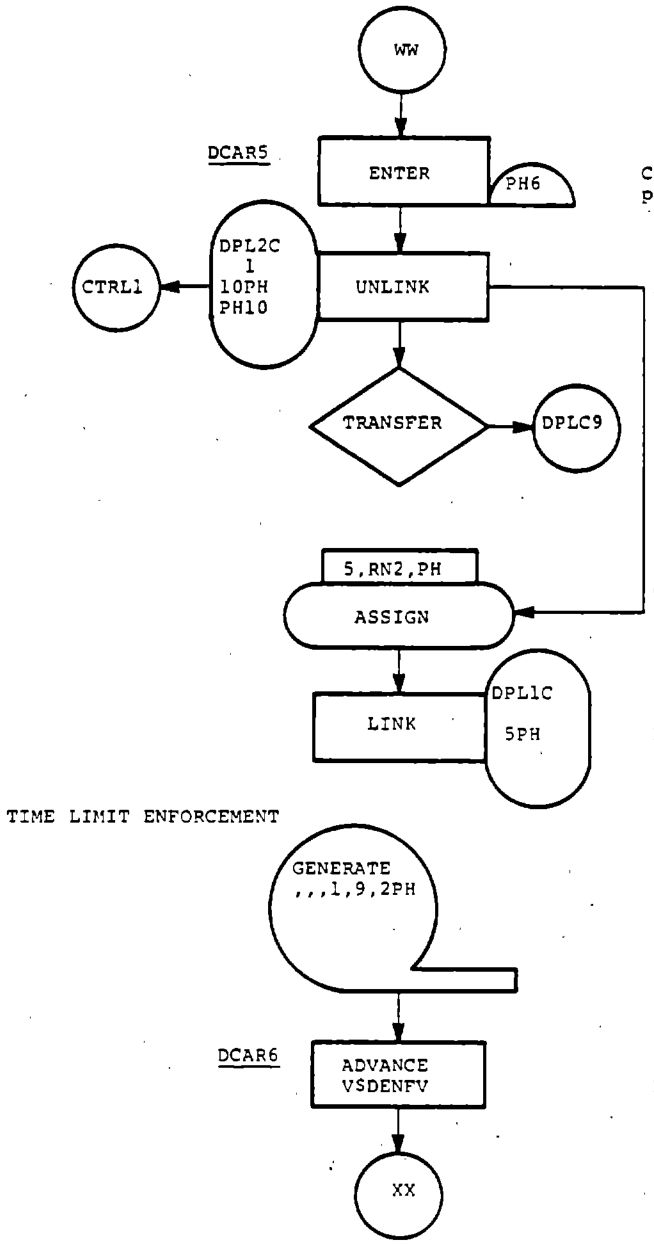
Branch if vehicle unlinked
is enplaning

Leave queue storage

Change PH6 to Double
parking storage no.

Now using double parking

Set double parking priority



Curb or double
parking storage

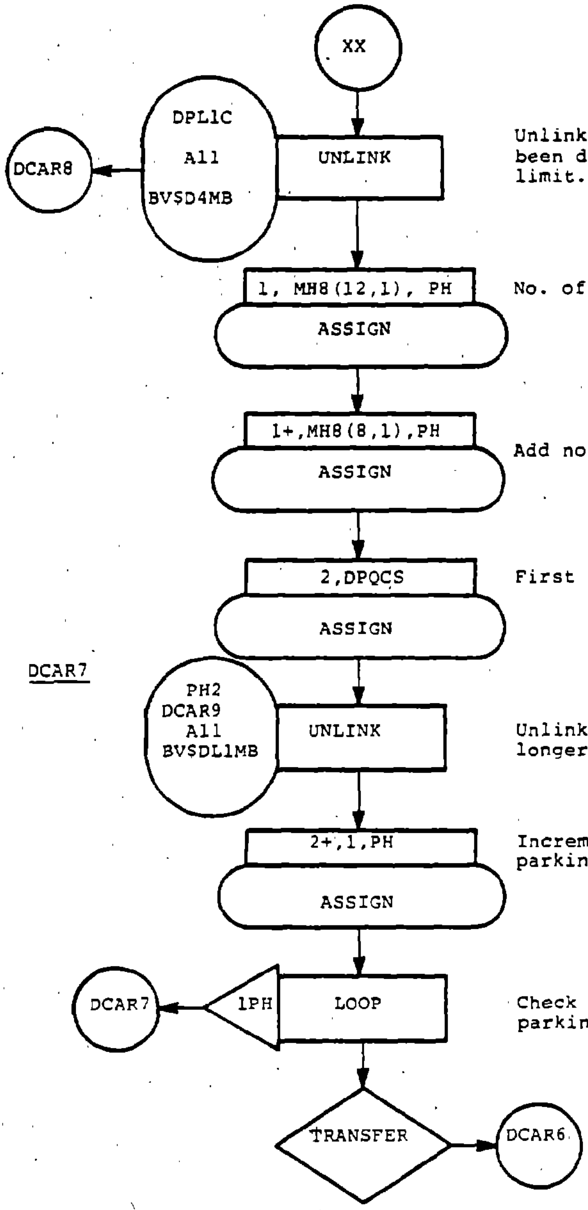
Match car with pax

Get random no.

Link in random order

'Police Officer'

Frequency of enforcement



Unlink any vehicles which have been double-parked longer than limit.

No. of deplaning curbs

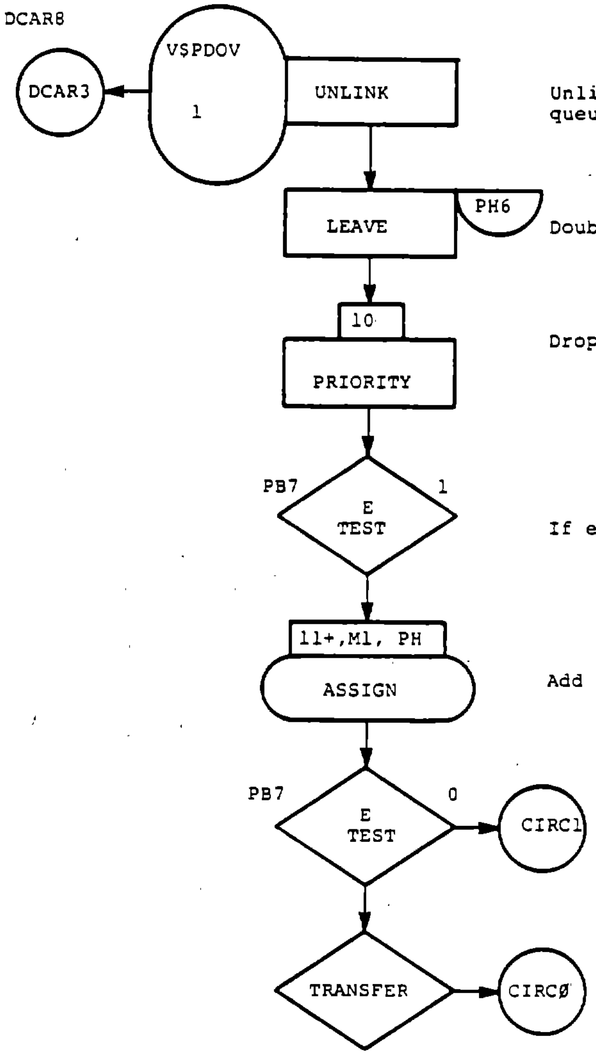
Add no. of deplaning curbs

First queue chain

Unlink any vehicles in queue longer than limit

Increment curb or double parking storage no.

Check next curb or double parking storage



Unlink one from corresponding queue chain

Double or queue storage

Drop to normal priority

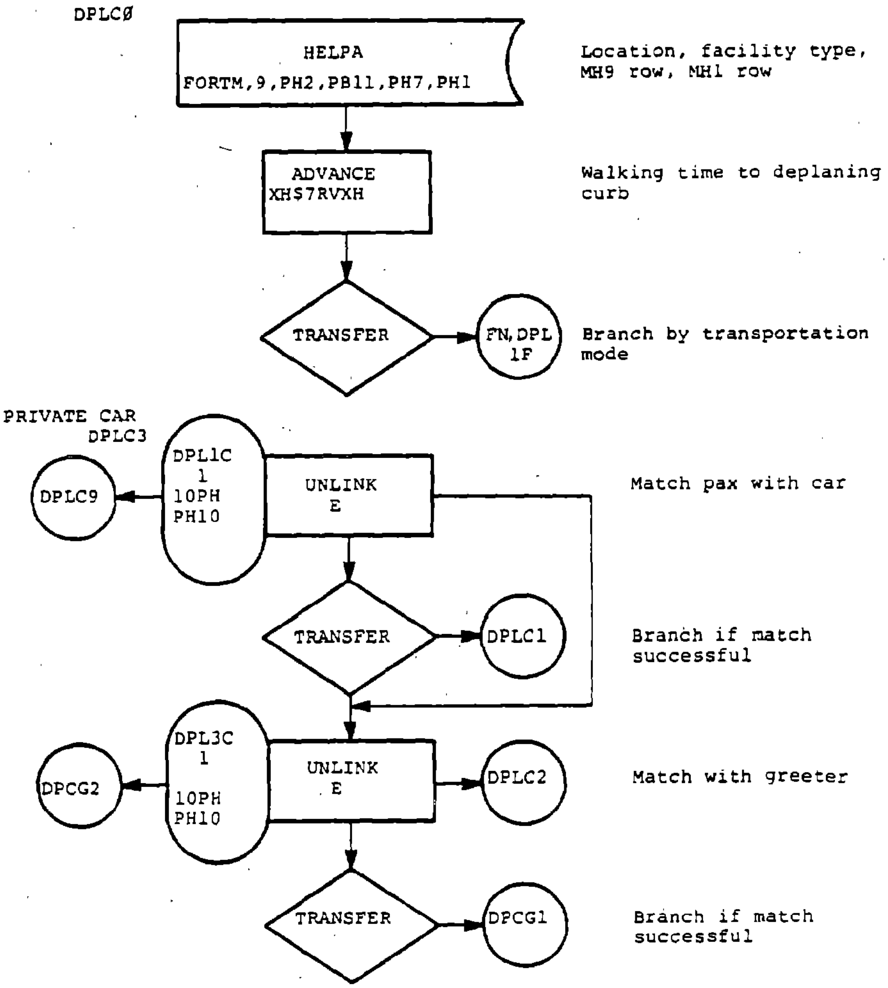
If enplaning, fall through

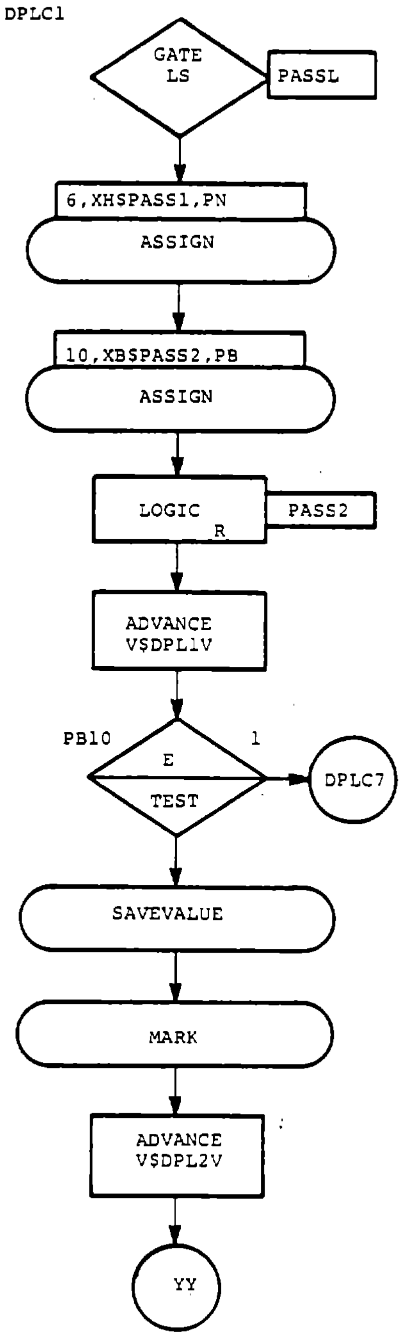
Add waiting time to PH11

Branch if unlinked vehicle is enplaning

Recirculate

DEPLANING CURB (PAX)





Wait for car location information

Pickup

Car location information

Reset logic switch

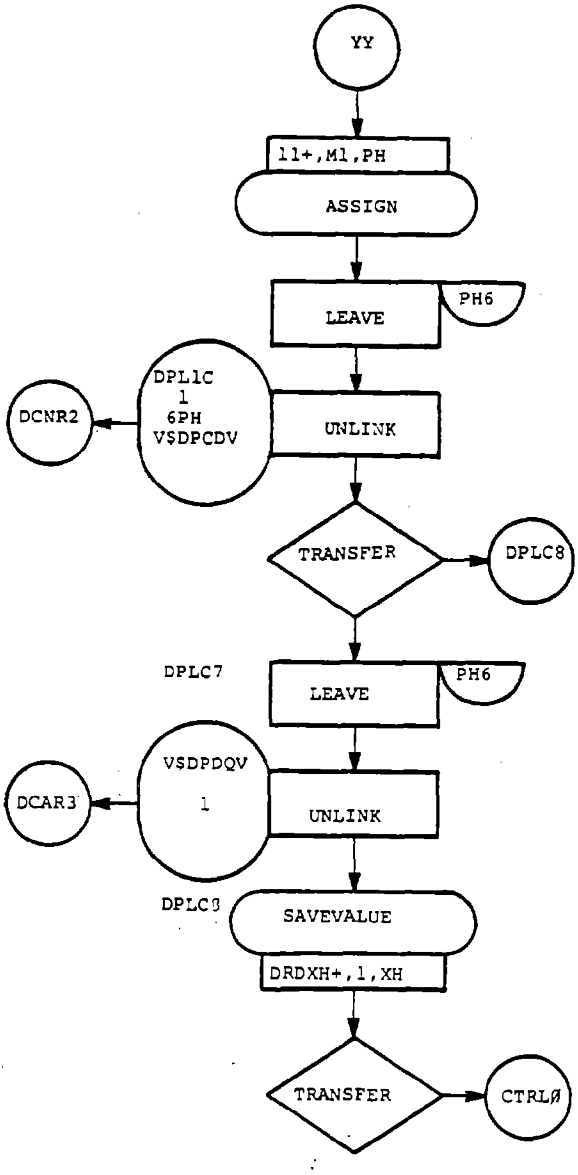
Loading time

Fall through if leaving curb

Indirect addressing using variable

Mark pax for waiting time

Random pullout delay

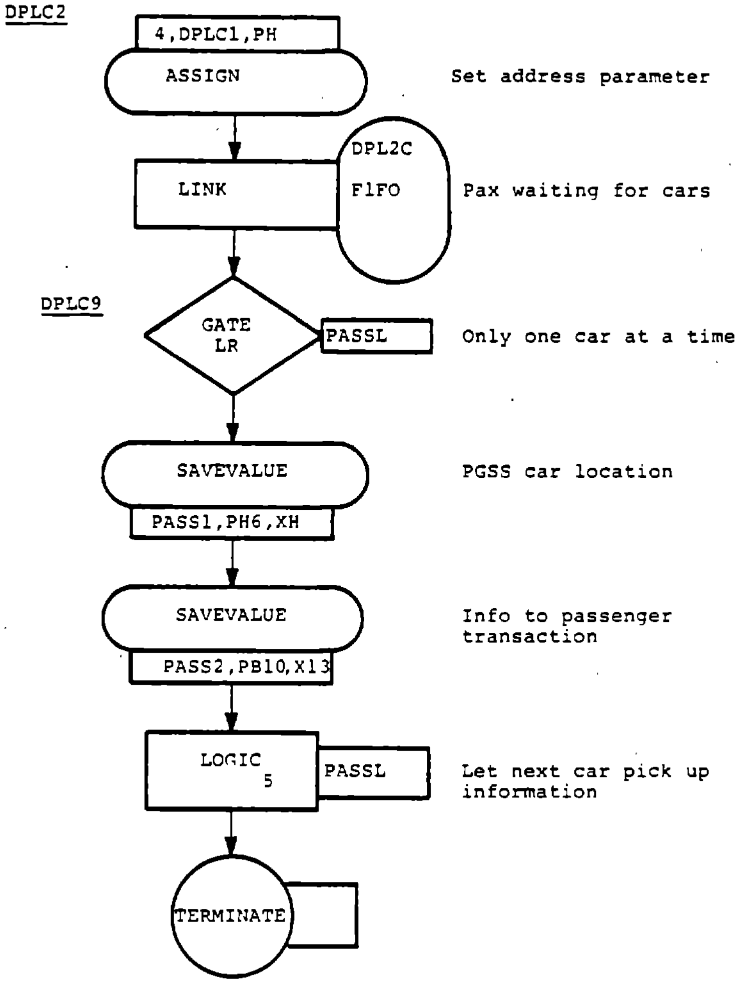


Add waiting time to PH11

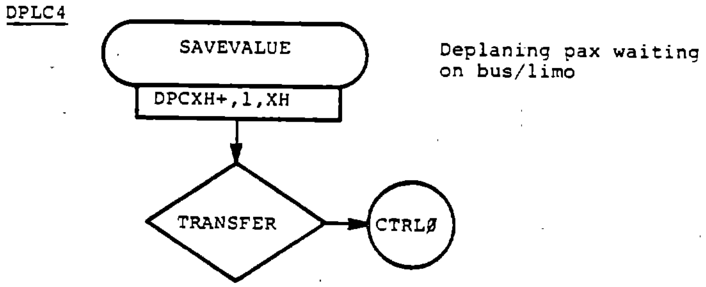
Unlink 1 from chain with PH6 equal to corresponding double part storage

Unlink one from corresponding queue chain

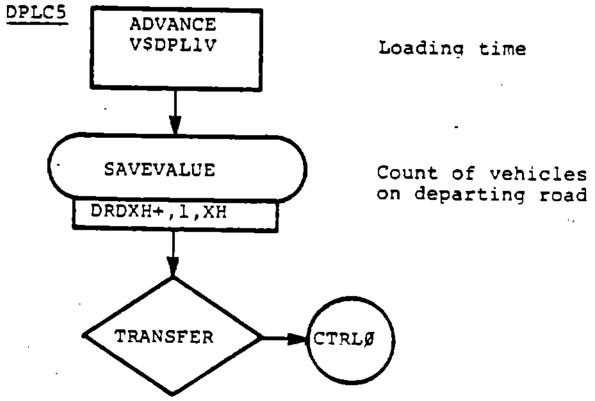
Count of vehicles on departing road


DPLC2
4,DPLC1,PH
ASSIGN
Set address parameter
LINK
DPL2C
F1FO
Pax waiting for cars
DPLC9
GATE
LR
PASSL
Only one car at a time
SAVEVALUE
PASS1,PH6,XH
PGSS car location
SAVEVALUE
PASS2,PB10,X13
Info to passenger transaction
LOGIC
5
PASSL
Let next car pick up information
TERMINATE

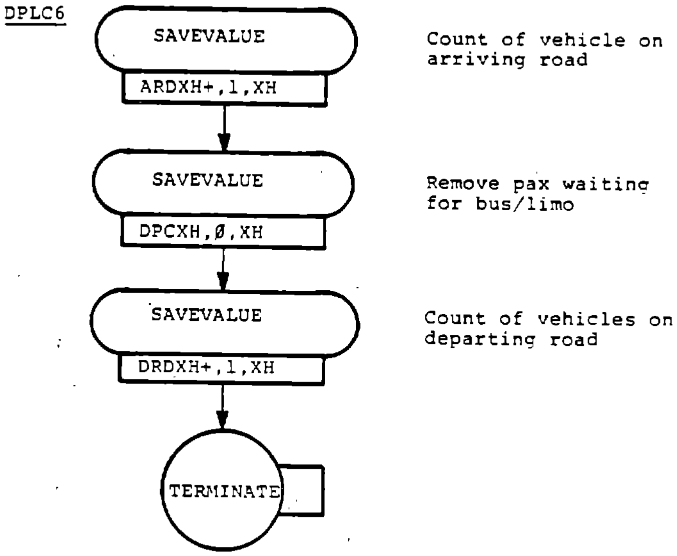
BUS/LIMO - PAX



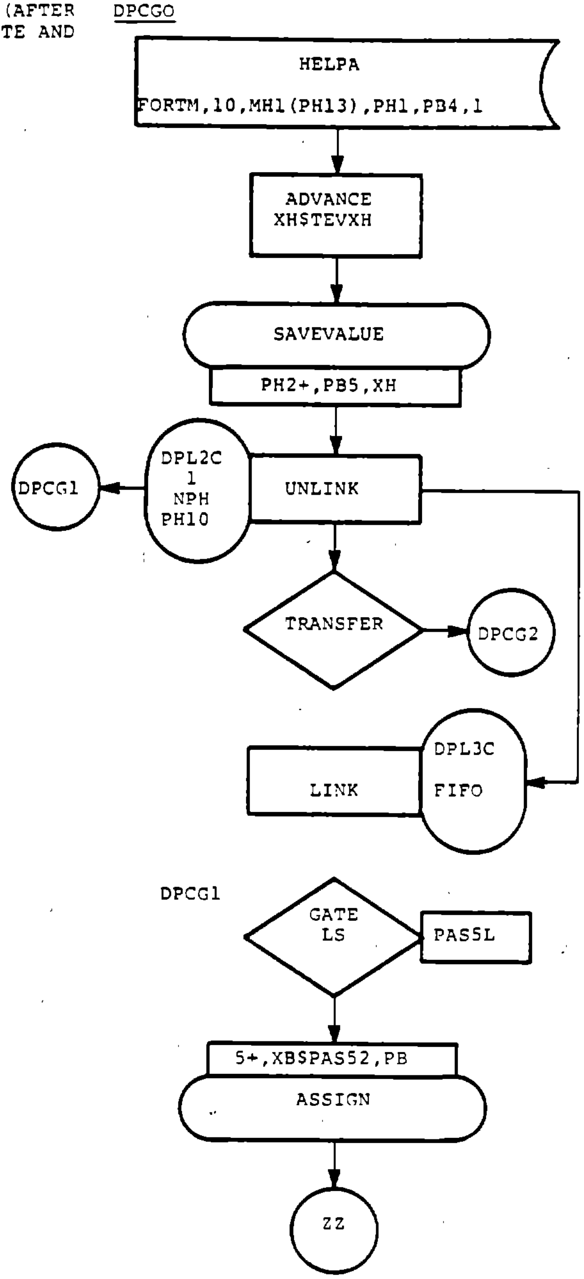
TAXI



BUS/LIMO-VEHICLE


DPLC6
SAVEVALUE
ARDXH+,1,XH
Count of vehicle on arriving road
SAVEVALUE
DPCXH,Ø,XH
Remove pax waiting for bus/limo
SAVEVALUE
DRDXH+,1,XH
Count of vehicles on departing road
TERMINATE

DEPLANING CURB
GREETERS (AFTER
RECIRCULATE AND
PARK)



Airline, flight,
no. of bags, greeter

Walking time for
greeter to deplaning
curb.

Congestion

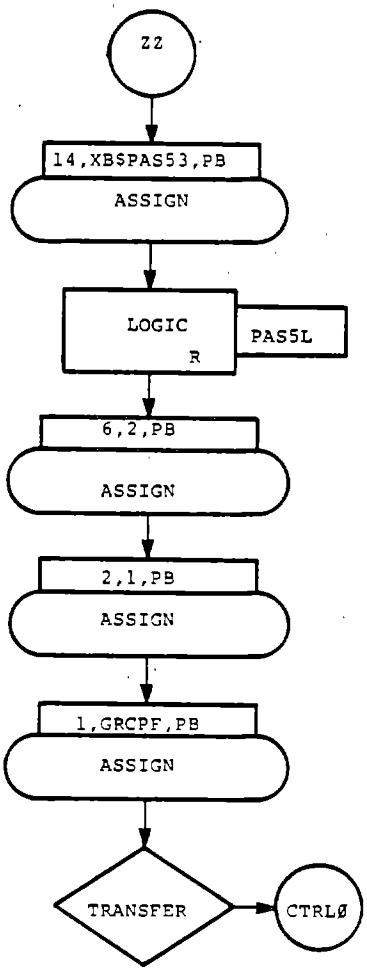
Match pax at curb

Branch if successful

Otherwise link to
special chain

Wait for information
from greeter

Add greeters to party



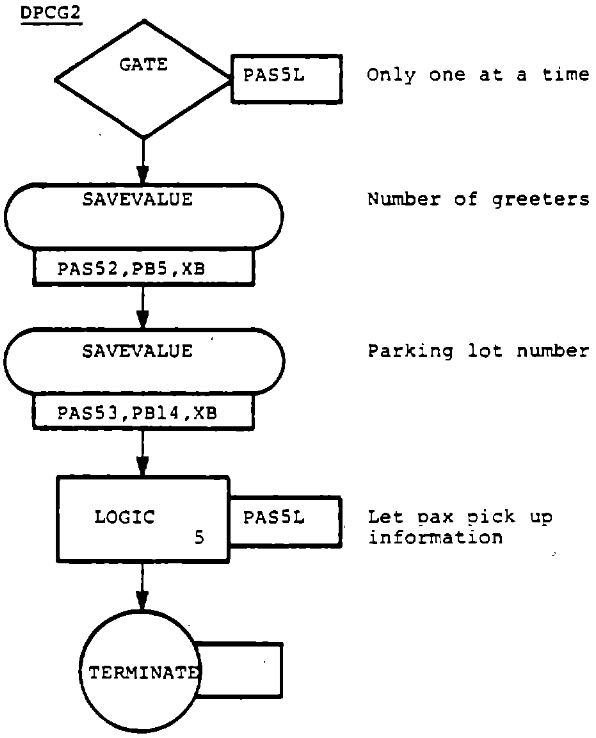
Pick up parking lot number

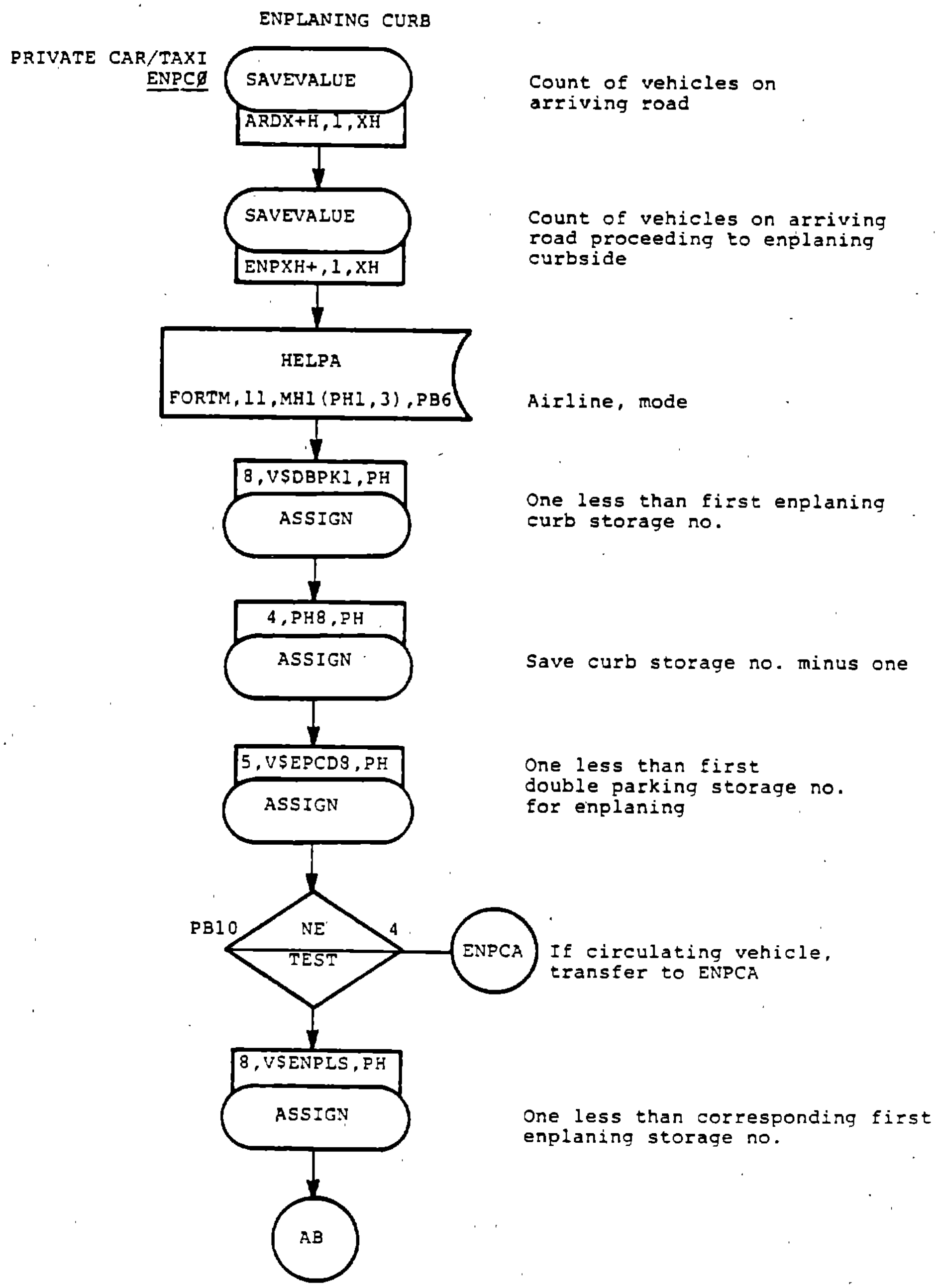
Permit next use of PAS5 savevalue

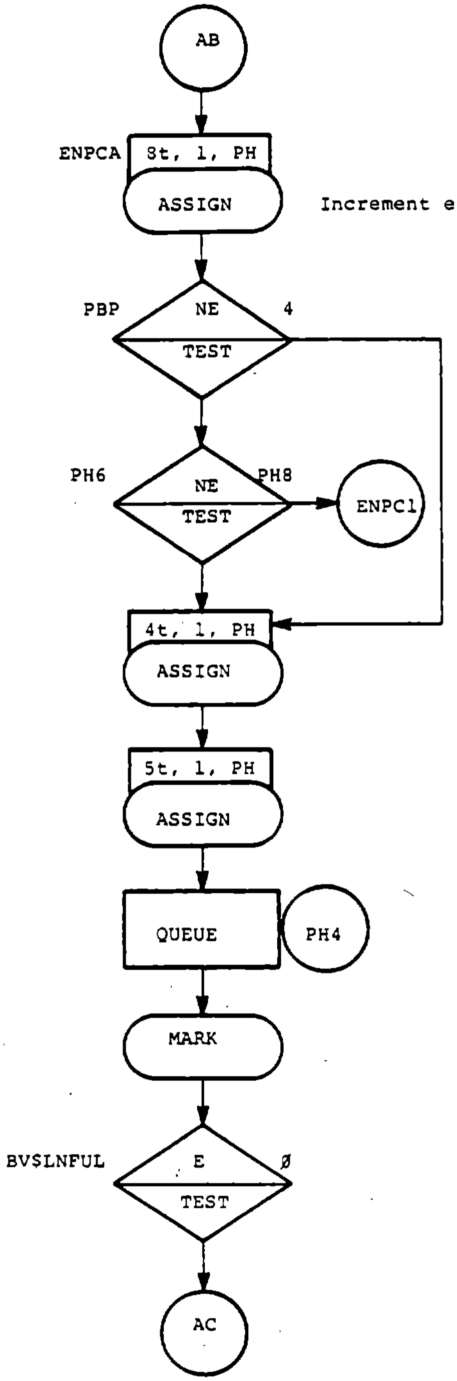
Change mode to parking

Process function pointer

New process function


DPCG2
GATE
PAS5L
Only one at a time
SAVEVALUE
PAS52,PB5,XB
Number of greeters
SAVEVALUE
PAS53,PB14,XB
Parking lot number
LOGIC
5
PAS5L
Let pax pick up information
TERMINATE


ENPLANING CURB
PRIVATE CAR/TAXI
ENPCØ
SAVEVALUE
ARDX+H,1,XH
Count of vehicles on arriving road
SAVEVALUE
ENPXH+,1,XH
Count of vehicles on arriving road proceeding to enplaning curbside
HELPA
FORTM,11,MH1(PH1,3),PB6
Airline, mode
8,V$DBPK1,PH
ASSIGN
One less than first enplaning curb storage no.
4,PH8,PH
ASSIGN
Save curb storage no. minus one
5,V$EPCD8,PH
ASSIGN
One less than first double parking storage no. for enplaning
PB10
NE
TEST
4
ENPCA
If circulating vehicle, transfer to ENPCA
8,V$ENPLS,PH
ASSIGN
One less than corresponding first enplaning storage no.
AB



Increment enplanning storage no.

If circulating vehicle,
skip next test

Is enplanning storage no.
same as destination storage no.

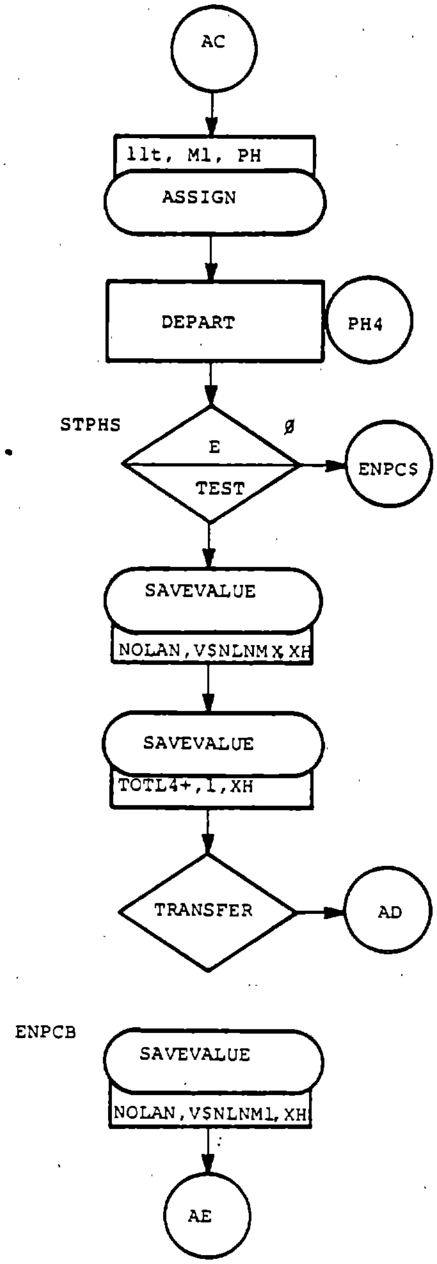
Increment enplaning curb
storage no.

Increment enplaning double
parking storage no.

Enter queue for blocked lane

Mark pax for waiting time

Stop traffic if double parking
and no. of passing lanes
available is one.



Add waiting time to PH11

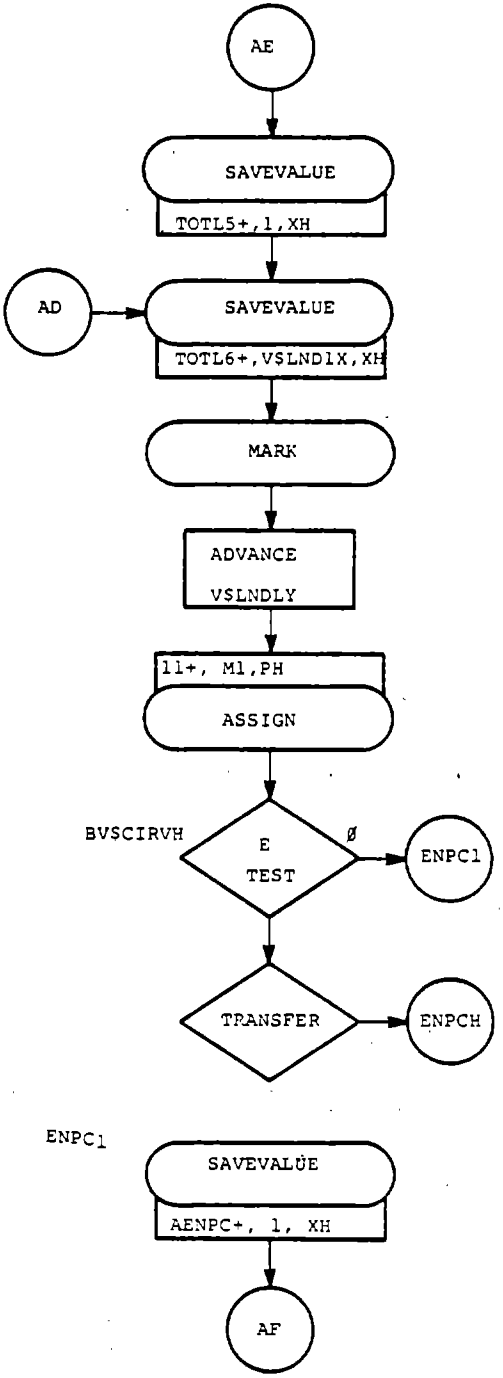
Leave queue for blocked lane

If current contents of double parking storage is fall through through

Set no. of passing lanes to maximum available.

Save count

Set no. of passing lanes to one less than maximum available.



Save Count

Save time

Mark pax for waiting time

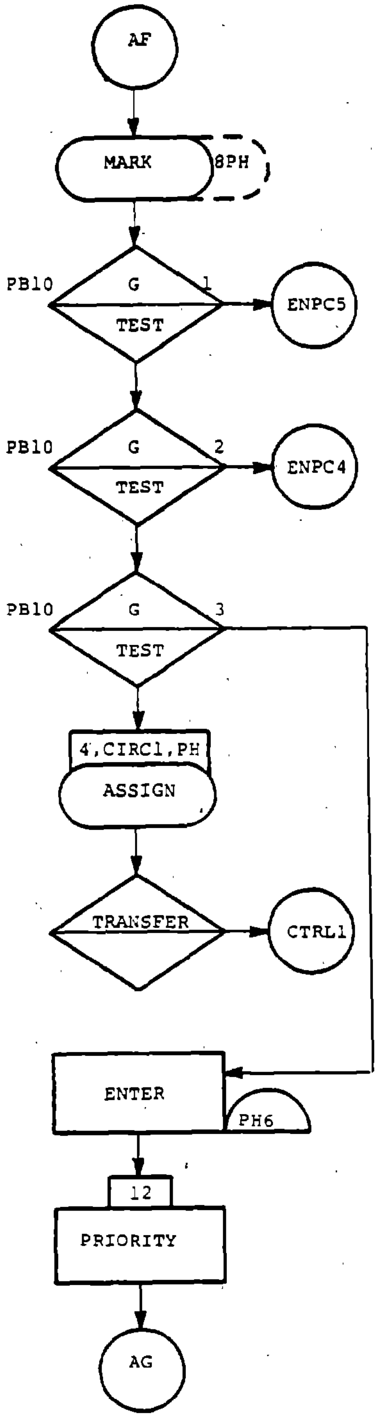
Delay vehicle according
to no. of lanes available

Add waiting time to PH11

If calculating vehicle is
at last enplaning curb
storage, transfer to ENPC1

Move to next enplaning
storage no.

Count of vehicles on
enplaning curb



For time limit enforcement

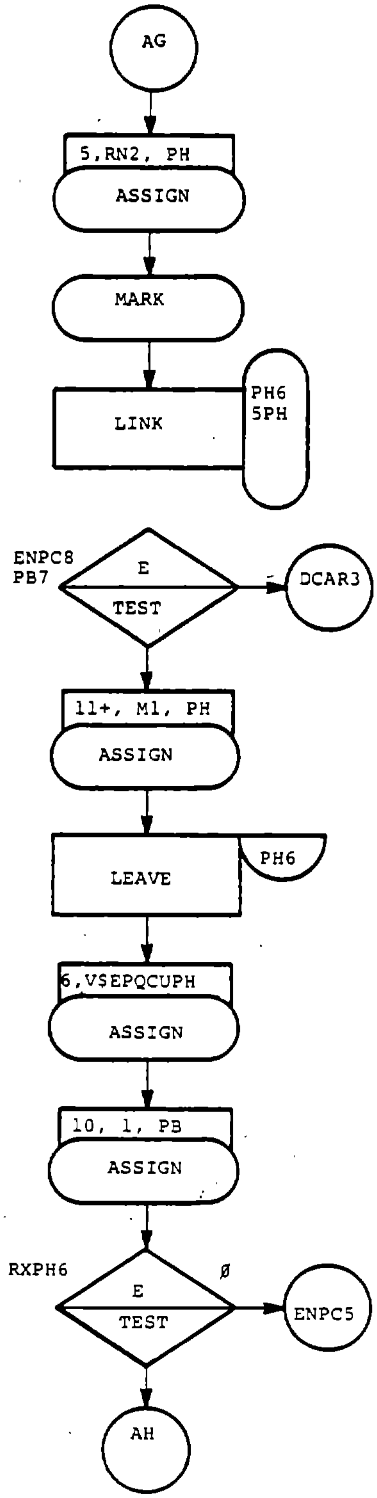
Branch to enter curb slot

Branch to enter double parking slot

Branch to enter queue slot

Sent to recirculation road

To assure priority for double parking slot



Get random number

Mark pax for waiting time

Link on queue in random order

Branch if vehicle unlinked is deplaning

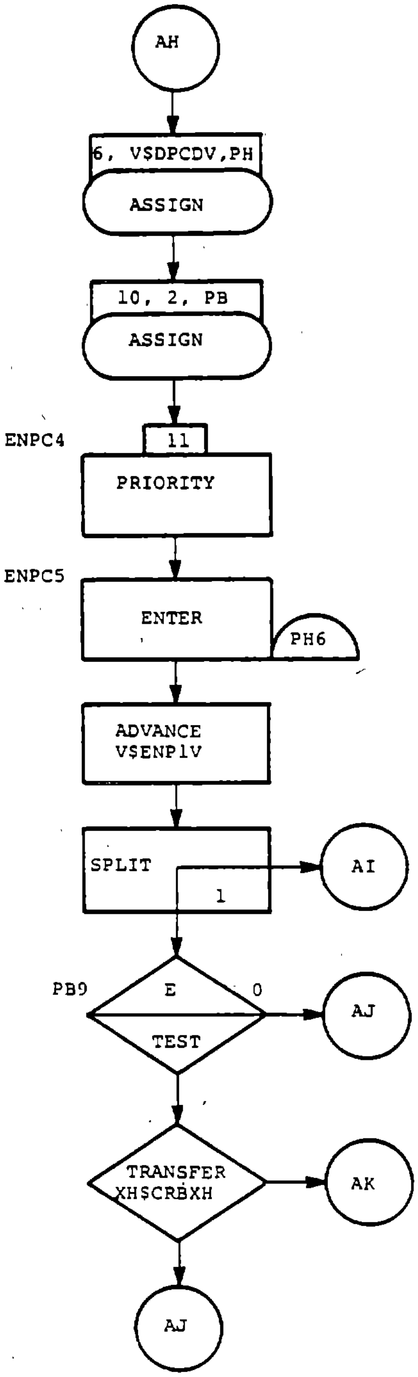
Add waiting time to PH11

Leave queue storage

Change PH6 to curb storage no.

Using curb

Fall through if storage filled



Change PH6 to double parking storage

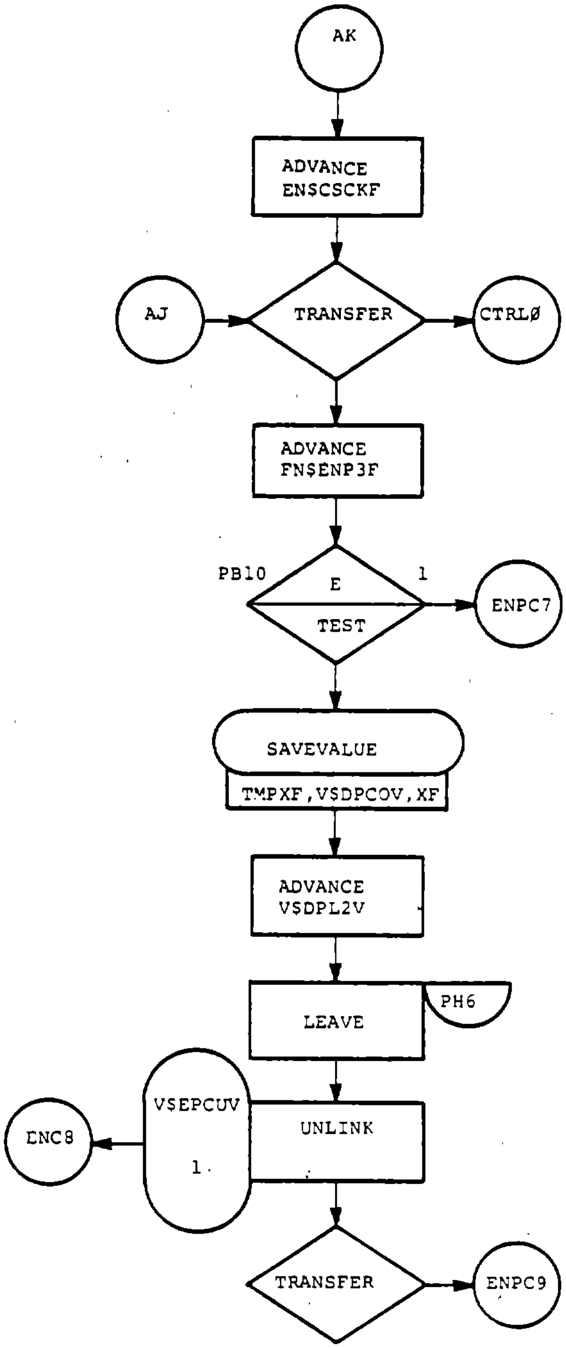
Using double parking

Double parking priority

Enplaning curb unload time - car/taxi

Fall through if preticketed

Fall through if using curb side check-in



Curb side check-in time

Empty car parking time

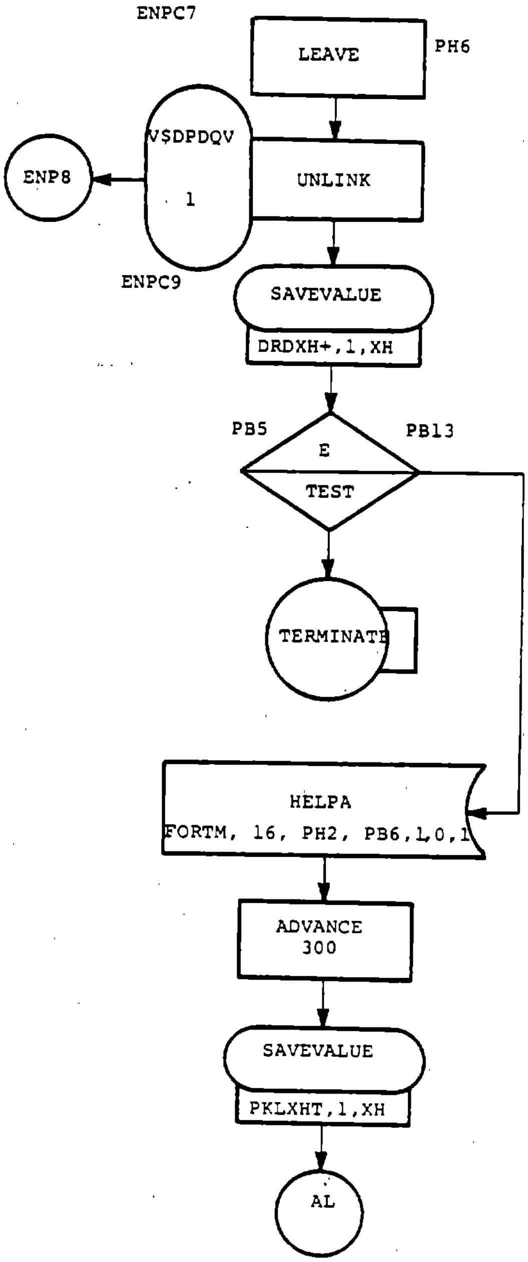
Fall through if leaving curb

Indirect addressing using variable

Random pullout delay

Curb storage

Unlink one from queue



Double parking storage

Unlink one from queue

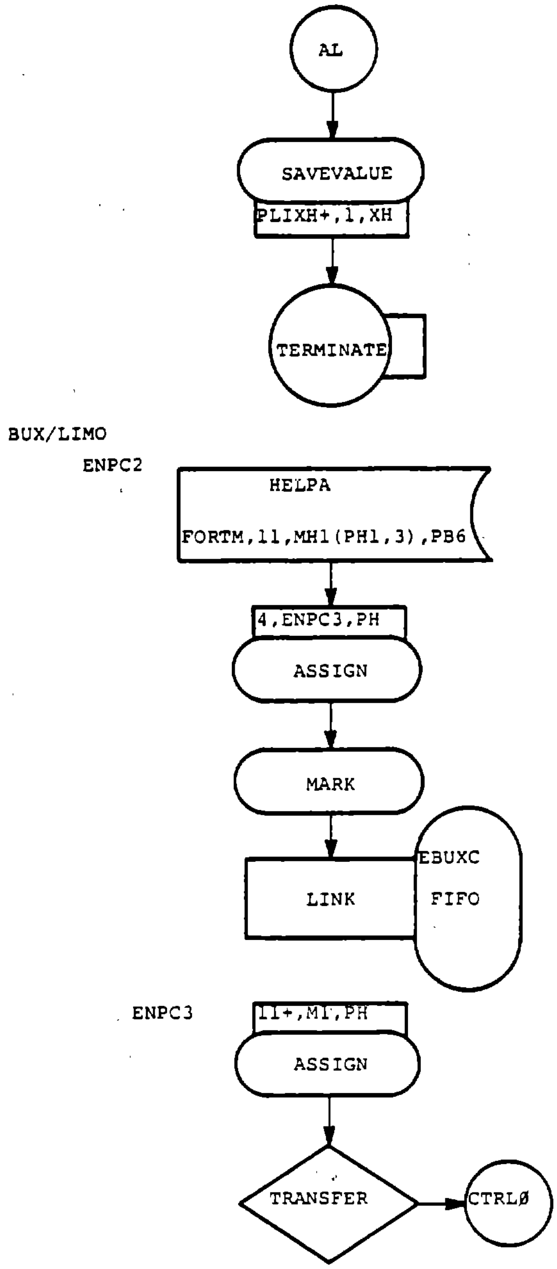
Count of vehicles on departing road

Fall through if no well-wishers

Location, mode, enplaning, zero, Lot no. only

Vehicle to lot

Enter car in parking lot



Vehicle count in parking lot

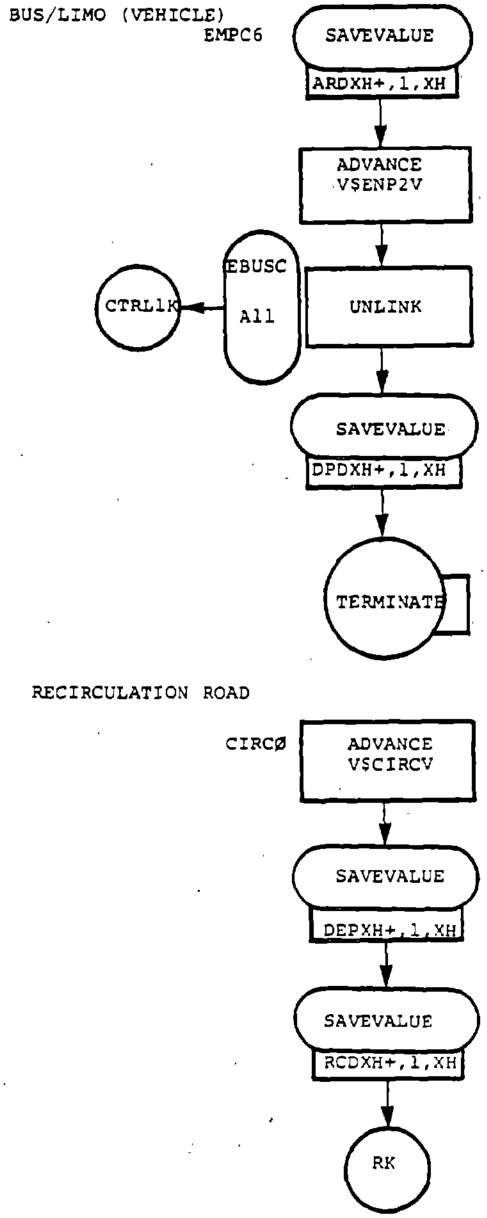
Air

Assign address

Mark pax for waiting time

Link to enplaning bus
user chain

Add waiting time to PH11



Count of vehicles on arriving road

Bus/Limo load time

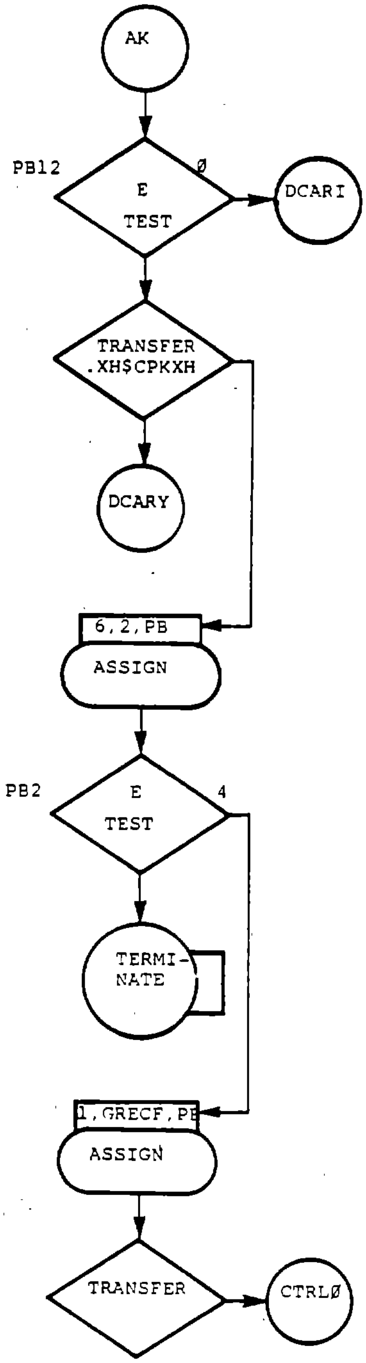
Bus leaves curbside

Count of vehicles on departing road

Recirculate

Include recirculation in deplaning curb counts

Recirculation to deplaning

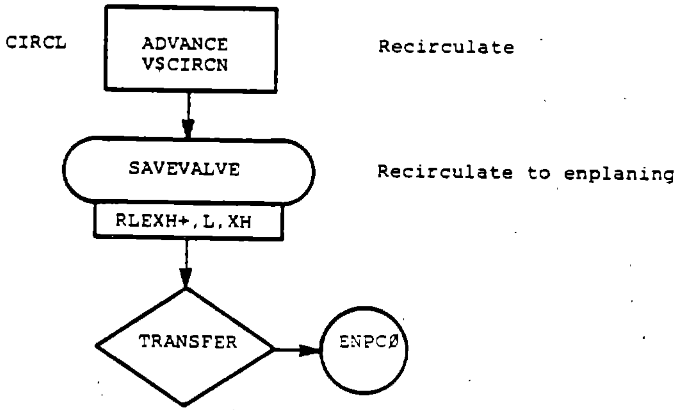


Branch N greater
who has all ready met

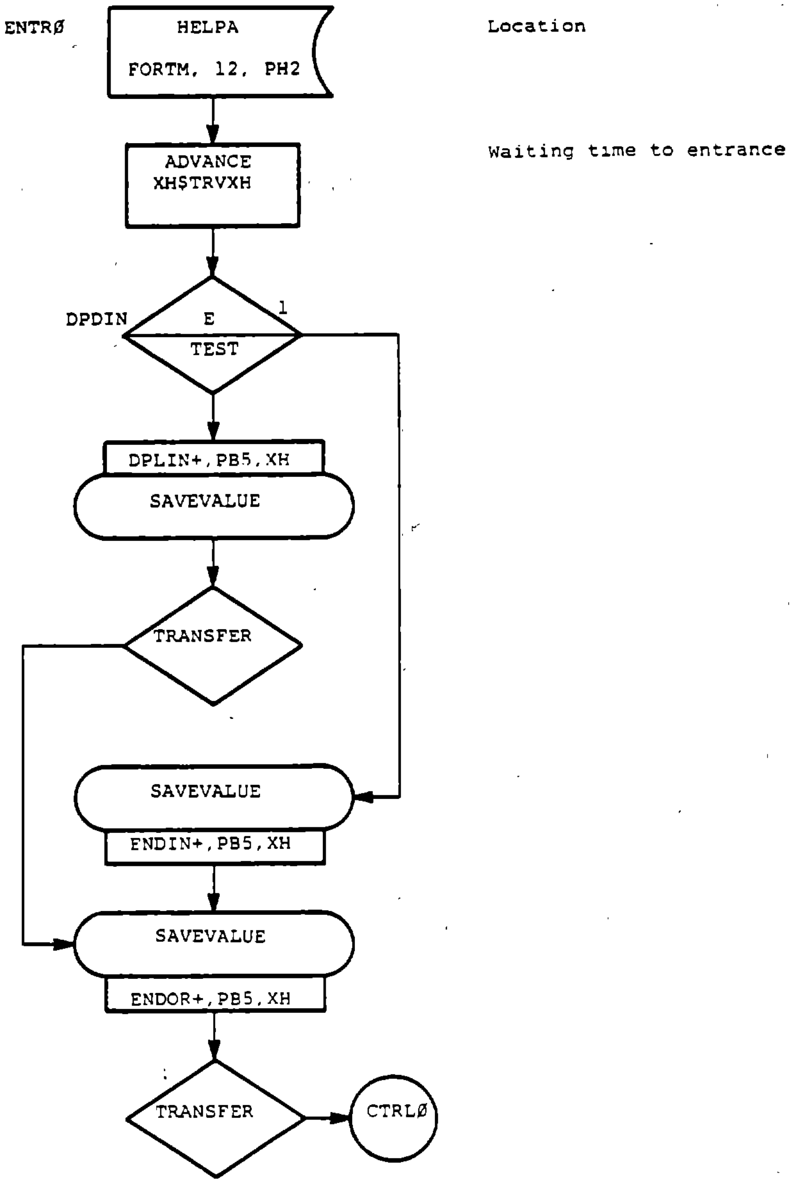
Fall through to enter park
or branch to deplaning road

Change mode to parking

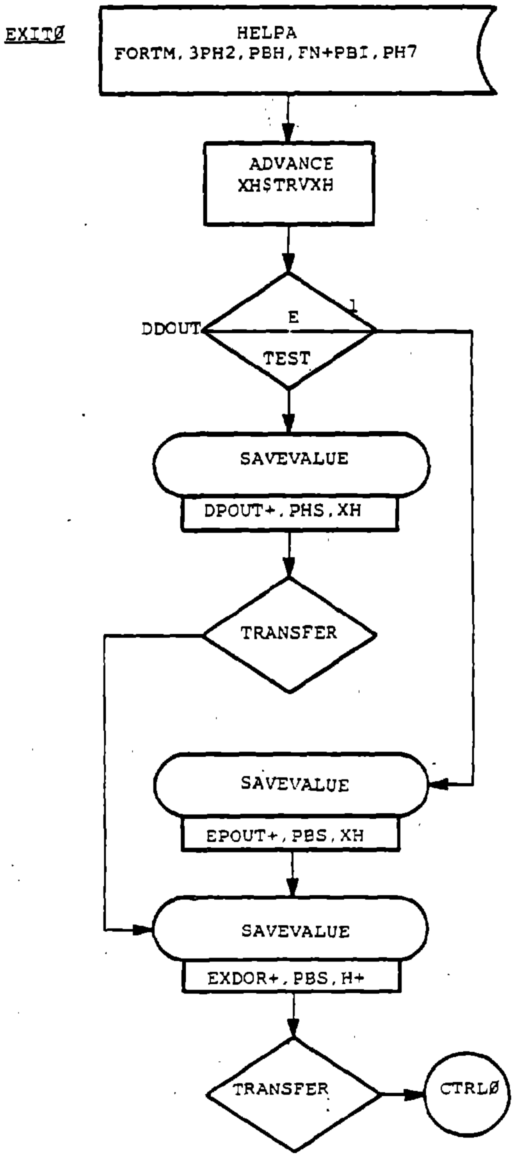
Process function to curb


CIRCL
ADVANCE
V$CIRCN
Recirculate
SAVEVALVE
Recirculate to enplaning
RLEXH+,L,XH
TRANSFER
ENPCØ

ENTRANCE


ENTRØ
HELPA
FORTM, 12, PH2
Location
ADVANCE
XH$TRVXH
Waiting time to entrance
DPDIN
E
1
TEST
DPLIN+,PB5,XH
SAVEVALUE
TRANSFER
SAVEVALUE
ENDIN+,PB5,XH
SAVEVALUE
ENDOR+,PB5,XH
TRANSFER
CTRLØ

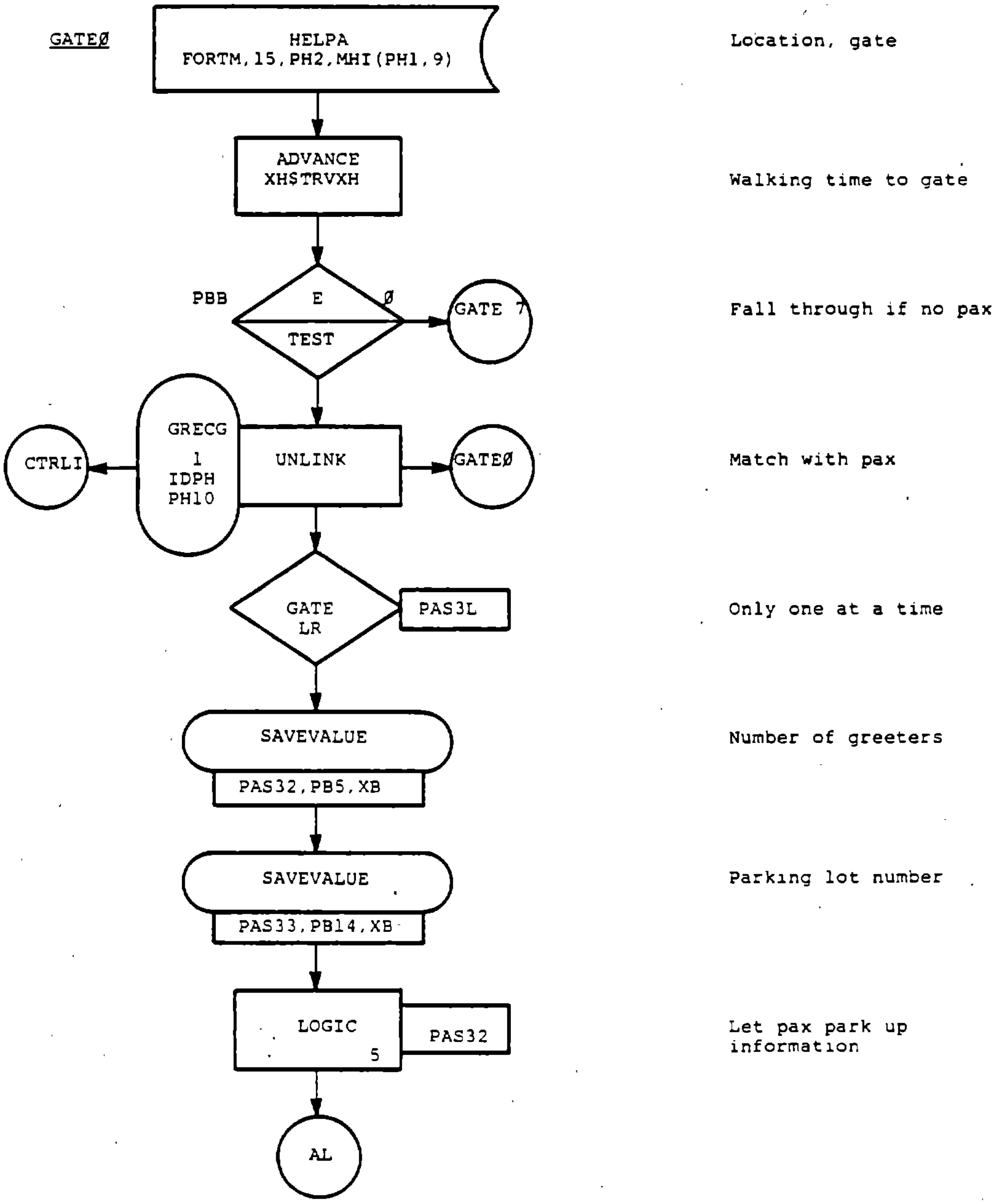
EXIT

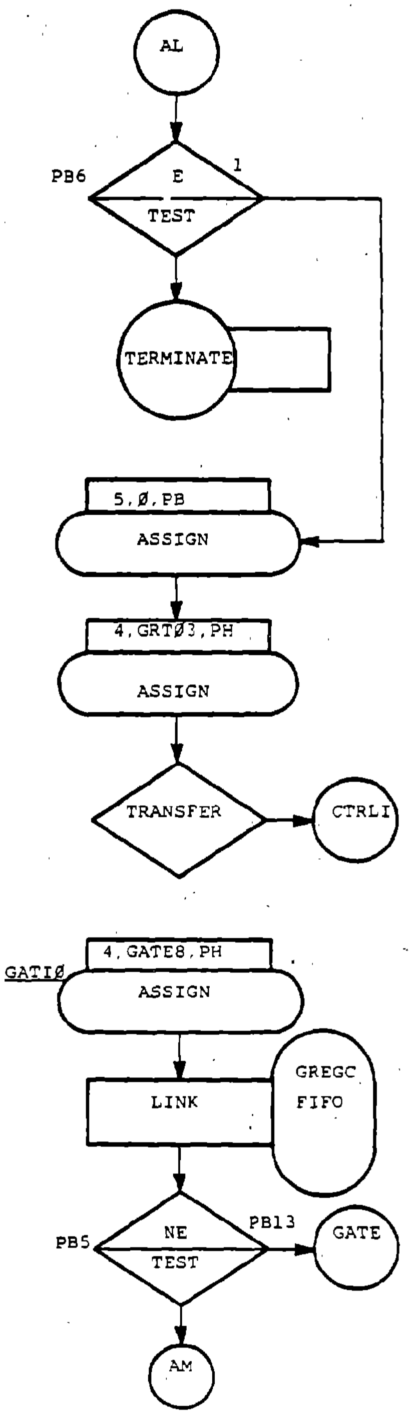


Location, current process,‘
next address, MH9 Row

Walking tıme to exit

GATE (ENPLANING PAX)





Branch H mode is curb

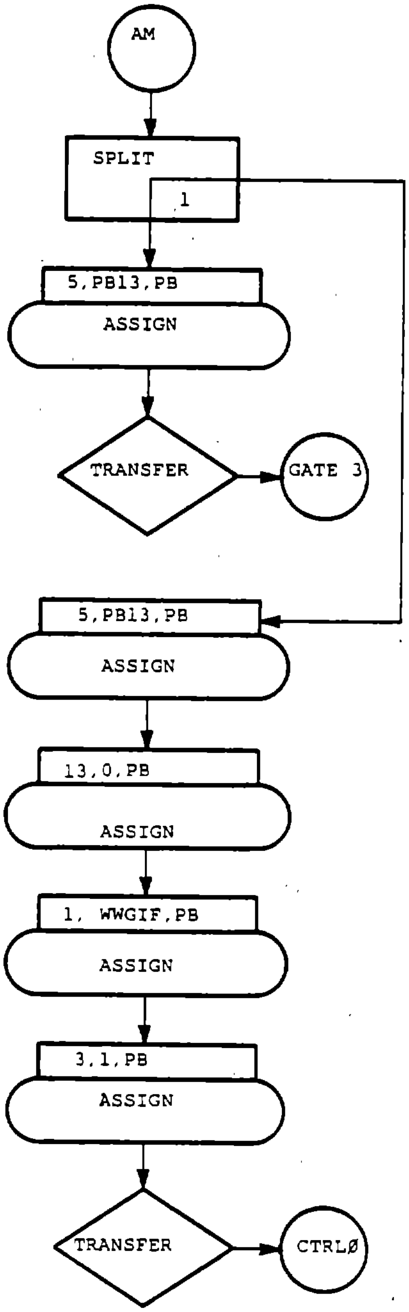
No one in party

Rate to greeter leaving parking

Will go to GATE when unlinked

Wait to meet greeter

Fall through if have well-wishers.



Split well-wishers off

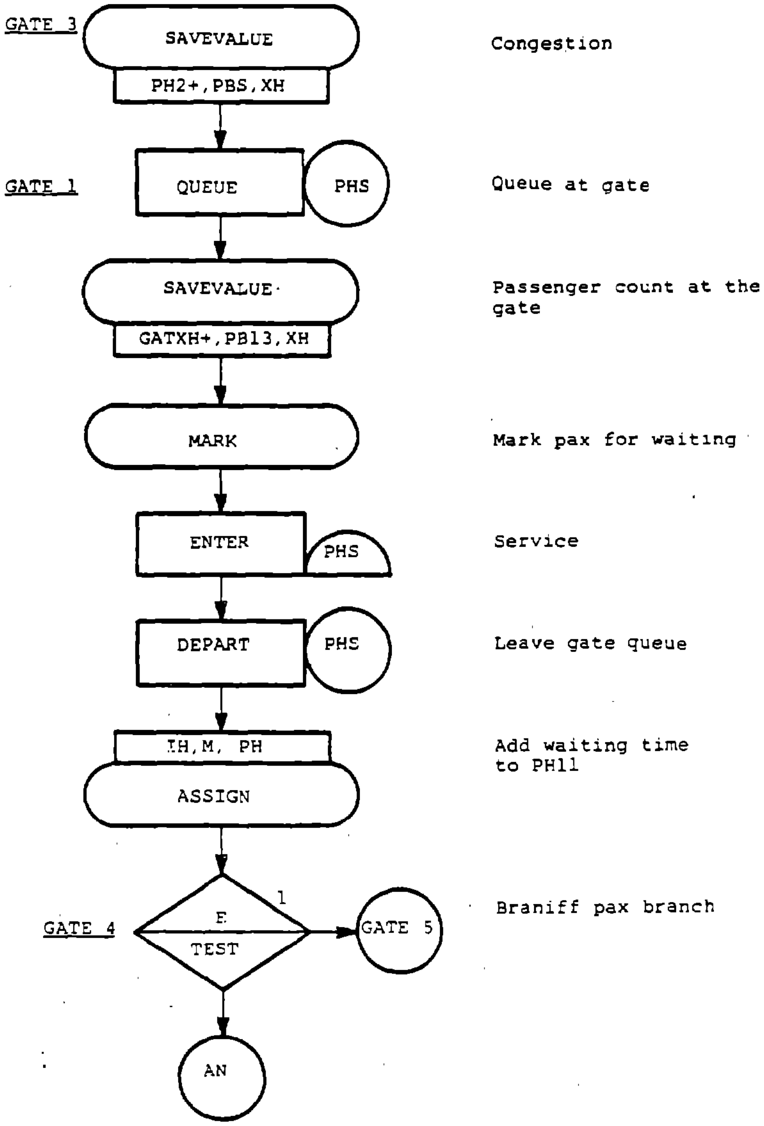
Drop well-wishers

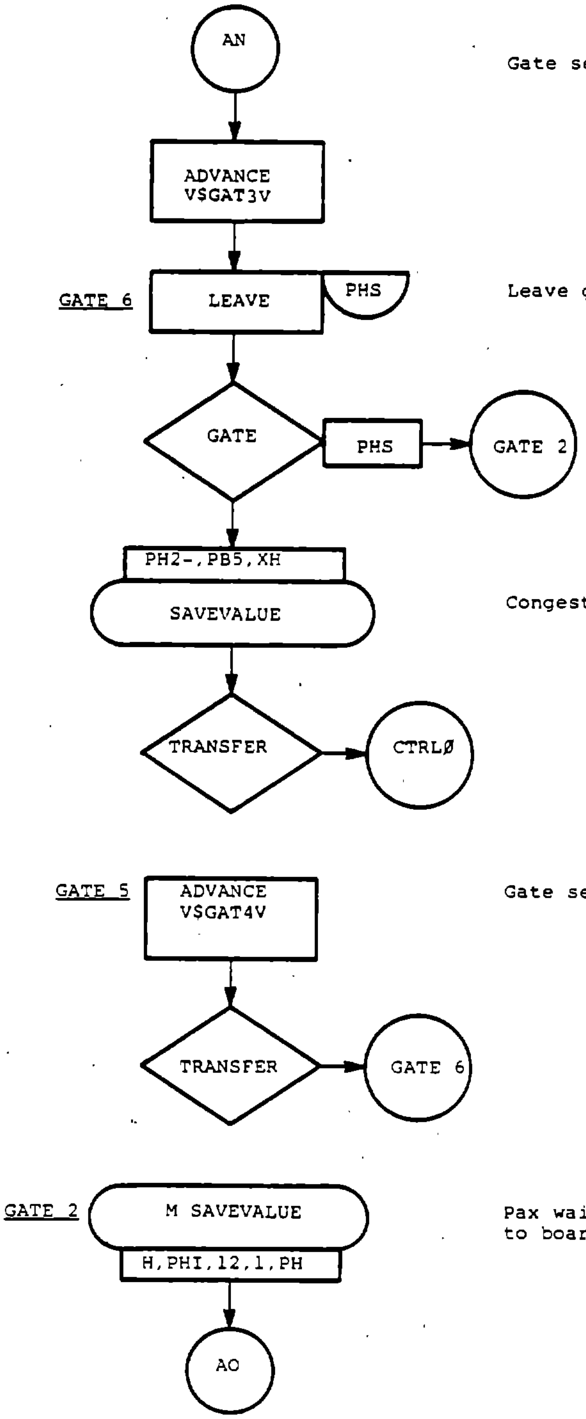
Subtract at Pax

Set pax to zero

Well-wishers leaving from
gate process function

Process function pointer


GATE 3
SAVEVALUE
PH2+,PBS,XH
Congestion
GATE 1
QUEUE
PHS
Queue at gate
SAVEVALUE
GATXH+,PB13,XH
Passenger count at the gate
MARK
Mark pax for waiting
ENTER
PHS
Service
DEPART
PHS
Leave gate queue
IH,M, PH
ASSIGN
Add waiting time to PH11
GATE 4
E
TEST
1
GATE 5
Braniff pax branch
AN



Gate service time-Braniff

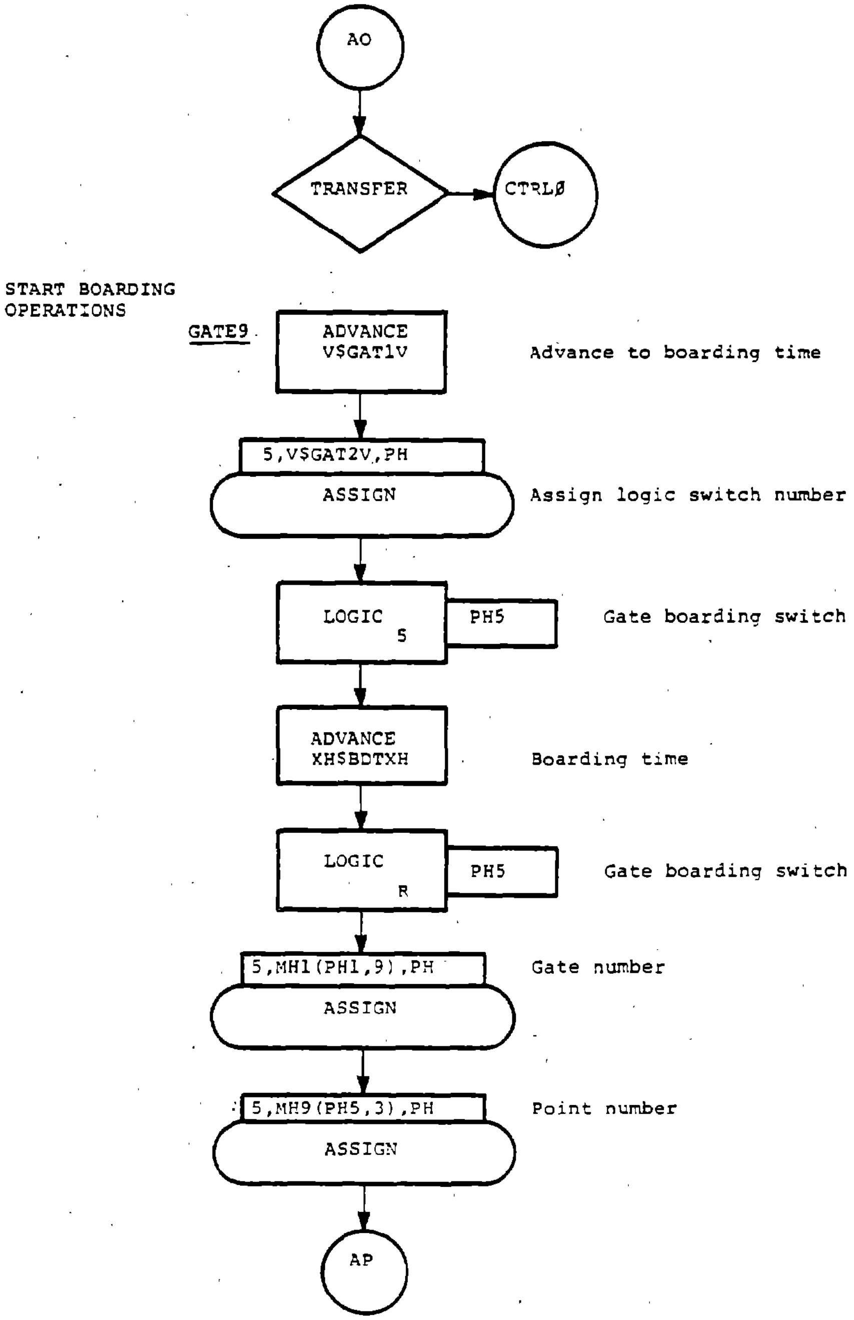
Leave gate service

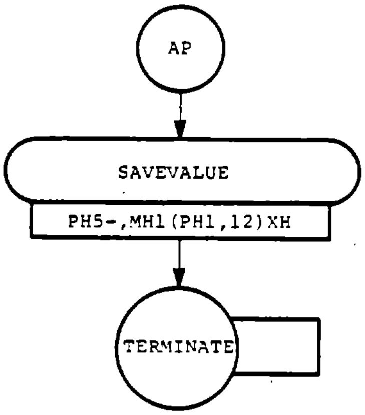
Gate set when
boarding begins

Congestion

Gate service time-Braniff

Pax waiting time at lounge
to board

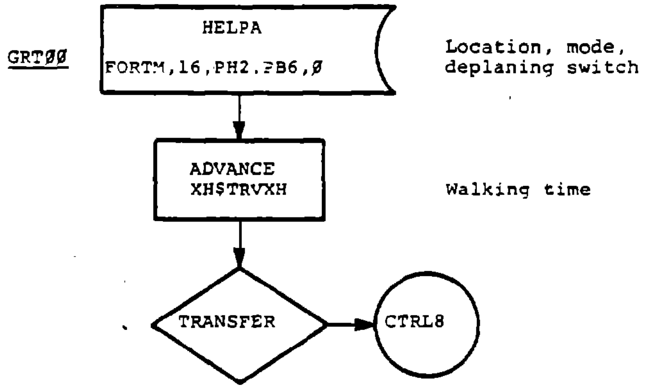

AO
TRANSFER
CTRLØ
START BOARDING
OPERATIONS
GATE9
ADVANCE
V$GAT1V
Advance to boarding time
5,V$GAT2V,PH
ASSIGN
Assign logic switch number
LOGIC
5
PH5
Gate boarding switch
ADVANCE
XH$BDTXH
Boarding time
LOGIC
R
PH5
Gate boarding switch
5,MH1(PH1,9),PH
ASSIGN
Gate number
5,MH9(PH5,3),PH
ASSIGN
Point number
AP



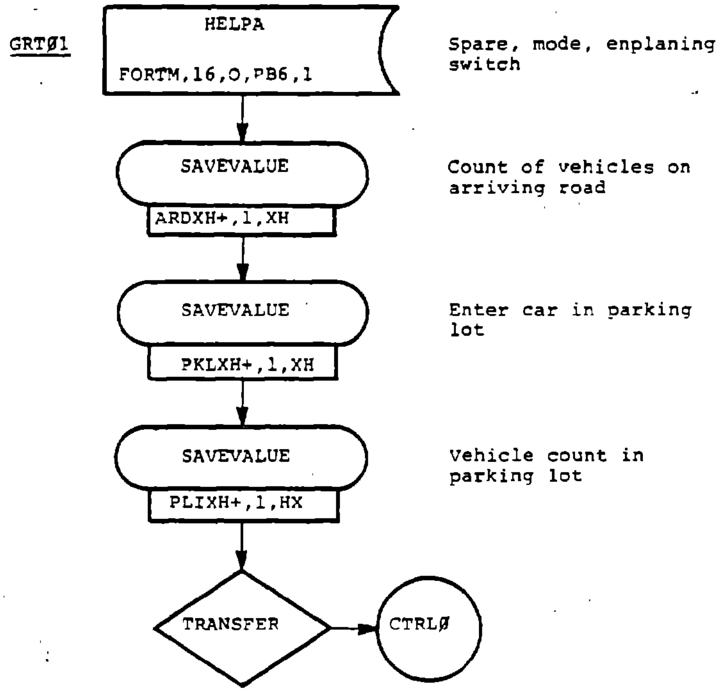
Remove pax boarding
from point

GROUND TRANSPORTATION (MISC)

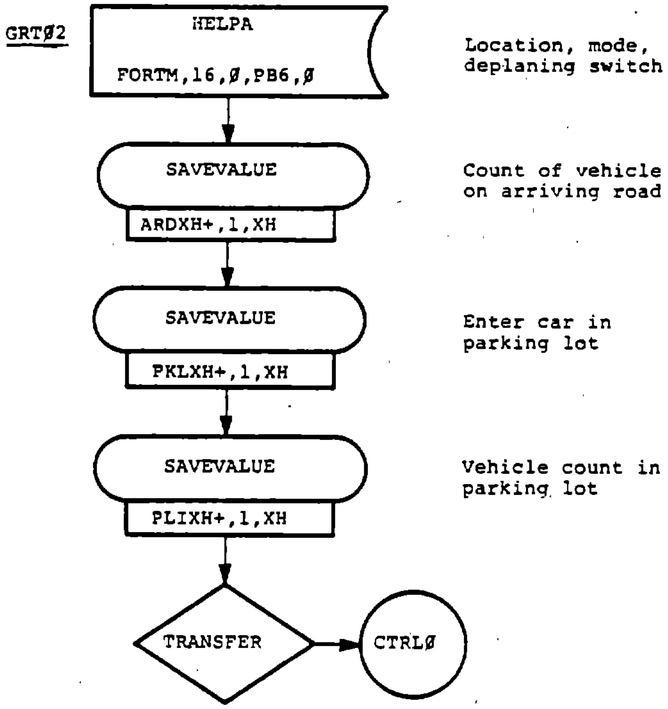
DEPLANING PAX-SELF



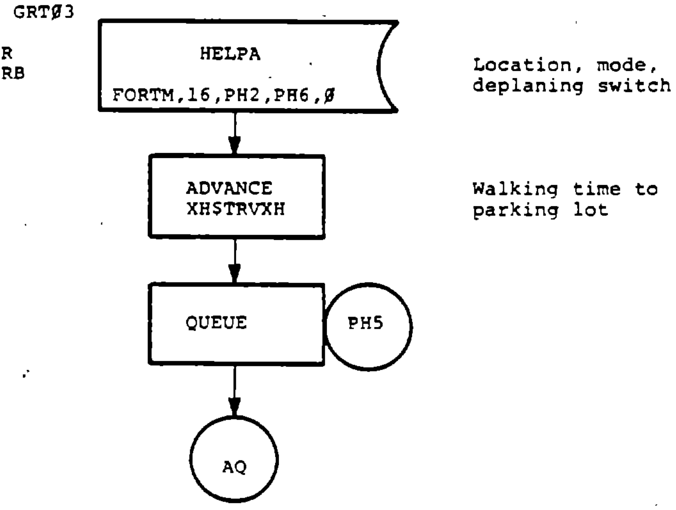
ENPLANING PAX-SELF

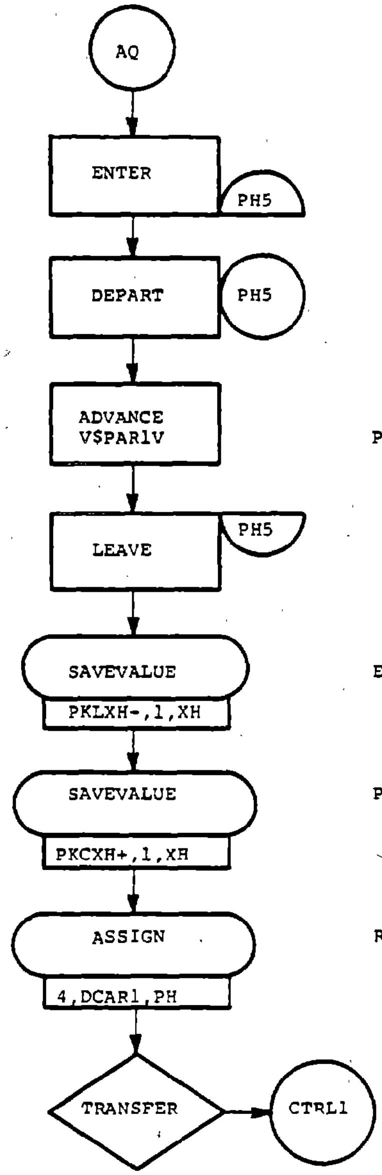


GREETERS



GREETERS TAKING CAR FROM PARKING TO CURB





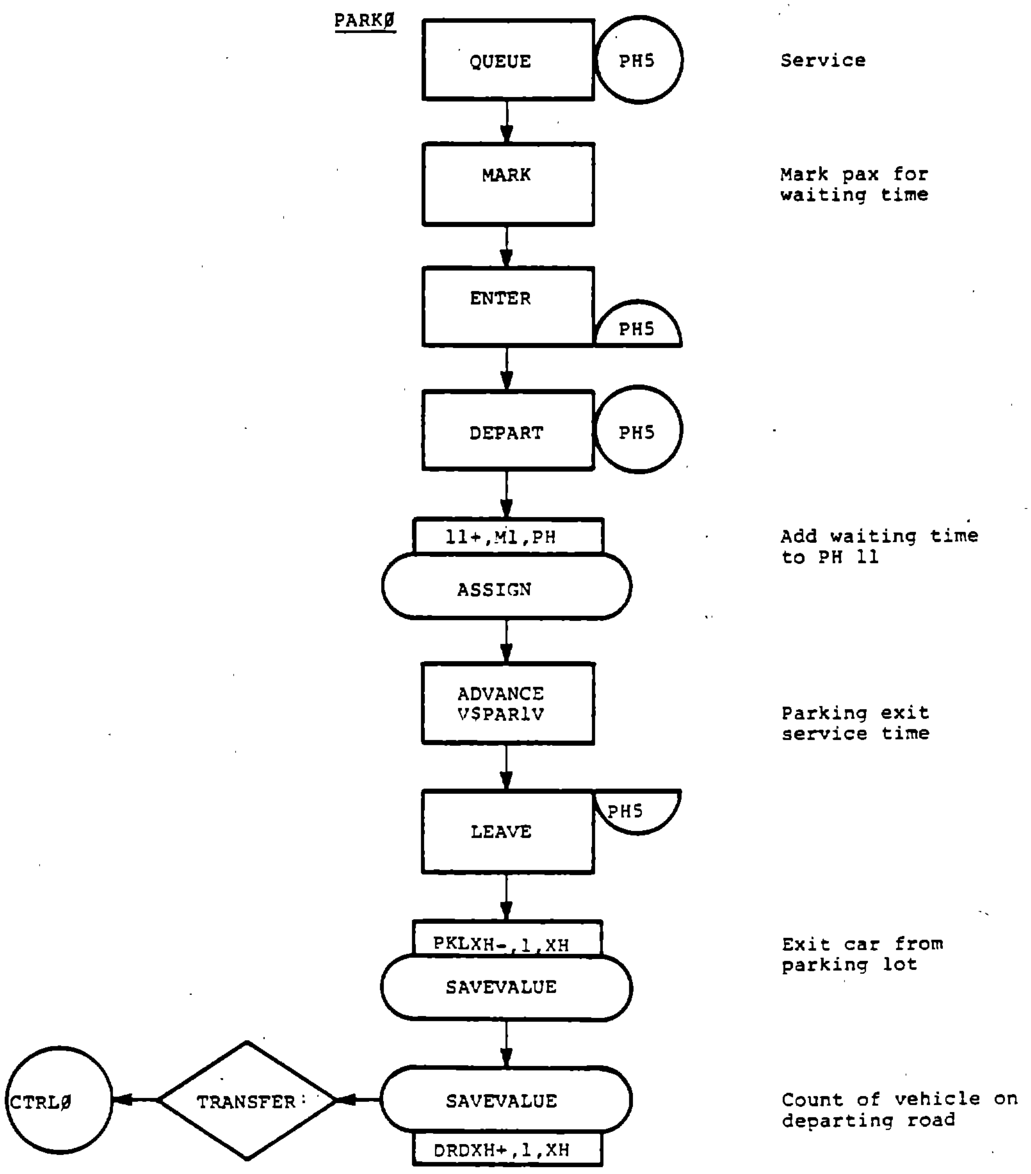
Parking lot service time

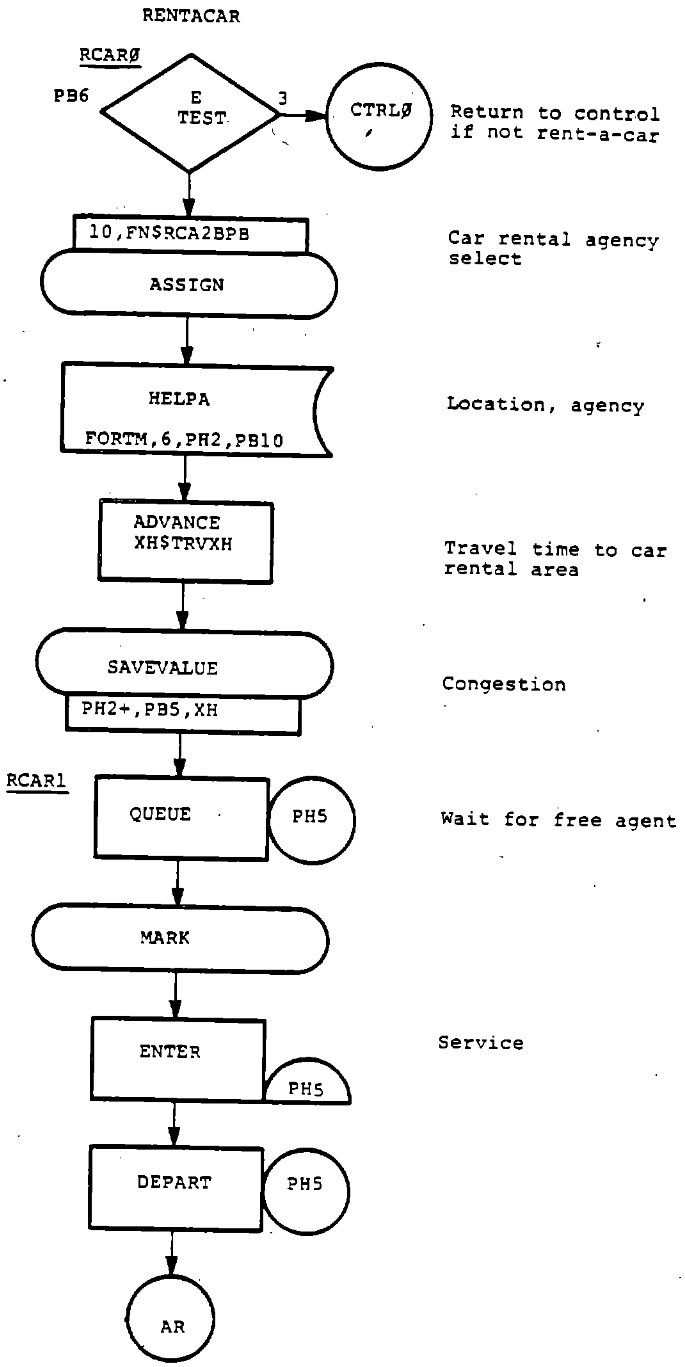
Exit car from parking lot

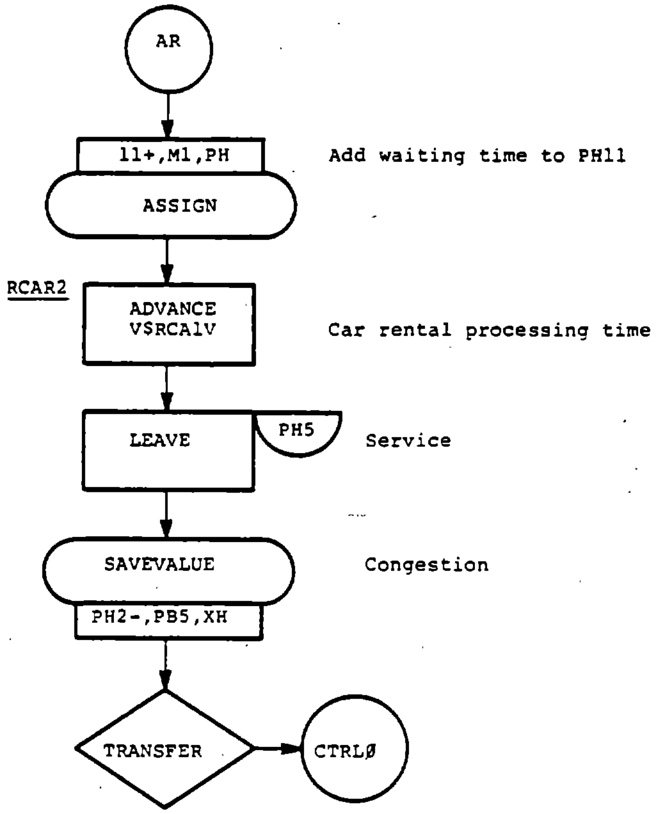
Parking to curb recirculation

Route to deplaning curb

PARKING (DEPL PAX - CARS)

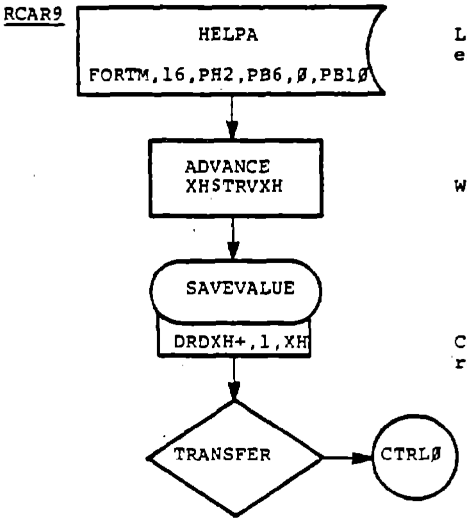

PARKØ
QUEUE
PH5
Service
MARK
Mark pax for waiting time
ENTER
PH5
DEPART
PH5
11+,M1,PH
ASSIGN
Add waiting time to PH 11
ADVANCE VSPAR1V
Parking exit service time
LEAVE
PH5
PKLXH-,1,XH
SAVEVALUE
Exit car from parking lot
SAVEVALUE
DRDXH+,1,XH
Count of vehicle on departing road
TRANSFER
CTRLØ


RENTACAR
RCARØ
PB6
E
TEST
3
CTRLØ
Return to control
if not rent-a-car
10,FN$RCA2BPB
ASSIGN
Car rental agency
select
HELPA
FORTM,6,PH2,PB10
Location, agency
ADVANCE
XH$TRVXH
Travel time to car
rental area
SAVEVALUE
PH2+,PB5,XH
Congestion
RCAR1
QUEUE
PH5
Wait for free agent
MARK
ENTER
PH5
Service
DEPART
PH5
AR


AR
11+,M1,PH
ASSIGN
Add waiting time to PH11
RCAR2
ADVANCE
V$RCA1V
Car rental processing time
LEAVE
PH5
Service
SAVEVALUE
PH2-,PB5,XH
Congestion
TRANSFER
CTRLØ

Deplaning pax in ground transport
who have already rented car.

NOTE: Current logic assumes pax
picks up car at agency parking lot.



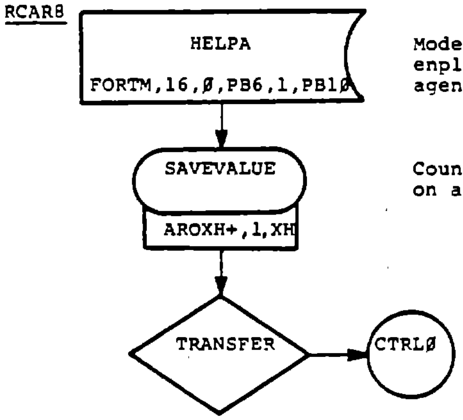
Location, mode, deplaning
enplaning switch agency

Waiting time

Count of cars on departing
road

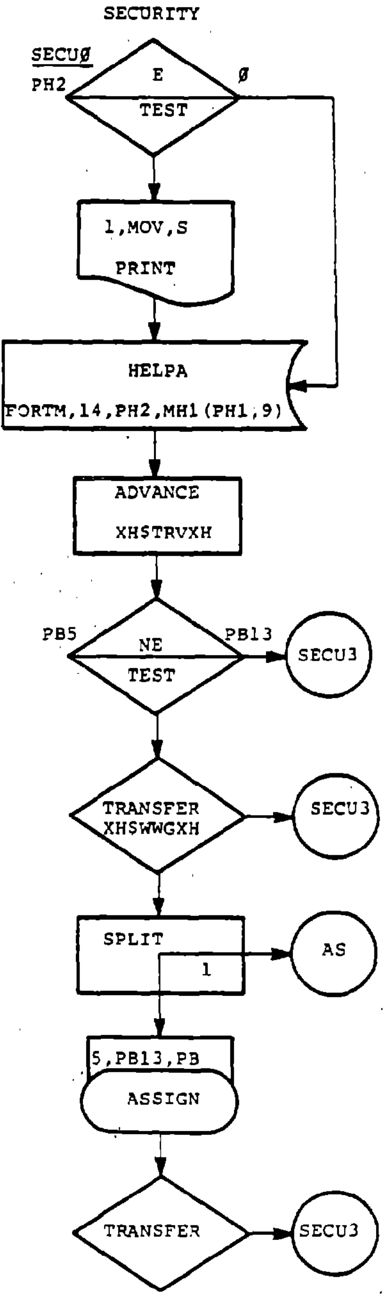
Enplaning Pax Rent-a-car

NOTE: Current logic assumes rental car is returned to a parking lot (general or agency lot). Processing, if any, is done in the terminal.



Mode, deplaning/ enplaning switch, agency

Count of vehicles on arriving road



If current location is point zero, print current events chain

Location, gate

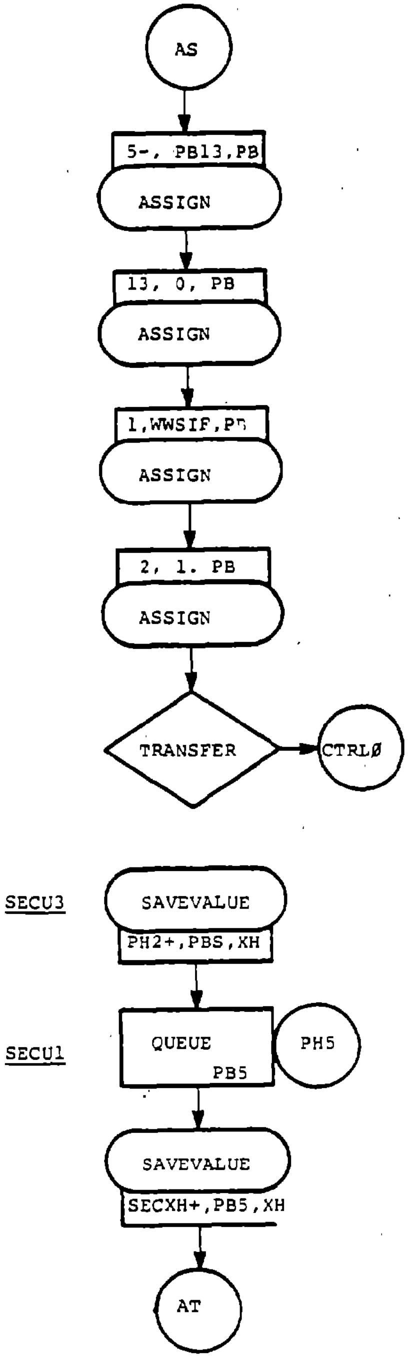
Walking time to security

Fall through if have well-wishers

Fall through if leaving well-wisher at security

Split well-wishers off

Drop well-wishers



Subtract out pax

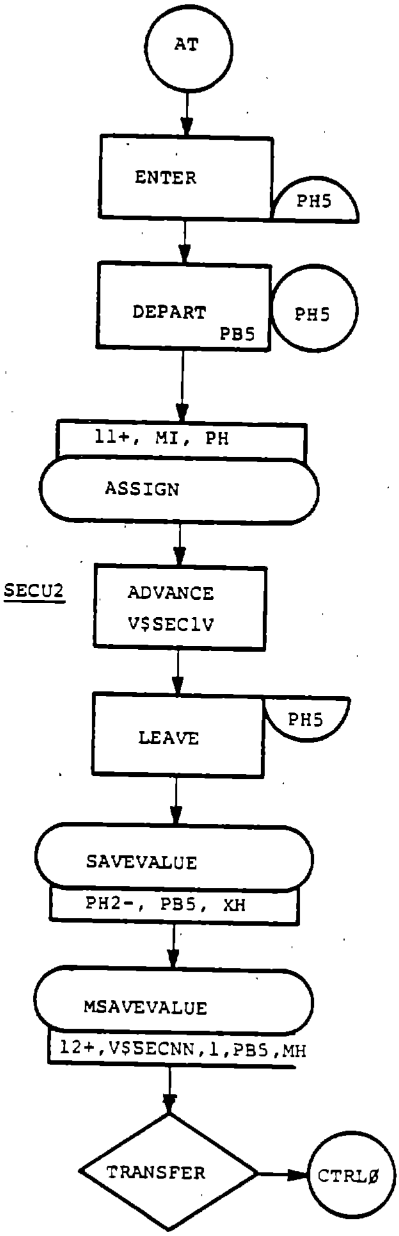
Set pax to Ø

Well-wisher leaving from security process function

Process function pointer

Congestion

Passenger count at security



Service

Add waiting time to PH11

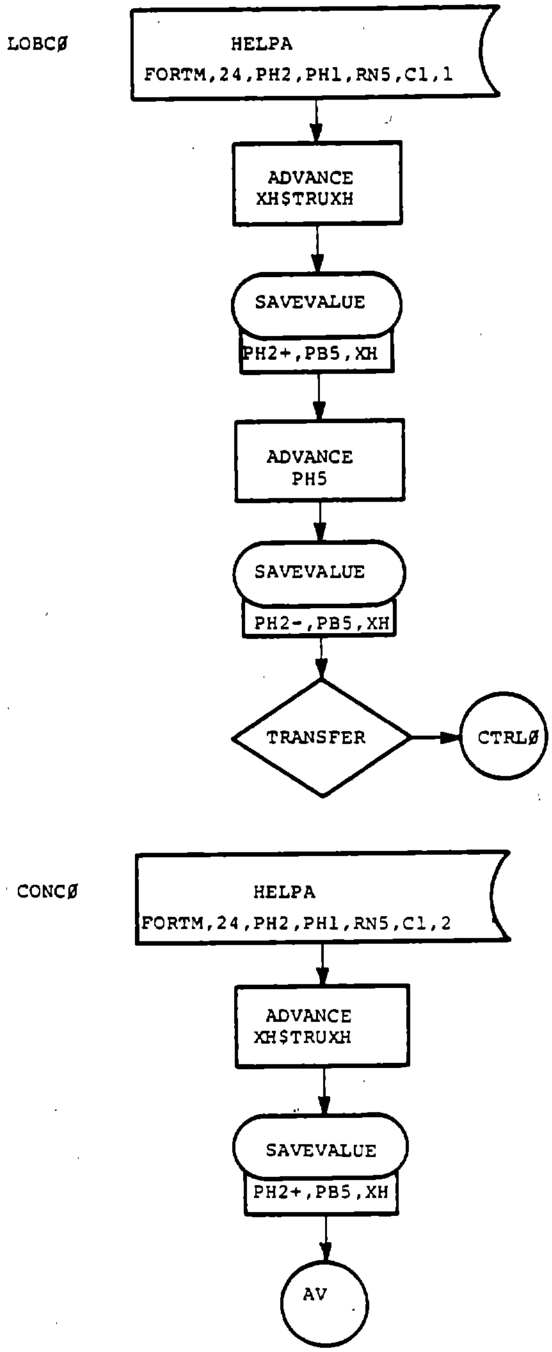
Security check time

Service

Congestion

No. of pax leaving security

CONCESSIONS



Location, flight,
random no., clock,
switch

Walking time to
lobby concession

Congestion

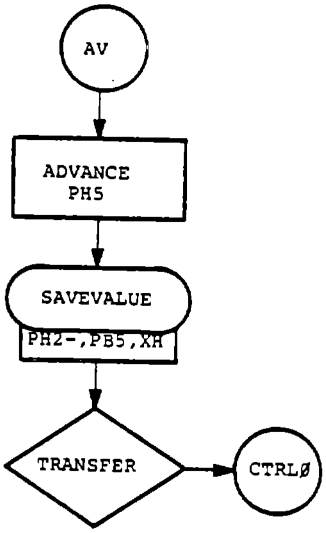
Waste time at
concession

Congestion

Location, flight,
random no., clock,
switch

Walking time to
concourse concession

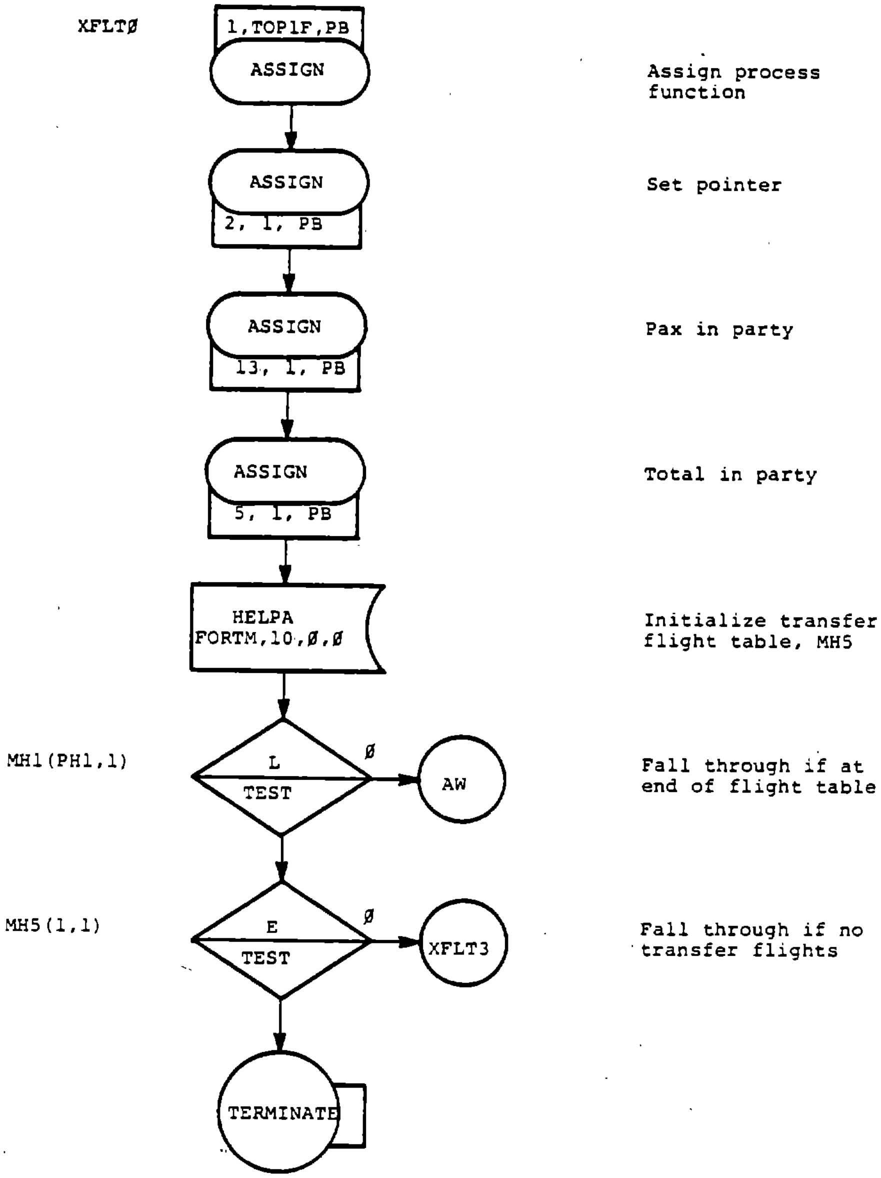
Congestion

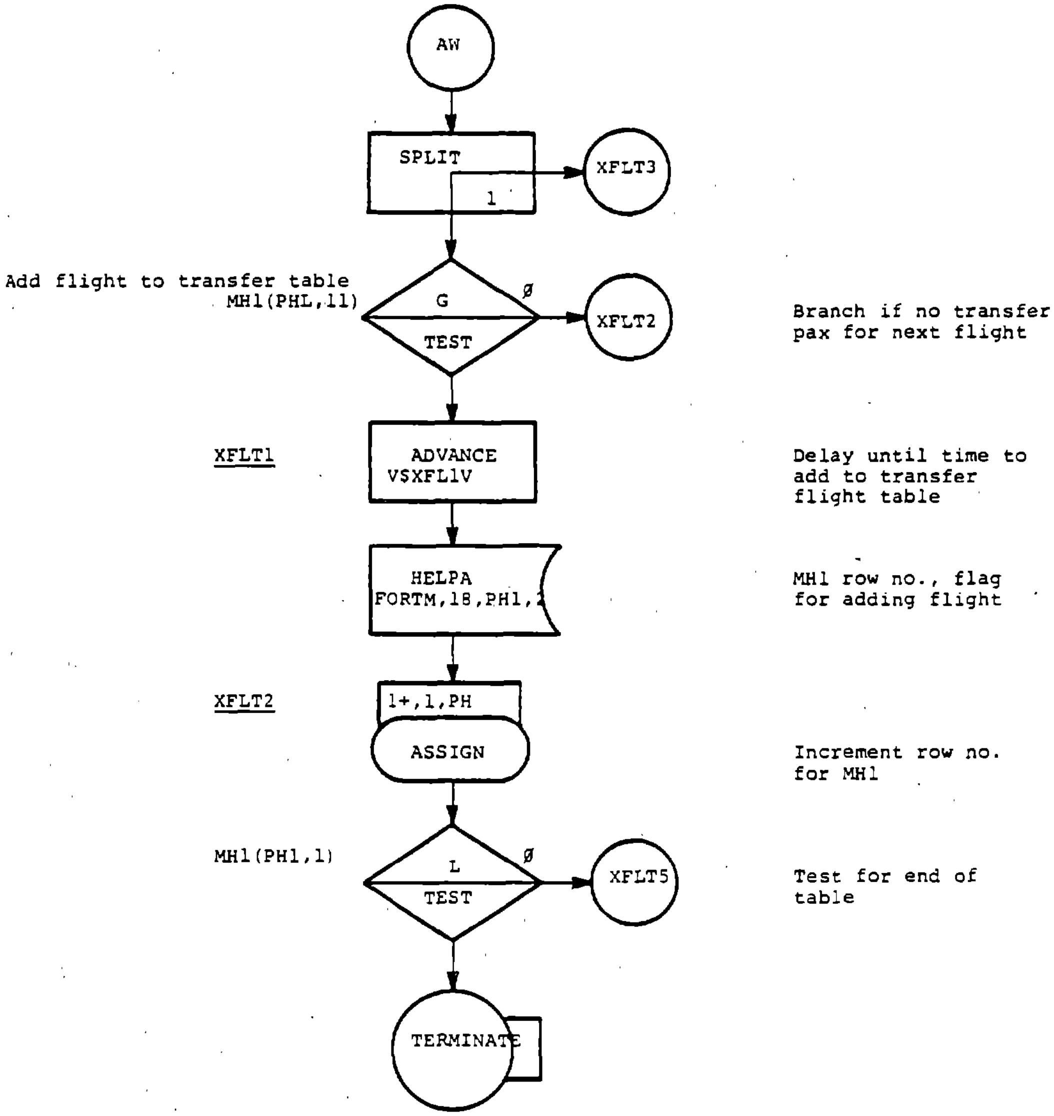


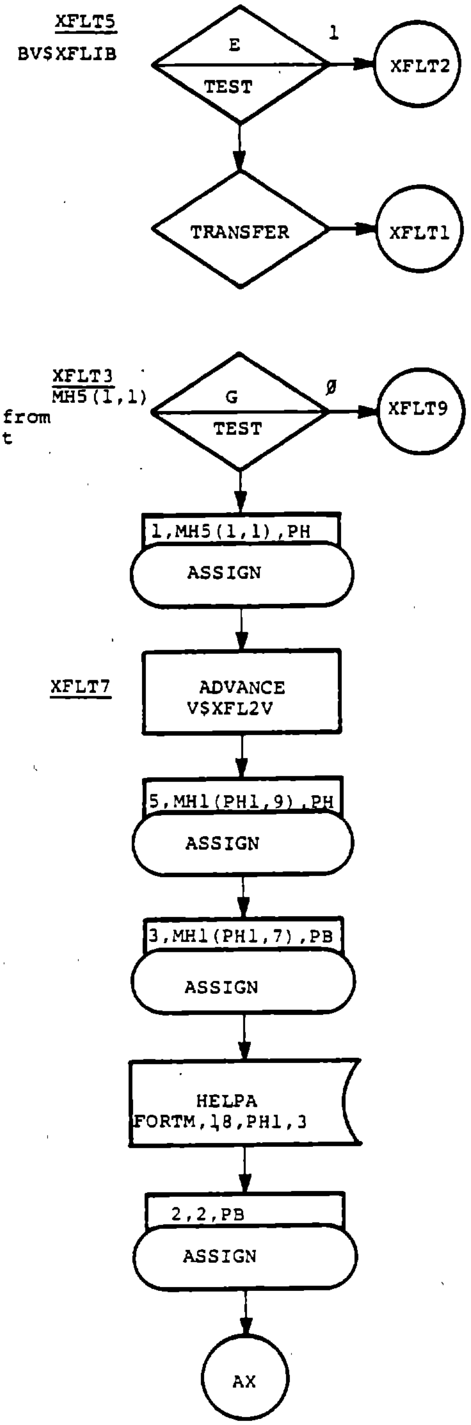
Waste time at concession

Congestion

TRANSFER FLIGHTS




AW
SPLIT
1
XFLT3
Add flight to transfer table
MH1(PHL,11)
G
TEST
Ø
XFLT2
Branch if no transfer pax for next flight
XFLT1
ADVANCE
V$XFL1V
Delay until time to add to transfer flight table
HELPA
FORTM,18,PH1,2
MH1 row no., flag for adding flight
XFLT2
1+,1,PH
ASSIGN
Increment row no. for MH1
MH1(PH1,1)
L
TEST
Ø
XFLT5
Test for end of table
TERMINATE



Branch if arriving flight or no transfer pax

Delete flight from transfer flight table

Branch if transfer flight table empty

First flight to delete

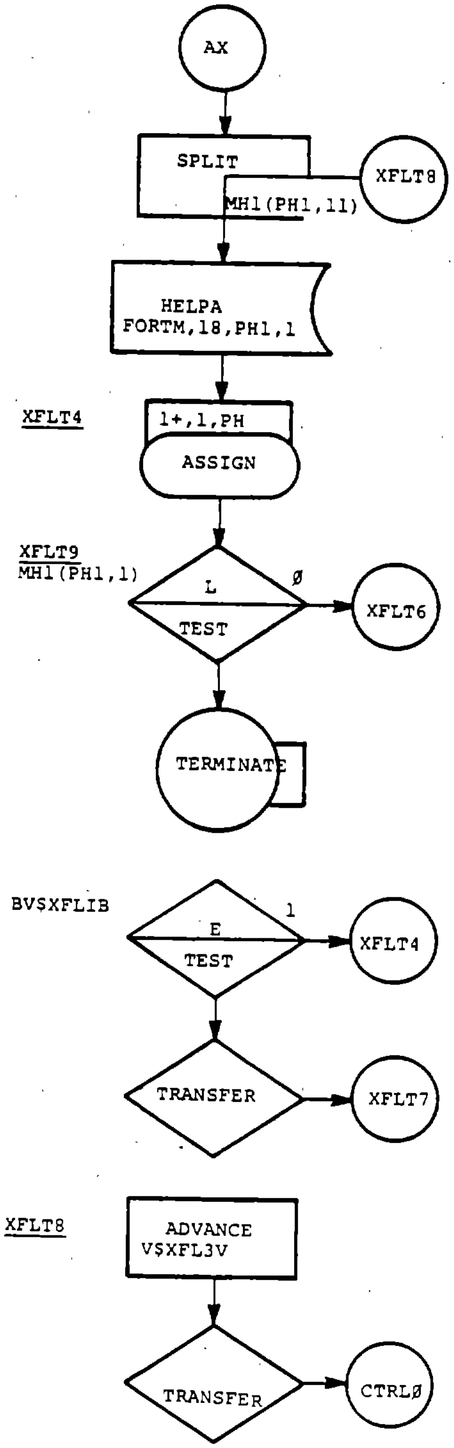
Delay before deleting flight from transfer table

Gate no.

Type of flight (DOM,COM,INT)

MH row no., flag for ticket counter

Skip concourse for TDPIF function



Create remaining transfer pax

Delete flight from table

Increment MH1 row no.

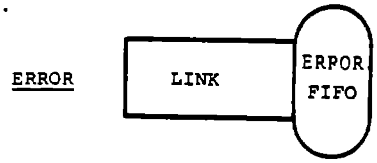
Test for end of table

Branch if arriving flight or no transfer pax

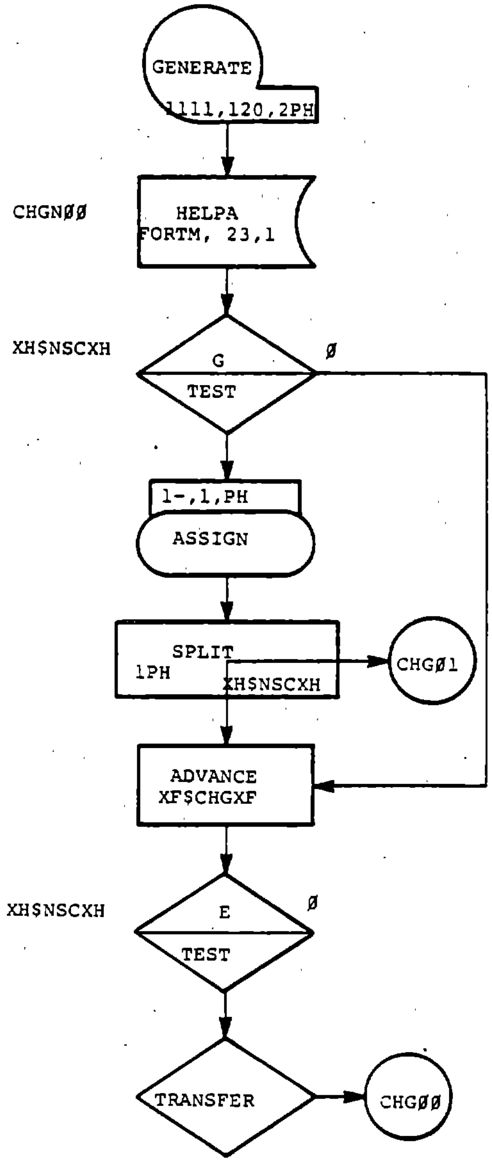
Stagger pax to security

ERROR CHAIN

Collects transactions causing selected errors. PH4 will normally give clue to nature of problem



CHANGE CARD READER



Perform previous
change card and
read next change card

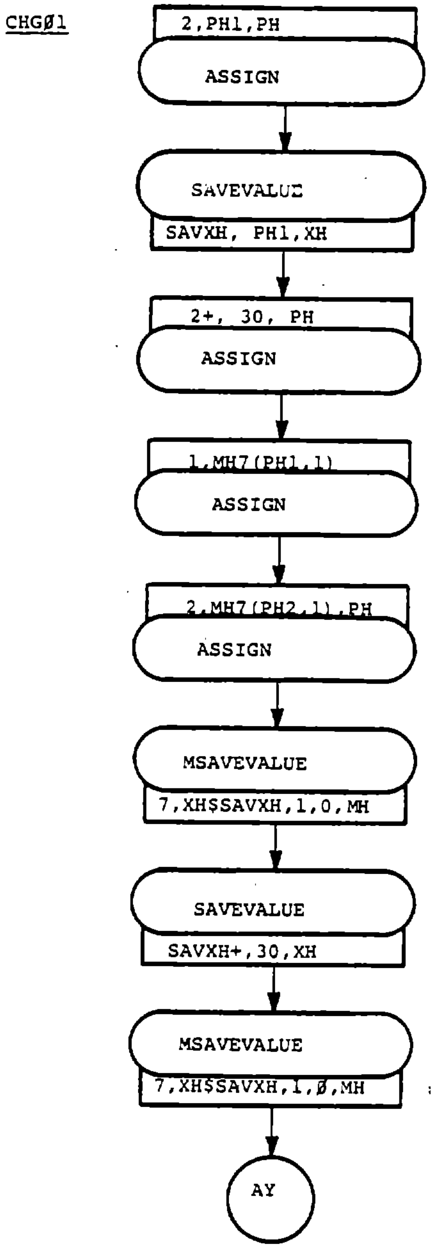
If no storage
changes, do not
split

Parent PH1 set to -1

Split 1 transaction
to change each
storage sequenced
in PH1

Wait until next
change

If wait was Ø, wait
till MH7 used



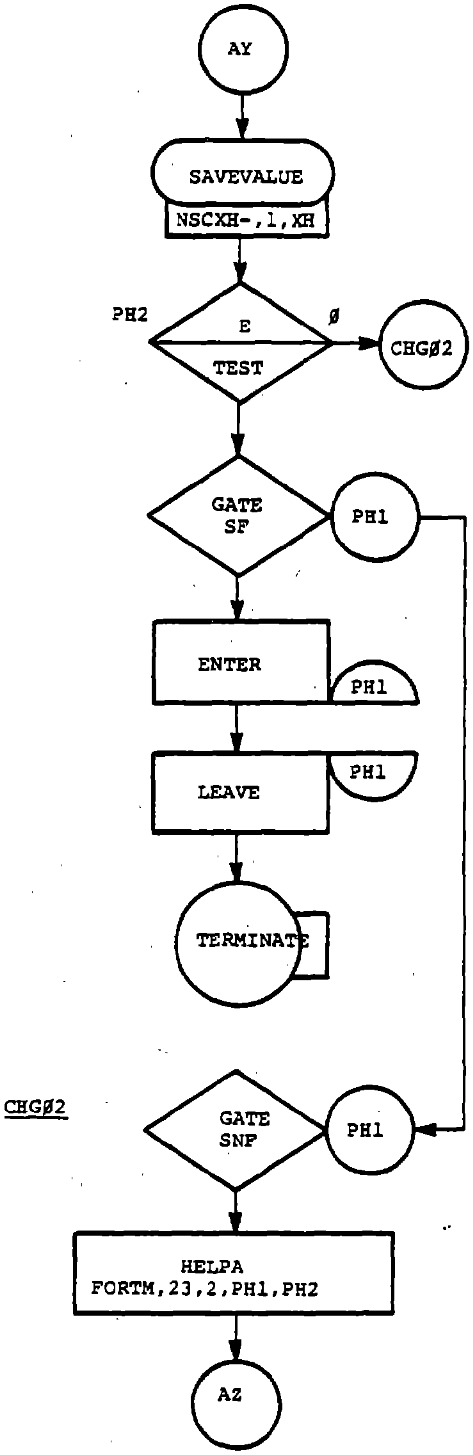
Save MH7 row pointer

PH2 points to second portion of MH7

Change PH1 from MH7 row reference to storage subscript (set in FORTM)

Change PH2 to required storage capacity (set in FORTM)

Zero out MH7



Count down of transactions using MH7

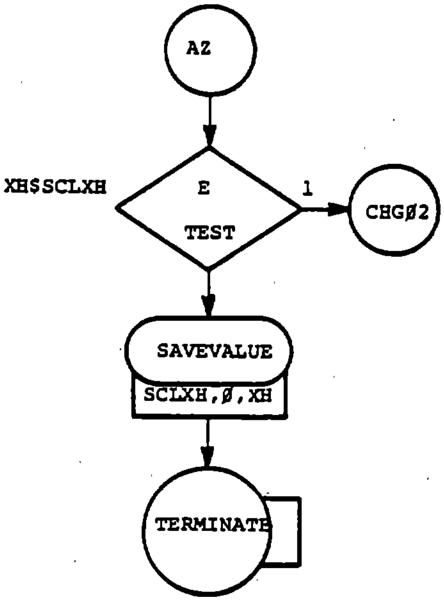
Fall through if no lowering needed

If new capacity higher...

Allow transactions waiting on delay chain to start moving

Wait until someone leaves

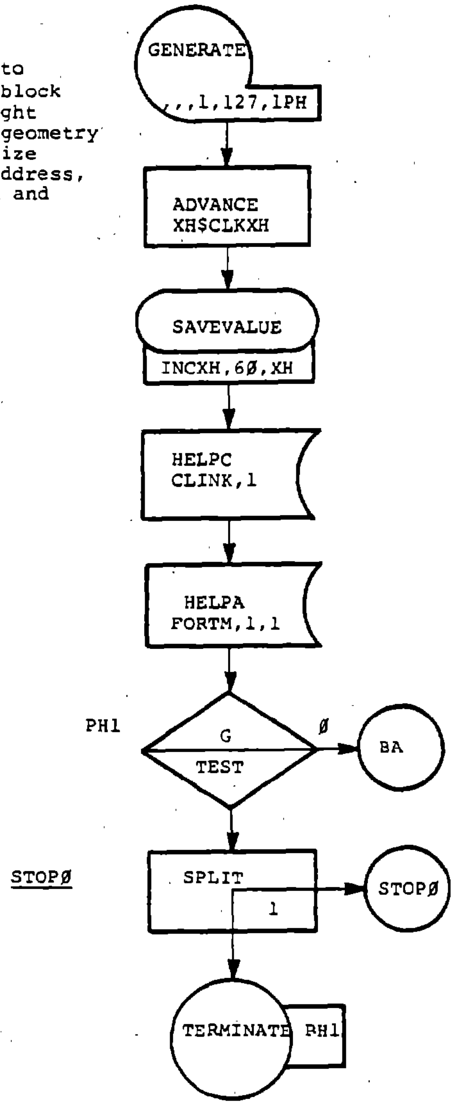
Push storage capacity down to current contents



Storage lowering complete

Reset storage lowering completion flag

TIMER - INITIAL FORTRAN CALL



Initial call to FORTRAN Help block to: read flight schedule and geometry data, initialize matrix base address, execute CLINK and MNLINK.

Needed to define "CLKXH" halfword

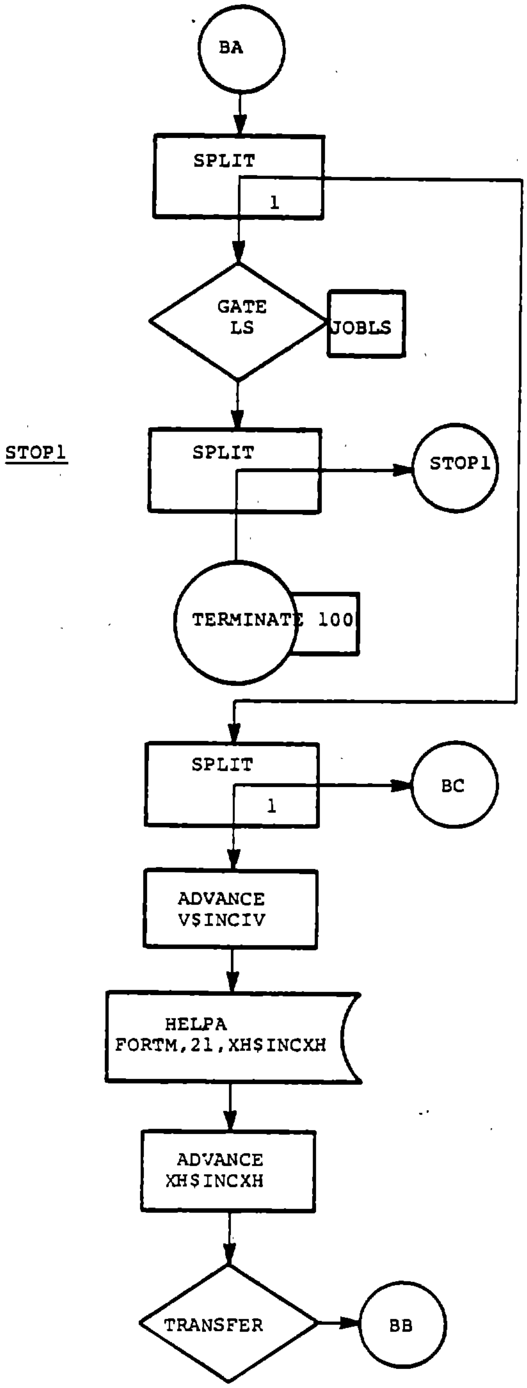
Set "clock" increment

Initiate FORTRAN-GPSS linking

Complete linking and read in input data

Was error detected in FORTM program

Stop simulation

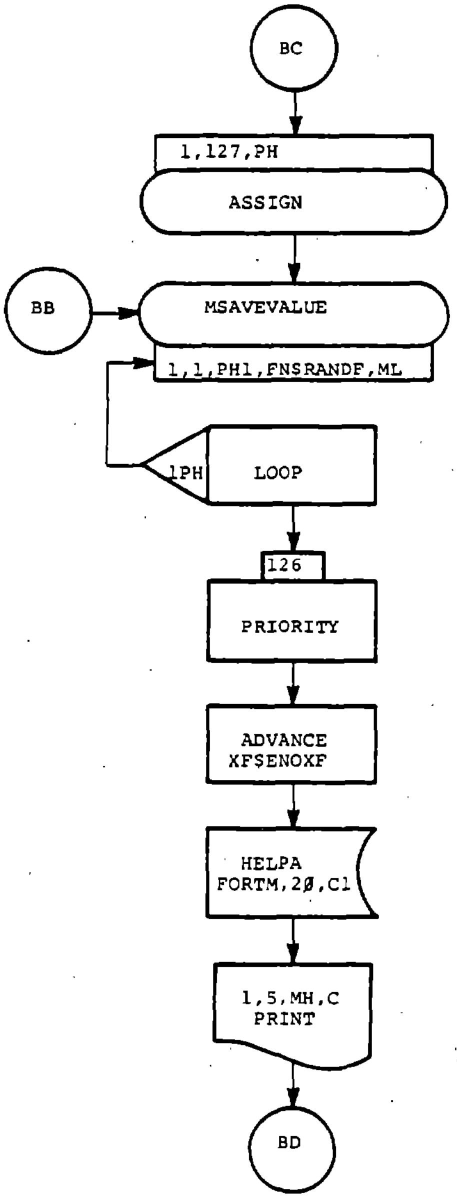


Set if error count
over limit

Clock increment

Clock update

Clock increment



Loop counter

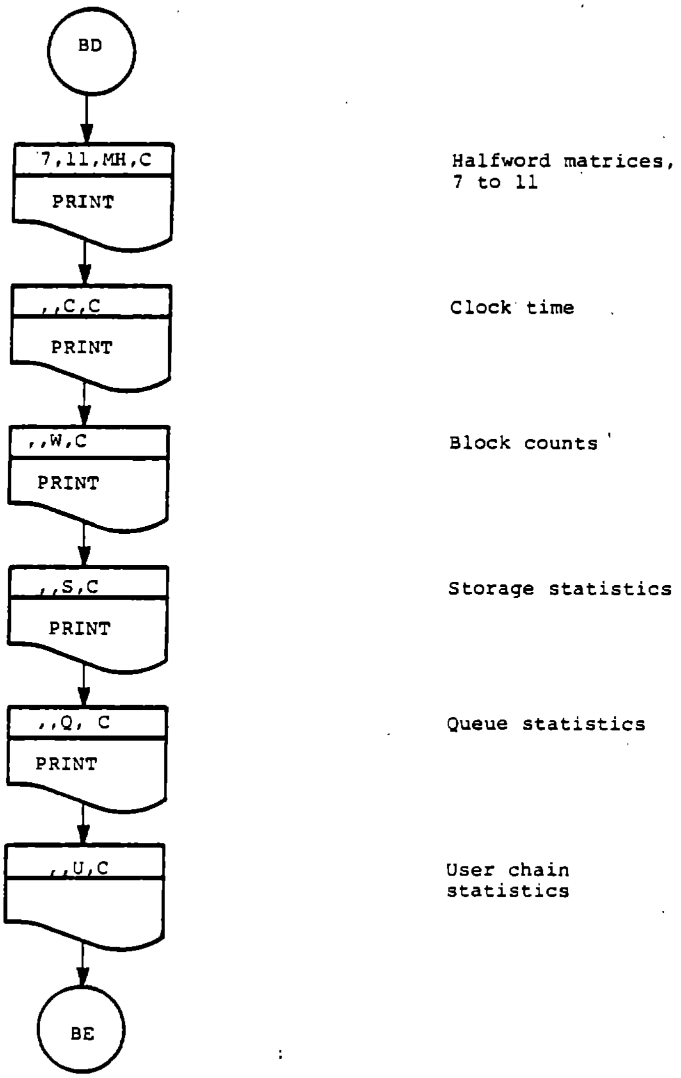
F,11 ML1 with
random numbers

Allows clock to
occur first

Advance to end
of run

Make formatted
reports

Print selected
output: Halfword
matrices 1 to 5


BD
7,11,MH,C
PRINT
Halfword matrices,
7 to 11
,,C,C
PRINT
Clock time
,,W,C
PRINT
Block counts
,,S,C
PRINT
Storage statistics
,,Q, C
PRINT
Queue statistics
,,U,C
User chain
statistics
BE

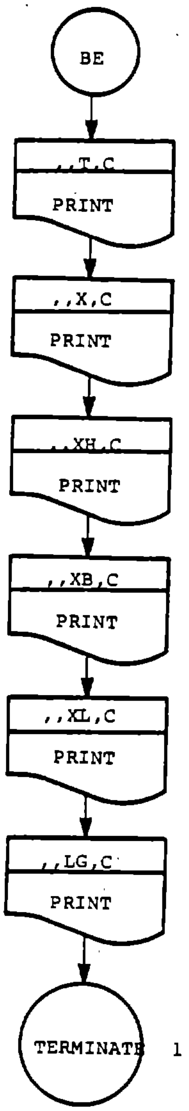


Table statistics

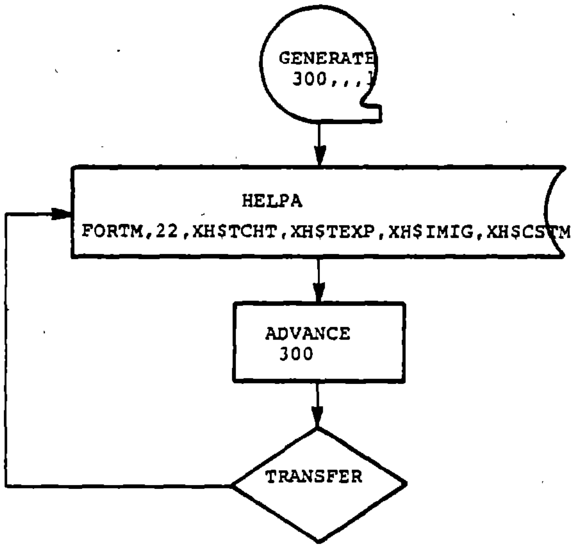
Fullword savevalues

Halfword savevalues

Byte savevalues

Floating point
savevalues

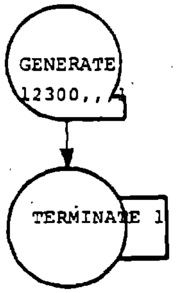
Logic switches



Generate transaction
for snapshot
reports

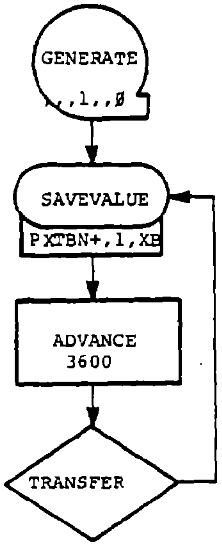
Call snapshot
section in FORTM

Five minute wait
until next snapshot



For reset

Timer for changing hourly table for pax waiting time



Generate timer transaction

Increment number of table

Wait one hour

APPENDIX A-3

PROGRAM LISTING

```
**********************************************************************
*                                                                    *
*                        AUXILIARY PROGRAM                           *
*                                                                    *
**********************************************************************
*        THIS PROGRAM GENERATES TRANSACTIONS REPRESENTING ENPLANING
*    PASSENGER GROUPS.  THESE TRANSACTION ARE STORED ON A 'JOBTAPE'.
      RMULT      ,,,743,31,743,31
      SIMULATE 3
      LOAD       FORTM
       UNLIST     ABS        ,USED TO LIST THE PROGRAM
*
*
CLKXH EQU        1,XH        C L O C K   -   USED IN MAIN MODEL
CMH01 SYN        16          NUMBER OF COLUMNS OF MATRIX MH1
CMH04 SYN        2           NUMBER OF COLUMNS OF MATRIX MH4
CML02 SYN        10          NUMBER OF COLUMNS OF MATRIX ML2
*
*
*   H A L F W O R D   M A T R I C E S
*
1     MATRIX     MH,340,16   FLIGHT SCHEDULE
*   1 ROW PER FLIGHT PLUS 1 ROW TO TERMINATE DEPL PAX GENERATION
*   (PAX XAC PH1 POINTS TO MH1 ROW)
*   COL 1 - ARRIVING OR DEPARTING FLIGHT (0 OR 1; -1 ---> END OF TABLE)
*       2 - FLIGHT NO
*       3 - AIR LINE
*       4 - SCHEDULED ARRIVAL/DEPARTURE TIME
*       5 - DELP. PAX BEING MET BY PRIVATE CAR
*       6 - TIME FROM START (MINUTES)
*       7 - DOMESTIC, COMMUTER OR INTERNATIONAL FLT (1,2,3)
*       8 - AIRCRAFT TYPE
*       9 - GATE NO
*      10 - ORIGINATING/TERMINATING PASSENGERS
*      11 - TRANSFER PASSENGERS
*      12 - BAG CLAIM AREA (ARV FLT)
*         - PAX WAITING AT LOUNGE TO BOARD (DEPT FLT)
*      13 - TRANSIT PASSENGERS
*      14 - TOTAL TERMINATING BAGS (ARV FLT)
*      15 - TOTAL TRANSFER BAGS (ARV FLT)
*      16 - TRANSFERS OUT OF SYSTEM
*
4     MATRIX     MH,3,2      PERCENT ENPLANING PAX TICKETED
*    ROW  1 - DOMESTIC           COL  1  % PRETICKETED
*         2 - COMMUTER                2  % PRETICKETED THAT GO DIRECT
*         3 - INTERNATIONAL              TO SECURITY
*
*   F L O A T I N G   M A T R I C E S
*
2     MATRIX     ML,3,10     GROUND TRANSACTION MODAL CHOICE %
*   ROW  1 - DOMESTIC
*        2 - COMMUTER
*        3 - INTERNATIONAL
*   COL  1 - PRIVATE CAR (PICKUP, DROP OFF)       PERCENTAGE
*        2 - DRIVES SELF (PARKING)                CUM PCT - PRIVATE
*        3 - RENTAL CAR                           CUM PCT - PRIVATE
*        4 - BUS/LIMOUSINE                        CUM PCT - PRIVATE
*        5 - TAXI                  ** 1.0 **      CUM PCT - PRIVATE
*        6 - 10  ** S P A R E **
```

```
*
*   F U N C T I O N S
*
 ARV1F FUNCTION    RN7,C6      ARRV TIME PRIOR TO FLT (0905-0400)
0.,0/.115,3180/.38,5160/.68,6360/.90,7440/1.0,8400
 ARV2F FUNCTION    RN7,C6      ARRV TIME PRIOR TO FLT (0405-0900)
0.,0/.02,720/.16,1680/.425,2760/.93,4080/1.0,4500
 TOS1F FUNCTION    RN7,C2      ARRV TIME (TRANSFERS FROM OUT OF SYSTEM)
0.,0/1.0,1200
 PPPEF FUNCTION    RN7,D6      PAX/PARTY  ENPLANING
0.32,1/0.71,2/0.91,3/0.97,4/0.99,5/1.0,6
 WWPPF FUNCTION    RN6,D4      WELL-WISHERS/PARTY  ENPLANING
0.9,0/0.96,1/0.99,2/1.0,3
*
*   V A R I A B L E S
*
 FLT1V VARIABLE    60*MH1(PH1,6)-8700-300  DELAY TO 150 MIN. BEFORE DEPT
 FLT2V VARIABLE    60*MH1(PH1,6)-4800-300  DELAY TO 85 MIN. BEFORE DEPT
 TOS1V VARIABLE    60*MH1(PH1,6)-2400   DELAY TO 40 MIN. BEFORE DEPT
 TOS2V VARIABLE    MH1(PH1,16)-1        CREATE T.O.O.S. TRANSACTIONS
*
*   P A S S E N G E R   T R A N S A C T I O N   P A R A M E T E R S
*
*        HALFWORD:  1 - FLIGHT TABLE ROW NUMBER
*                   2 - LOCATION (POINT NUMBER)
*                   3 - MAX. BAG RANDOM NUMBER
*                       PAX STORAGE NO IN DEPLANING CURB LOGIC
*                   4 - ADDRESS PARAMETER
*                   5 - ** S C R A T C H **
*                   6 - USER CHAIN FOR BAG CLAIM (DEPL. PAX ONLY)
*                       CAR STORAGE NO IN DEPLANING CURB LOGIC
*                   7 - MH9 ROW FOR LAST FACILITY
*                       (NOT IMPLEMENTED IN EXIT LOGIC)
*                   8 - ** S C R A T C H **
*                       FORTM USES AS RETURN FOR ADDR IN XFER PAX LOGIC
*                   9 - CUMULATIVE WALKING TIME
*                   10 - TRANSACTION SEQUENCE NUMBER FOR PASSENGER
*                        AND GREETER MATCHING
*                   11 - CUMULATIVE PASSENGER WAITING TIME
*        BYTE:      1 - PROCESS FUNCTION NUMBER
*                   2 - PROCESS FUNCTION POINTER
*                   3 - TYPE OF FLIGHT (1=DOM, 2=COM, 3=INT)
*                   4 - NUMBER OF BAGS
*                   5 - NUMBER IN PARTY (VISITORS + SELF)
*                   6 - MODE OF GROUND TRANSPORTATION
*                       (SEE MATRIX ML2 FOR CODES)
*                   7 - 0=DEPLANING, 1=ENPLANING
*                   8 - 1 = TERMINATING, 2 = TRANSFER
*                       3 = TRANSIT, 4 = TRANSFER OUT OF SYSTEM
*                   9 - BAG CLAIM AREA LOG SWITCH (DEPL. PAX ONLY)
*                     - 0 = TICKETED, 1 = NOT (ENPL. PAX ONLY)
*                  10 - USED IN DEPLANING FOR BAG FUNCTION
*                     - RENTACAR AGENCY
*                  11 - CURRENT PROCESS CODE
*                  13 - PAX IN PARTY
*                  14 - PARKING LOT NUMBER
*
*   E N P L A N I N G   P A S S E N G E R   C R E A T I O N
*
      GENERATE     ,,,1,10,11PH,14PB
      ASSIGN       2,1,PB                 INIT PROCESS FN POINTER
```

```
         ASSIGN      7,1,PB                 MARK AS ENPLANING PAX            00113000
ENPLO    ASSIGN      1+,1,PH                FLT TABLE (MH1) ROW SUBSCRIPT    00114000
         TEST L      MH1(PH1,1),0,*+2       MH1(PH1,1)=-1 --> TABLE END (*+2)00115000
         TERMINATE                                                           00116000
         TEST E      MH1(PH1,1),1,ENPLO     SKIP ARRIVING FLIGHTS            00117000
         SPLIT       1,ENPLO                                                 00118000
         ASSIGN      3,MH1(PH1,7),PB        TYPE FLT (1,2,3 = DOM,COM,INT)   00119000
         ASSIGN      5,MH1(PH1,9),PH        GATE NO                          00120000
         SPLIT       1,ENPL3                TO ORIGINATING PAX LOGIC         00121000
         TEST LE     MH1(PH1,16),0,*+2      FT IF NO T.O.O.S.          (*+2) 00122000
         TERMINATE                                                           00123000
         ASSIGN      8,4,PB                 MARK AS TRANSFER-OUT-OF-SYSTEM   00124000
         ASSIGN      13,1,PB                ALL TRANSFERS SINGLE PAX         00125000
         ASSIGN      5,1,PB                 ...WITH NO VISITORS              00126000
         ADVANCE    V$TOS1V                 DELAY TO 40 MIN. BEFOR DEPT      00127000
         SPLIT       V$TOS2V,*+1                                       (*+1) 00128000
         ADVANCE     FN$TOS1F                                                00129000
         TRANSFER   ,WRITE                  WRITE ON JOBTAPE                 00130000
ENPL3    TEST E      MH1(PH1,10),0,*+2      CHECK FOR PAX                    00131000
         MSAVEVALUE  1,PH1,10,1,MH          IF PAX=0 SET=1                   00131100
         TEST L      MH1(PH1,4),905,ENPL1                                    00131200
         ADVANCE     V$FLT2V                0405-0904 DEPARTURES             00132000
         ASSIGN      8,MH1(PH1,10),PH       ORIGINATING PAX                  00133000
         ASSIGN      13,FN$PPPEF,PB         # OF PAX IN PARTY                00134000
         ASSIGN      8-,PB13,PH             SUBTRACT FROM TOTAL              00135000
         TEST LE     PH8,0,*+5              FT IF TOTAL <= 0                 00136000
         ASSIGN      13+,PH8,PB             ADJUST LAST PARTY SIZE           00137000
         TEST LE     PB13,0,*+2             FT IF PB13 LE ZERO               00137100
         ASSIGN      13,1,PB                SET PB13 TO ONE                  00137200
         TRANSFER    ,*+2                   OUT OF LOOP                 (*+2) 00138000
         SPLIT       1,*-7                  GO BACK TO CREATE ANOTHER (*-7) 00139000
         ADVANCE     FN$ARV2F               ARRV TIME PRIOR TO FLIGHT        00140000
         TRANSFER    ,ENPL2                                                  00141000
ENPL1    ADVANCE     V$FLT1V                0905-0400 DEPARTURES             00142000
         ASSIGN      8,MH1(PH1,10),PH       ORIGINATING PAX                  00143000
         ASSIGN      13,FN$PPPEF,PB         # OF PAX IN PARTY                00144000
         ASSIGN      8-,PB13,PH             SUBTRACT FROM TOTAL .            00145000
         TEST LE     PH8,0,*+5              FT IF TOTAL <= 0                 00146000
         ASSIGN      13+,PH8,PB             ADJUST LAST PARTY SIZE           00147000
         TEST LE     PB13,0,*+2             FT IF PB13 LE ZERO               00147100
         ASSIGN      13,1,PB                SET PB13 TO ONE                  00147200
         TRANSFER    ,*+2                   OUT OF LOOP                 (*+2) 00148000
         SPLIT       1,*-7                  GO BACK TO CREATE ANOTHER (*-7) 00149000
         ADVANCE     FN$ARV1F                                                00150000
ENPL2    HELPA       FORTM,5,RN4,RN5,PB3                                     00151000
         ASSIGN      5,PB13,PB              NO WELL-WISHERS                  00152000
         TEST LE     PB6,2,*+2              FT IF CURB OR PARK          (*+2) 00153000
         ASSIGN      5+,FN$WWPPF,PB         ADD IN WELL-WISHERS              00154000
WRITE    WRITE       JOBTA1                                                  00155000
         TERMINATE                                                           00156000
*                                                                            00157000
*   CREATE   DUMMY   XAC   FOR   JOBTAPE                                     00158000
*                                                                            00159000
         GENERATE    ,,,1,127,1PH                                            00160000
         HELPC       CLINK,1                SET UP 2 WAY COMMUNICATION       00161000
         HELPA       FORTM,1,1              READ INPUT DATA                  00162000
         TEST G      PH1,0,*+2              CHECK FOR ERRORS                 00163000
         TERMINATE   PH1                                                     00164000
         SPLIT       1,WRITE                                                 00165000
         ADVANCE     100000                                                  00166000
         SPLIT       1,WRITE                PREVENT JOBTAPE END MESSAGE      00167000
```

```
       ADVANCE    1                    WAIT FOR ABOVE XAC TO BE WRITTEN 00168000
       TERMINATE  1                                                     00169000
*                                                                       00170000
 1     FUNCTION   PH1,L4     MNEMONIC LINK FUNCTION - SEE FORTRAN CALL  00171000
,CMH01/,CMH04/,CML02/,CLKXH                                             00172000
       NOXREF                                                           00172900
       START      1,,,1                                                 00173000
       NOXREF                          USED TO CROSS REF. ITEMS         00173040
       REPORT                                                           00173050
 HMS   TITLE      1,FLIGHT SCHEDULE                                     00173100
 HMS   TITLE      4,%ENPLANING PAX TICKETED                             00173200
 LMS   TITLE      2,GROUND TRANSACTION MODAL CHOICE %                   00173300
*      OUTPUT                          REMOVE * TO PROVIDE ALL STATICS  00173400
       END                                                              00174000
0.468,1/0.833,2/0.917,3/0.967,4/0.987,5/1.0,8                           00278000
       ASSIGN     1,DDP1F,PB                                            00656100
       ASSIGN     1,DDP1F,PB                                            00687100
       REALLOCATE HMS,13,BLO,980,FAC,0,QUE,212,TAB,14,FUN,63,VAR,67       00001000
       REALLOCATE FSV,25,CHA,80,FMS,0,LMS,2,STO,212,LOG,185,BMS,0         00002000
       REALLOCATE LSV,10,BSV,10,HSV,170,BVR,11,XAC,3600,COM,220000        00003000
**********************************************************************    00003010
*                                                                    *    00003020
*                           MAIN PROGRAM                             *    00003030
*                                                                    *    00003040
**********************************************************************    00003050
*          THIS PROGRAM ENACTS THE MOVEMENT OF PASSENGERS AND VISTORS     00003060
*      THOUGH THE TERMINAL AREA.  THIS PROGRAM GENERATES DEPLANING PAS-   00003070
*      SENGER, MERGING THEM WITH THE ENPLANING PASSENGER FROM THE AUX-    00003080
*      ILIIARY PROGRAM.  THE FLOW OF PASSENGER AND VISTORS IS MODEL       00003090
*      THROUGH A SEQUENCE OF SIMULATED LANDSIDE PROCESSING FACILITES.     00003095
        RMULT      ,,,743,743,31,31                                       00003100
       UNLIST      ABS                 USED TO LIST PROGRAM               00004000
       LOAD        FORTM,BAGS,DAG06,XACNO                                 00005000
       JOBTAPE    JOBTA1,ENPLO                                            00007000
*                                                                         00008000
 CHA1D SYN         1                     BAG CLAIM SWITCH-CHAIN           00009000
 CHA1B EQU         1(54),L,C             BAG CLAIM SWITCH-CHAIN           00010000
 CHA2D SYN         55                    BAG CLAIM SWITCH-CHAIN           00011000
 CHA2B EQU         55,L,C                BAG CLAIM SWITCH-CHAIN           00012000
 CHKQS SYN         1                     EXPRESS CHECKIN QUE-STO          00013000
 CHKQS EQU         1(18),Q,S              EXPRESS CHECKIN QUE-STO         00014000
 CUSQS SYN         204                   CUSTOMS QUE-STO                  00015000
 CUSQS EQU         204,Q,S               CUSTOMS QUE-STO                  00016000
 CMH01 SYN         16                    NO. OF COL. - MH1                00017000
 CMH02 SYN         3                     NO. OF COL. - MH2                00018000
 CMH03 SYN         4                     NO. OF COL. - MH3                00019000
 CMH04 SYN         2                     NO. OF COL. - MH4                00020000
 CMH06 SYN         110                    NO. OF COL. - MH6               00021000
 CMH07 SYN         1                     NO. OF COL. - MH7                00022000
 CMH08 SYN         2                     NO. OF COL. - MH8                00023000
 CMH09 SYN         6                     NO. OF COL. - MH9                00024000
 CML02 SYN         10                    NO. OF COL. - ML2                00025000
*  CURB, DP, AND QUEUE MUST ALL HAVE SAME NUMBER OF ENTITIES, AND         00026000
*  ENPL CURB ENTITIES MUST IMMEDIATELY FOLLOW DEPL CURB ENTITIES          00027000
*                                                                         00027050
*  INPUT DATA  CARDS MUST SHOW A FACILITY FOR ALL ENPLANING AND           00027100
*  DEPLANING CURB STORAGES ALLOCATED. I. E., THE NUMBER OF FACNO'S OF     00027200
*  ENPLANING AND DEPLANING CURBS MUST MATCH THE NUMBERS IN PARENTHESES    00027300
*  OF THESE CURB STORAGES.                                                00027400
 DPCBS SYN         44                                                     00028000
 DPCBS EQU         44(4),S               DEPLANING CURB STO               00029000
 EPCBS SYN         48                                                     00030000
 EPCBS EQU         48(4),S               ENPLANING CURB STO               00031000
 DPDPS SYN         52                                                     00032000
 DPDPS EQU         52(4),S               DEPLANING DOUPLE PARK STO        00033000
 EPDPS SYN         56                                                     00034000
 EPDPS EQU         56(4),S               ENPLANING DOUBLE PARK STO        00035000
 DPQCS SYN         60                                                     00036000
 DPQCS EQU         60(4),C,S             DEPLANING QUEUE CHAIN-STO        00037000
 EPQCS SYN         64                                                     00038000
 EPQCS EQU         64(4),C,S             ENPLANING QUEUE CHAIN-STO        00039000
 GAQSL SYN         80                     GATE QUE-STO-SWITCH             00040000
 GAQSL EQU         80(105),Q,S,L         GATE QUE-STO-SWITCH              00041000
 DEPXH EQU         111,XH                DEP. CURB COUNT                  00041100
 RCDXH EQU         112,XH                RECIRC. TO DEPL.                 00041200
 RCEXH EQU         113,XH                RECIRC. TO ENPL.                 00041300
 RERXH EQU         114,XH                RECIRC. ROADWAY COUNT            00041400
 PLIXH EQU         115,XH                PARKING FACILITY ENTRANCE        00041500
 ARDXH EQU         116,XH                 AIRPORT ENTRANCE FLOW             00041510
 DRDXH EQU         117,XH                 AIRPORT EXIT FLOW                 00041520
 EALXH EQU         118,XH                 EASTERN EXPRESS FLOW              00041530
 AMAXH EQU         119,XH                 AMERICAN EXPRESS FLOW             00041540
 UNTXH EQU         120,XH                 UNITED EXPRESS FLOW               00041550
 IMMQS SYN         42                      IMMIGRATION QUE-STO              00042000
 IMMQS EQU         42,Q,S                  IMMIGRATION QUE-STO              00043000
 PARQS SYN         19                      PARKING LOT EXIT QUE-STO         00044000
 PARQS EQU         19(8),Q,S               PARKING LOT EXIT QUE-STO         00045000
 PTNXH EQU         1(110),XH               POINT NO XH'S (CONGESTION)       00046000
 RCRQS SYN         27                      RENTACAR QUE-STO                 00047000
 RCRQS EQU         27(8),Q,S              RENTACAR QUE-STO                  00048000
 SECQS SYN         36                     SECURITY QUE-STO                  00049000
 SECQS EQU         36(5),Q,S              SECURITY QUE-STO                  00050000
 TICQS SYN         185                     TICKETS&CHECKIN QUE-STO          00051000
 TICQS EQU         185(18),Q,S             TICKETS&CHECKIN QUE-STO          00052000
 PAXWT EQU         13,T                   TABLE FOR PAX WALKING TIME        00052100
*                                                                           00052200
*                                                                           00053000
*          D E F I N I T I O N   S T A T E M E N T S                        00053100
*     USED TO DEFINE MATRIX SIZES, FUNCTIONS, VARIABLES AND TABLES          00053200
*                                                                           00054000
*                                                                           00055000
*    H A L F W O R D   M A T R I C E S                                      00056000
*                                                                           00057000
1       MATRIX      MH,330,16    FLIGHT SCHEDULE                            00058000
*    1 ROW PER FLIGHT PLUS 1 ROW TO TERMINATE DEPL PAX GENERATION           00059000
*    (PAX XAC PH1 POINTS TO MH1 ROW)                                        00060000
*    COL 1 -   -1 ---> END OF TABLE                                         00061000
*               0 ---> ARRIVING FLIGHT                                      00062000
*               1 ---> DEPARTING FLIGHT                                     00063000
*               2 ---> INDICATES ALL BAGS FROM ARRIVING FLT AT BAG CLAIM    00064000
*          2 - FLIGHT NO                                                    00065000
*          3 - AIR LINE                                                     00066000
*          4 - SCHEDULED ARRIVAL/DEPARTURE TIME                             00067000
*          5 - DEPL. PAX BEING MET BY PRIVATE CAR                           00068000
*          6 - TIME FROM START (MINUTES)                                    00069000
*          7 - DOMESTIC, COMMUTER OR INTERNATIONAL FLT (1,2,3)              00070000
*          8 - AIRCRAFT TYPE                                                00071000
*          9 - GATE NO                                                      00072000
*         10 - ORIGINATING/TERMINATING PASSENGERS                           00073000
*         11 - TRANSFER PASSENGERS                                          00074000
*         12 - BAG CLAIM AREA (ARRV FLT)                                    00075000
*            - PAX WAITING AT LOUNGE TO BOARD (DEPT FLT)                    00076000
*         13 - TRANSIT PASSENGERS                                           00077000
*         14 - TOTAL TERMINATING BAGS (ARRV FLT)                            00078000
*         15 - TOTAL TRANSFER BAGS (ARRV FLT)                               00079000
*         16 - TRANSFERS OUT OF SYSTEM                                      00080000
*                                                                           00081000
2       MATRIX      MH,15,3                  AIRLINE INFORMATION TABLE      00082000
*    S I Z E   T O   R E Q U I R E M E N T S                                00083000
*    1 ROW PER CARRIER                                                      00084000
*    COL  1 - ENPLANING CURB FAC NO                                         00085000
*         2 - 0/00 PRETICKETED PAX USING EXPRESS CHECKIN (0 = NO EXP CHK)00086000
*         3 - ENPLCURB FAC NO FOR BUS STOP (IF DIFF FROM COL 1)             00087000
*                                                                           00088000
3       MATRIX      MH,110,4     TABLE OF POINTS   *** S I Z E  ***         00089000
*    1 ROW PER POINT                                                        00090000
*    COL 1 - X-COORDINATE                                                   00091000
*        2 - Y-COORDINATE                                                   00092000
*        3 - NEAREST EXIT POINT NO                                          00093000
*       4 - NEAREST ENTRANCE POINT NO
*
 4     MATRIX     MH,3,2      PERCENT ENPLANING PAX TICKETED/DIRECT
*   COL  1 - % PRETICKETED
*        2 - % OF PRETICKETED PAX WHO GO DIRECT TO GATES VIA SECURITY
*   ROW  1 - DOMESTIC
*        2 - COMMUTER
*        3 - INTERNATIONAL
*
 5     MATRIX     MH,101,1    TRANSFER FLIGHT TABLE
*   CONTAINS MH1 ROW OF DEPARTING FLIGHTS TAKING XFER PAX
*
 6     MATRIX     MH,110,110  WALKING TIME EETWEEN POINTS
*
 7     MATRIX     MH,64,1     USED AS WORK AREA BY BAG CLAIM ROUTINES.
*   1 ROW FOR EACH POSSIBLE RANDOM NUMBER 1-64 GENERATED BY "BAGS"
*
 8     MATRIX     MH,20,2     USED TO ACCESS FACILITY DATA IN MH9.
*   CONTAINS SAME INFORMATION AS FORTRAN "NFACSM" ARRAY.
*   COL  1 - COUNT OF THIS FACILITY TYPE
*        2 - INDEX NO OF FACILITY TYPE; ADD FAC NO IN TYPE FOR MH9 ROW
*   1 ROW PER FACILITY TYPE:
*        1 - GATE                        2 - CHECKIN
*        3 - SECURITY                    4 - BAGCLAIM
*        5 - CUSTOMS                     6 - ENTRANCE
*        7 - EXIT                        8 - ENPLCURB (ENPLANING CURB)
*        9 - TRANSFER (STAIRS, ETC.)    10 - PARKING
*       11 - RENTACAR                   12 - DEPLCURB (DEPLANING CURB)
*       13 - IMMIGRAT (HEALTH, ETC)     14 - TICKETS&CHECKIN
*       15 - 20 ** S P A R E **
*
 9     MATRIX     MH,230,6     FACILITY TABLE *** SIZE TO NEEDS ***
*   1 ROW PER ACTUAL OR DUMMY FACILITY
*   COL 1 - FACILITY TYPE (SEE MH8 ROW DESCRIPTION FOR CODES)
*       2 - FACILITY NUMBER WITHIN TYPE
*       3 - LOCATION (POINT NUMBER)
*       4-6  USED TO IDENTIFY FACILITY WITH OTHER MODEL COMPONENTS
*            BAGCLAIM -  4 - DEPLCURB FACNO (ARV FLTS)
*            CUSTOMS  -  4 - DEPLCURB FACNO (ARV FLTS)
*            GATE     -  4 - SECURITY POINT FACNO (DEP FLTS)
*                        5 - IMMIGRATION AREA FACNO (ARV FLTS)
*            IMMIGRAT -  4 - ASSOCIATED CUSTOMS FACNO (ARV FLTS)
*            RENTACAR -  4 - AGENCY CODE (SEE FN$RCA2F)
*                        5 - PARKING FACNO(IF OTHER THAN 1, SPECIAL LOT)
*            TICK&CKIN-  4 - AIRLINE CODE
*            CONCESSI -  4 - ASSOCIATED SECURITY (=0 FOR LOBBY)
*
 11    MATRIX     MH,7,1        A COUNTER FOR PAX LEAVING CONCOURSE
*
 12    MATRIX     MH,7,1      A COUNTER FOR PAX LEAVING SECURITY
*
 13    MATRIX     MH,15,1                FOR AIRLINES
*
*   F L O A T I N G   M A T R I C E S
*
 1     MATRIX     ML,1,127               REUSABLE RANDOM NUMBER TABLE
*
 2     MATRIX     ML,3,10          GROUND TRANSACTION MODAL CHOICE
*   ROW  1 - DOMESTIC
*        2 - COMMUTER
*        3 - INTERNATIONAL
*   COL  1 - PRIVATE CAR - CURB AND PARK PERCENTAGE
*
*        2 - RENTAL CAR  CUM. % PRIVATE
*        3 - BUS/LIMO CUM. % PRIVATE
*        4 - TAXI CUM. % PRIVATE
*        6 - 10  ** S P A R E **
*
*   F U N C T I O N S
*
*   DEPLANING PAX PROCESS FUNCTIONS
*
 DDP1F FUNCTION   PB2,L6      DEPLANING DOMESTIC PROCESS FN 1
,CNCR0/,RCAR0/,BAGC0/,EXIT0/,CGTR0/,DEP99
 DCP1F FUNCTION   PB2,L5      DEPLANING COMMUTTER PROCESS FN 1
,CNCR0/,BAGC0/,EXIT0/,CGTR0/,DEP99
 DIP1F FUNCTION   PB2,L8      DEPLANING INTERNATIONAL PROCESS FN 1
,CNCR0/,IMMI0/,BAGC0/,CUST0/,RCAR0/,EXIT0/,CGTR0/,DEP99
 DLP1F FUNCTION   PB2,L4      DEPLANING LOBBY-BOUND PROCESS FN 1
,CNCR0/,CHEK0/,CGTR0/,DEP99
*
*   ENPLANING PAX PROCESS FUNCTIONS
*
 EDP1F FUNCTION   PB2,L6      DOMESTIC ENPLANING PAX - PROCESS FN 1
,CGTR1/,ENTR0/,CHEK0/,SECU0/,GATE0/,ENP99
 ECP1F FUNCTION   PB2,L6      COMMUTER ENPLANING PAX - PROCESS FN 1
,CGTR1/,ENTR0/,CHEK0/,SECU0/,GATE0/,ENP99
 EIP1F FUNCTION   PB2,L6      INTERNATIONAL ENPLANING PAX - PROCESS FN 1
,CGTR1/,ENTR0/,CHEK0/,SECU0/,GATE0/,ENP99
*
*
*   TRANSFER PAX PROCESS FUNCTIONS
*
 TIP1F FUNCTION   PB2,L8      INTERNATIONAL TRANSFER PAX - PROCESS FN 1
,CNCR0/,IMMI0/,BAGC0/,CUST0/,CHEK0/,SECU0/,GATE0/,TRX99
 TDP1F FUNCTION   PB2,L4      DOMEST/COMMUTE TRANSFER PAX - PROCESS FN 1
,CNCR0/,SECU0/,GATE0/,TRX99
 TDP2F FUNCTION   PB2,L5      DOMEST/COMMUTE TRANSFER PAX - PROCESS FN 2
,CNCR0/,SECU0/,CONC0/,GATE0/,TRX99
 TDP3F FUNCTION   PB2,L5      DOMEST/COMMUTE TRANSFER PAX - PROCESS FN 3
,CNCR0/,LOBC0/,SECU0/,GATE0/,TRX99
 TDP4F FUNCTION   PB2,L6      DOMEST/COMMUTE TRANSFER PAX - PROCESS FN 4
,CNCR0/,CHEK0/,LOBC0/,SECU0/,GATE0/,TRX99
 TDP5F FUNCTION   PB2,L5      DOMEST/COMMUTE TRANSFER PAX - PROCESS FN 5
,CNCR0/,CHEK0/,SECU0/,GATE0/,TRX99
 TDP6F FUNCTION   PB2,L2      SAME AS TDP1F FOR SAME-CONCOURSE PAX
,GATE0/,TRX99
 TDP7F FUNCTION   PB2,L3      SAME AS TDP2F FOR SAME-CONCOURSE PAX
,CONC0/,GATE0/,TRX99
 TDLDF FUNCTION   RN3,S5,Z    FUNCTION SELECTOR--LONG-STAY/DIFF CONCOURSE
.1,TDP1F/.3,TDP2F/.8,TDP3F/.9,TDP4F/1.0,TDP5F
 TDLSF FUNCTION   RN3,S5,Z    FUNCTION SELECTOR--LONG-STAY/SAME CONCOURSE
.1,TDP6F/.3,TDP7F/.8,TDP3F/.9,TDP4F/1.0,TDP5F
 TDSDF FUNCTION   RN3,S3,Z    FUNCTION SELECTOR-SHORT-STAY/DIFF CONCOURSE
.6,TDP1F/.9,TDP2F/1.0,TDP5F
 TDSSF FUNCTION   RN3,S3,Z    FUNCTION SELECTOR-SHORT-STAY/SAME CONCOURSE
.6,TDP6F/.9,TDP7F/1.0,TDP5F
 TDPXF FUNCTION   RN3,S3,Z    FUNCTION SELECTOR--TRANSIT PAX
.6,TDP6F/.9,TDP7F/1.0,TDP3F
 TOSDF FUNCTION   PB2,L2      TRANSFER-OUT-OF-SYSTEM PROCESS FN (DEPL)
,CNCR0/,TRX99
 TOSEF FUNCTION   PB2,L3      TRANSFER-OUT-OF-SYSTEM PROCESS FN (ENPL)
```

```
,SECU0/,GATE0/,TRX99
*
*   WELL-WISHER PROCESS FUNCTIONS
*
 WWS1F FUNCTION    PB2,L3     WELL-WISHERS LEFT AT SECURITY
,EXIT0/,GRT00/,ENP99
 WWG1F FUNCTION    PB2,L4     WELL-WISHERS LEFT AT GATE
,CNCR0/,EXIT0/,GRT00/,ENP99
*
*   GREETER PROCESS FUNCTIONS
*
 GREGF FUNCTION    PB2,L4     GREETERS GOING TO GATE
,GRT02/,ENTR0/,SECU0/,GATE0
 GRELF FUNCTION    PB2,L3     GREETERS GOING TO LOBBY
,GRT02/,ENTR0/,CHEK0
 GREBF FUNCTION    PB2,L3     GREETERS GOING TO BAG CLAIM
,GRT02/,ENTR0/,BAGC0
 GRECF FUNCTION    PB2,L2     GREETERS GOING TO CURB (AFTER RECIRC&PARK)
,GRT02/,DPCG0
 GRCPF FUNCTION    PB2,L2     GREETED-AT-CURB PAX
,GRT00/,DEP99
*
*   OTHER ADDRESSING FUNCTIONS
*
 CTR1F FUNCTION    PB6,L5     ROUTE DEPL PAX BY GROUND TRANSPORT MODE
,DPLC0/,GRT00/,RCAR9/,DPLC0/,DPLC0
 CTR2F FUNCTION    PB6,L5     ROUTE ENPL PAX BY GROUND TRANSPORT MODE
,ENPC0/,GRT01/,RCAR8/,ENPC2/,ENPC0
 DPL1F FUNCTION    PB6,L5     ROUTE DEPLCURB PAX BY GROUND TRANSPORT MODE
,DPLC3/,ERROR/,ERROR/,DPLC4/,DPLC5
 ENP1F FUNCTION    PB3,L3     ROUTE ENPL PAX BY DOM-COM-INT FLT
,ENPL1/,ENPL2/,ENPL3
*
*   SERVICE TIME FUNCTIONS
*
 BUN1F FUNCTION    PH5,S4,Z   BAGGAGE UNLOAD ENTITY FUN F(A/C TYPE)
9,BUN3F/10,BUN3F/11,BUN3F/727,BUN2F
 BUN2F FUNCTION    RN6,C5     BAGGAGE TIME TO CLAIM AREA - 727
0.,440/.135,630/.88,1110/.95,1260/1.0,1770
 BUN3F FUNCTION    RN6,C5     BAGGAGE TIME TO CLAIM AREA - DC8 DC10
0.,260/.065,420/.33,660/.96,930/1.0,1140
 CHK1F FUNCTION    RN7,C7                 EXPRESS CHECKIN TIME
0.0,0/0.11,60/0.54,120/0.71,180/.82,240/.89,300/1.,480
 GAT3F FUNCTION    RN7,C6     GATE PROCESS TIME
0.,0/0.43,60/.81,120/.97,180/.99,240/1.,300
 CHK2F FUNCTION    RN7,C7                 TICKET/CHECKIN TIME
0.,0/0.08,60/0.19,120/0.82,300/0.91,360/.97,420/1.,480
 IMM1F FUNCTION    RN7,C7     IMMIGRATION PROCESS TIME
.0,0/.17,30/.49,60/.81,90/.91,120/.99,180/1.0,240
 CUS1F FUNCTION    RN7,C6     CUSTOMS PROCESS TIME
0.0,0/0.106,60/0.480,120/0.750,180/0.93,300/1.0,421
 PAR1F FUNCTION    RN7,C7                 PARKING LOT EXIT SERVICE TIME
0.,0/.41,15/.64,30/.75,45/.85,60/.97,90/1.,180
 RCA1F FUNCTION    RN7,C5                 CAR RENTAL PROCESSING TIME
0.,0/.39,180/.85,360/.96,540/1.,780
 SEC1F FUNCTION    RN7,C7     SECURITY SERVICE TIME PER PERSON
0.0,0/0.47,15/0.769,30/0.842,60/0.903,120/0.945,240/1.0,480
 CSCKF FUNCTION    RN7,C6     CURBSIDE CHECKIN PROCESS TIME
0.0,0/0.168,120/0.62,240/0.81,360/0.92,540/1.0,900
 ATK1F FUNCTION    XH$CHKXH,E11    CHOSE AIRLINE TICKET/CHECKIN VARIABLE
1,V$CHK2V/2,V$CHK2V/3,V$CHK2V/4,V$CHK2V/5,V$CHK2V/6,V$CHK2V/7,V$CHK2V
```

```
8,V$CHK2V/9,V$CHK2V/10,V$CHK2V/11,V$CHK2V
*
*     DWELL TIME FUNCTIONS
*
 ENP2F FUNCTION    RN7,C7           VEHICLE UNLOAD TIME - ENPLANING CURB
0.,0/.54.15/.71,30/.83,45/.88,60/.94,90/1.,150
 ENP3F FUNCTION    RN7,C7     EMPTY VEHICLE PARK TIME - ENPLANING CURB
0.,0/.2,30/.4,45/.54,60/.72,90/.96,150/1.0,180
*
*     OTHER FUNCTIONS
*
 PPPDF FUNCTION    V$RANDV,D6             PAX/PARTY - DEPLANING
.32,1/.71,2/.91,3/.97,4/.99,5/1.0,6
 GRPPF FUNCTION    RN6,D5     GREETERS/PARTY (PARTIES W/GREETERS ONLY)
.68,1/.91,2/.95,3/.97,4/1.0,5
 DCA1F FUNCTION    RN7,C6     ARRV. DISTRIBUTIONS-CARS MEETING PAX
0.,0/.575,1800/.66,2700/.75,3600/.81,4500/1.,6300
 DCA2F FUNCTION    RN7,C6     ARRIVAL DISTRIBUTIONS - GREETERS
0.,0/.575,1800/.66,2700/.75,3600/.81,4500/1.,6300
 DBAGF FUNCTION    V$RANDV,D6 NO. OF BAGS - DOMESTIC FLIGHT
.37,0/.63,1/.83,2/.92,3/.96,4/1.0,5
 CBAGF FUNCTION    V$RANDV,D7 NO. OF BAGS - COMMUTER FLIGHT
.2,0/.52,1/.79,2/.92,3/.96,4/.98,5/1.0,6
 IBAGF FUNCTION    V$RANDV,D9  NO. OF BAGS - INTERNATIONAL FLIGHT
0.06,0/0.2,1/0.5,2/0.71,3/0.84,4/0.89,5/0.93,6/0.95,7/1.0,8
 RCA2F FUNCTION    RN7,D4     CAR RENTAL AGENCY SELECTION
.40,1/.75,2/.96,3/1.0,4
 RANDF FUNCTION    RN6,C2     USED TO GET RN 0-1 INSTEAD OF 0-999
0.0,0.0/1.0,1.0
*
*
*
*   V A R I A B L E S
*
*   SERVICE TIME VARIABLES
*
*   NOTE: WHEN PASSENGERS SCALED, MULTIPLY BY XH$SCLXH.
*
 CHK1V VARIABLE   FN$CHK1F*XH$SCLXH     EXPRESS CHECKIN
 CHK2V VARIABLE   FN$CHK2F*XH$SCLXH FULL SERVICE TICKETING TIME
 CUS1V VARIABLE   FN$CUS1F*XH$SCLXH*PB13  CUSTOMS TIME
 GAT3V VARIABLE   FN$GAT3F*XH$SCLXH*PB13 GATE SERVICE TIME
 IMM1V VARIABLE   FN$IMM1F*XH$SCLXH*PB13   IMMIGRATION TIME
 PAR1V VARIABLE   FN$PAR1F*XH$SCLXH     PARKING LOT SERVICE TIME
 RCA1V VARIABLE   FN$RCA1F*XH$SCLXH     CAR RENTAL CHECKOUT TIME
 SCLV0 VARIABLE   XH$SCLXH              DUMMY VAR - DEFINES XH$SCLXH
 SEC1V VARIABLE   8*XH$SCLXH*PB5        SECURITY SERVICE TIME (PARTY)
 CIRCV VARIABLE   300                   RECIRCULATION TIME
*
*   OTHER VARIABLES
*
 ACUNV FVARIABLE  V$ACU1V*3+90
*                                       MAX UNLOADING TIME - PAX
 ACU1V VARIABLE   MH1(PH1,10)+MH1(PH1,11)+MH1(PH1,13)+MH1(PH1,16)
*                                       TOTAL PAX ON AIRCRAFT
 ACU2V FVARIABLE  XL$ACUNL*RN6/1000.+.5   STAGGER PAX OFF A/C
 BUN4V VARIABLE   MH8(4,2)+MH1(PH1,12) BAGCLAIM INDEX + AREA NO
 DPL1V VARIABLE   30*PB4+45                DEPLCURB LOAD TIME CARS/TAXI
 DPL2V VARIABLE   RN3*SR*XF$TMPXF*60/1000000 RANDOM PULLOUT TIME <60 SEC
 DPL3V VARIABLE   PH1@125               MOD FUNCTION
 ENP1V VARIABLE   FN$ENP2F              ENPL CURB VEHICLE DWELL TIME
 ENP2V VARIABLE   180                   CURB LOAD/UNLOAD TIME - BUS/LIMO 00326000
 FLT1V VARIABLE   60*MH1(PH1,6)-AC1-3599 DELAY TO 1 HR BEFORE SCH ARRV  00327000
 FLT2V FVARIABLE  MH1(PH1,10)*XL$GRTXL*(1.-XL$CGTXL)*XL$PCBXL           00328000
*                  NO. OF PARKING GREETERS PROCEEDING TO CURB           00328100
*                GRTXL- % GREETED, CGTXL- % GREETED AT CURB             00328200
*                PCBXL- % ACCESSING PARK AND CURB                       00328300
 FLT3V FVARIABLE  MH1(PH1,10)*XL$GRTXL*(1.-XL$CGTXL)*(1.-XL$PCBXL)      00329000
*                NO. OF GREETERS PROCEEDING TO PARKING FACILITY         00329100
 FLT4V FVARIABLE  MH1(PH1,10)*XL$GRTXL*XL$CGTXL                         00330000
*                NO. OF GREETERS PROCEEDING TO CURB ONLY                00330010
 FLT5V FVARIABLE  MH1(PH1,10)*ML2(PB3,1)-MH1(PH1,5)                     00330100
*                  NO. OF TERMINATING PAX, SELF DRIVEN                  00330110
 FLT6V FVARIABLE  MH1(PH1,10)*(1.-ML2(PB3,1))                           00330200
*                  NO. OF TERMINATING PAX USING OTHER MODES             00330210
 GAT1V VARIABLE   3600-XH$BDTXH         START BOARDING PRIOR TO DEPARTURE00331000
 GAT2V VARIABLE   GAQSL+MH1(PH1,9)-1    GATE BOARDING SWITCH            00332000
 INC1V VARIABLE   XH$INCXH-1            ALLOW FOR START TIME = 1 - TIMER 00333000
 XFL1V VARIABLE   60*MH1(PH1,6)-AC1-XH$XFAXH DELAY TO ADD FLT TO XFRFLT 00334000
 XFL2V VARIABLE   60*MH1(PH1,6)-AC1-XH$XFDXH DELAY TO DLT FLT FM XFRFLT 00335000
 XFL3V VARIABLE   XH$XFDXH*RN7/2000     STAGGER XFER PAX TO SECURITY    00336000
 XFL4V VARIABLE   (60*MH1(PH1,6)-AC1)/60  TIME UNTIL FLT (MIN)          00337000
 DPCDV VARIABLE   PH6-DPCBS+DPDPS       DEPLANING CURB-DOUBLE PARK      00338000
 DPDCV VARIABLE   PH6-DPDPS+DPCBS       DEPLANING DOUBLE PARK-CURB      00339000
 DPQDV VARIABLE   PH6-DPQCS+DPDPS       DEPLANING QUEUE-DOUBLE PARK     00340000
 DPDQV VARIABLE   PH6-DPDPS+DPQCS       DEPLANING DOUBLE PARK-QUEUE     00341000
*  ABOVE VARIABLES USED FOR ENPLANING CURB ALSO                         00342000
 EPCQV VARIABLE   PH6-EPCBS+EPQCS       ENPLANING CURB-QUEUE            00343000
 EPQCV VARIABLE   PH6-EPQCS+EPCBS       ENPLANING QUEUE-CURB            00344000
 DENFV VARIABLE   300                   DEPL CURB ENFORCEMENT EVERY 5 MIN00345000
 DLIMV VARIABLE   PH8+300               DEPL CURB TIME LIMIT 5 MIN      00346000
 RANDV FVARIABLE  ML1(1,PB10)           USED AS ARGUMENT OF FUNCTIONS   00347000
 RND2V FVARIABLE  FN$RANDF*126.0+1.0    YIELDS RANDOM INDEX FOR ML1     00348000
 TMODV VARIABLE   AC1@32767             MOD FNUNCTION FOR ABS CLOCK TIME 00348001
*                                                                       00348003
*                                                                       00348006
*                                                                       00348010
* VARIABLES AND BOOLEAN VAR. FOR ENPLANING AND DEPLANING VEHICLE DELAY  00348020
*                                                                       00348030
*                                                                       00348035
 LNFLD BVARIABLE  (S*PH5'G'0)*(V$NLNMX'E'1)   TRUE IF DBL PARKING &     00348040
*                                     NO OF PASSING LANES AVAILABLE IS 1 00348050
 DPCD8 VARIABLE   PH8-DPCBS+DPDPS   DOUBLE PARKING STORAGE NUMBER       00348060
*                                       GIVEN DEPL CURB STORAGE         00348070
 CIRVD BVARIABLE  (PB10'E'4)*(PH4'E'V$DBPKD+MH8(12,1))  TR IF CIR VEH & 00348080
*                                       AT LAST STO NO OF DEPL CURB     00348085
 DBPKD VARIABLE   DPCBS-1          SET TO 1 LESS THAN FIRST DEPL CURB STO00348100
 CIRVH BVARIABLE  (PB10'E'4)*(PH4'E'V$DBPK1+MH8(8,1))   TR IF CIR VEH & 00348120
*                                       AT LAST STO NO OF ENPL CURB     00348130
 NLNM1 VARIABLE   V$NLNMX-1  ONE LESS THAN MAX NO OF PASSING LANES      00348140
 LNFUL BVARIABLE  (S*PH5'G'0)*(V$NLNMX'E'1)   TRUE IF DBL PARKING &     00348150
*                                     NO OF PASSING LANES AVAILABLE IS 1 00348160
 DBPK1 VARIABLE   EPCBS-1  SET TO 1 LESS THAN 1ST ENPL CURB STO NO      00348170
 EPCD8 VARIABLE   PH8-EPCBS+EPDPS   DOUBLE PARKING STORAGE NUMBER       00348180
*                                       GIVEN ENPL CURB STO NO          00348190
 LNDLY VARIABLE   2*3/XH$NOLAN  DELAY IN SEC DUE TO LOSS OF LANES       00348200
 NLNMX VARIABLE   3           MAX. NO. OF PASSING LANES AVAILABLE       00348210
 DEPLS VARIABLE   PH4+(PB10-1)*2*MH8(12,1)   1 LESS THAN CORR DEPL STO  00348220
*                                       OF DESTINATION                  00348225
 ENPLS VARIABLE   PH4+(PB10-1)*2*MH8(8,1)    1 LESS THAN CORR ENPL STO NO00348230
*                                       OF DESTINATION                  00348235
 EDOUT BVARIABLE  (PB1'E'5+PB1'E'24+PB1'E'25)                           00348240
 DDOUT BVARIABLE  PB1'E'2                                                00348245
 ENDIN BVARIABLE  (PB1'E'6+PB1'E'27+PB1'E'26)                            00348250
 DPDIN BVARIABLE  PB1'E'28                                               00348255
*                                                                        00349000
 SECNN VARIABLE   PH5+1-SECQS         OBTAIN CONCOURSE NUMBER FROM PH5   00349200
 TCKTN VARIABLE   PH5+1-TICQS         OBTAIN A/L NO FROM PH5             00349400
*   B O O L E A N   V A R I A B L E S                                    00350000
*                                                                        00351000
 XFL1B BVARIABLE  (MH1(PH1,1)'E'1)*(MH1(PH1,11)'G'0)  DEPT*AND*TPAX>0    00352000
 CHK1B BVARIABLE  (PB7'NE'1)+(PB9'E'1)+(RN7'GE'MH4(PB3,2))               00353000
*                     NOT ENPL .OR. NOT TICKETED .OR. NOT DIRECT         00354000
 DLIMB BVARIABLE  (V$TMODV'G'V$DLIMV)*(PB10'G'1) DEPL CURB TIME LIMIT    00355000
*                                                                        00355100
*                                                                        00355200
*                                                                        00355300
*      T A B L E S                                                       00355350
*                                                                        00355400
 HOUR1 TABLE      PH11,100,100,20      HOURLY TABLES FOR PAX WAITING TIME00355410
 HOUR2 TABLE      PH11,100,100,20                                        00355420
 HOUR3 TABLE      PH11,100,100,20                                        00355430
 HOUR4 TABLE      PH11,100,100,20                                        00355440
 HOUR5 TABLE      PH11,100,100,20                                        00355450
 HOUR6 TABLE      PH11,100,100,20                                        00355460
 HOUR7 TABLE      PH11,100,100,20                                        00355470
 HOUR8 TABLE      PH11,100,100,20                                        00355480
 HOUR9 TABLE      PH11,100,100,20                                        00355490
 HOR10 TABLE      PH11,100,100,20                                        00355495
 HOR11 TABLE      PH11,100,100,20                                        00355496
 HOR12 TABLE      PH11,100,100,20                                        00355497
 PAXWT TABLE      PH9,100,100,20      TABLE FOR PAX WALKING TIME         00355500
*                                                                        00356000
*   P A S S E N G E R   T R A N S A C T I O N   P A R A M E T E R S      00357000
*                                                                        00358000
*         HALFWORD:   1 - FLIGHT TABLE ROW NUMBER                        00359000
*                     2 - LOCATION (POINT NUMBER)                        00360000
*                     3 - MAX. BAG RANDOM NUMBER                         00361000
*                         PAX STORAGE NO IN DEPLANING CURB LOGIC         00362000
*                     4 - ADDRESS PARAMETER                              00363000
*                     5 - ** S C R A T C H **                            00364000
*                         INITIALLY CARRIES GATE NO FOR DEPL PAX         00365000
*                         FORTM USES AS RETURN FOR STO, QUE, NOS, ETC.   00366000
*                     6 - USER CHAIN FOR BAG CLAIM (DEPL. PAX ONLY)      00367000
*                         CAR STORAGE NO IN DEPLANING CURB LOGIC         00368000
*                     7 - MH9 ROW FOR LAST FACILITY                      00369000
*                         (NOT IMPLEMENTED IN EXIT LOGIC)                00370000
*                     8 - ** S C R A T C H **                            00371000
*                         FORTM USES AS RETURN FOR ADDR IN XFER PAX LOGIC 00372000
*                     9 - CUMULATIVE WALKING TIME                        00373000
*                    10 - XAC SEQUENCE NUMBER FOR DEPL PAX MEETING       00374000
*                    11 - CUMULATIVE PAX WAITING TIME                    00374100
*         BYTE:       1 - PROCESS FUNCTION NUMBER                        00375000
*                     2 - PROCESS FUNCTION POINTER                       00376000
*                     3 - TYPE OF FLIGHT (1=DOM, 2=COM, 3=INT)           00377000
*                     4 - NUMBER OF BAGS                                 00378000
*                     5 - NUMBER IN PARTY (VISITORS + SELF)              00379000
*                     6 - MODE OF GROUND TRANSPORTATION                  00380000
*                         (SEE MATRIX ML2 FOR CODES)                     00381000
*                     7 - 0=DEPLANING, 1=ENPLANING                       00382000
*                     8 - 1 = TERMINATING, 2 = TRANSFER                  00383000
*                         3 = TRANSIT, 4 = TRANSFER OUT OF SYSTEM        00384000
*                     9 - BAG CLAIM AREA LOG SWITCH (DEPL. PAX ONLY)     00385000
*                      - 0 = TICKETED, 1 = NOT (ENPL. PAX ONLY)
*                   10 - USED IN DEPLANING FOR SCRATCH
*                      - RENTACAR AGENCY
*                   11 - CURRENT PROCESS CODE
*                        (SEE MH8 FOR CODES)
*                        (NOT IMPLEMENTED IN EXIT LOGIC)
*                   12 - =1 FOR MEETING AT GATE (DEPL PAX ONLY)
*                        =2 FOR MEETING AT BAG CLAIM (DEPL ONLY)
*                        =3 FOR MEETING AT TICKETING (DEPL ONLY)
*                   13 - PAX IN PARTY
*                   14 - # OF BAGS FN NUMBER; LATER PARKING LOT NUMBER
*
*...C O N T R O L  ... USED TO ROUTE TRANSACTIONS
*   THIS SECTION ROUTES TRANSACTIONS FROM ONE LOGIC SECTION TO ANOTHER
*                 ALSO CONTAINS DGTR* CODE - DEPLANING GROUND TRANSPORT.
*
 CTRL0 ASSIGN     4,FN*PB1,PH             ADDRESS OF NEXT OPERATION
       ASSIGN     2+,1,PB                 INCREMENT PROCESS FN POINTER
 CTRL1 TRANSFER   ,PH4
*
*   HANDLES ROUTING OF DEPL PAX TO PARKING LOTS.
*
 CTRL8 TRANSFER   ,PARK0
*
*   THE FOLLOWING SECTIONS OF "CONTROL" HANDLE PAX IN GROUND TRANSPORT.
*
 CGTR0 ASSIGN     4,FN$CTR1F,PH           ROUTE DEPL PAX BY GR TX MODE
       TRANSFER   ,PH4
 CGTR1 ASSIGN     4,FN$CTR2F,PH           ROUTE ENPL PAX BY GR TX MODE
       TRANSFER   ,PH4
*
*   THIS SECTION OF "CONTROL" GENERATES THE BUS/LIMO SERVICE TO THE
*   ENPLANING & DEPLANING CURBS.   THE CURRENT MODEL SIMULATES A SINGLE
*   STOP AT EACH, THOUGH THE ENPLANING CURB LOGIC OF THE FORTRAN ROUTINE
*   "FORTM" SUPPORTS A MULTIPLE STOP SIMULATION WHEN DESIRED.   COMPLEX
*   BUS SCHEDULES CURRENTLY REQUIRE INDIVIDUAL PROGRAMMING.
*
       GENERATE   ,,,1,,2PH
       ASSIGN     2,1,PH
       SPLIT      1,CGTR3
       TEST E     XH$ABUXH,0,*+2          FT IF NO ARV BUS SIMULATED
       TERMINATE
 CGTR2 ADVANCE    XH$ABUXH                INTERVAL BETWEEN ARRIVING BUSSES
       SPLIT      1,CGTR2
       TRANSFER   ,ENPC6
 CGTR3 TEST E     XH$DBUXH,0,*+2          FT IF NO DEP BUS SIMULATED
       TERMINATE
 CGTR4 ADVANCE    XH$DBUXH                INTERVAL BETWEEN DEPARTING BUSSES
       SPLIT      1,CGTR4
       TRANSFER   ,DPLC6
*
*   PASSENGER TERMINATION
*
 DEP99 TABULATE   PAXWT                   PASSENGER WALKING TIME
       TABULATE   XB$PXTBN                PASSENGER WAITING TIME
       TERMINATE
 ENP99 TABULATE   PAXWT                   PASSENGER WALKING TIME
       TABULATE   XB$PXTBN                PASSENGER WAITING TIME
       TERMINATE
 TRX99 TABULATE   PAXWT                   PASSENGER WALKING TIME
       TABULATE   XB$PXTBN                PASSENGER WAITING TIME
       TERMINATE                                                            00441200
*                                                                           00442000
*...E N D   O F   C O N T R O L                                             00443000
*                                                                           00444000
*                                                                           00445000
*   D E P L A N I N G   P A S S E N G E R   L O G I C                       00446000
*                                                                           00447000
       GENERATE    ,,,1,10,11PH,14PB                                        00448000
       ASSIGN      2,1,PB                  INTIIALIZE PROCESS FN POINTER    00449000
       SPLIT       1,XFLTO                 COPY TO TRANSFER FLT LOGIC       00450000
       ADVANCE     1                       ASSURE XFER FLT TABLE GETS BUILT 00451000
       ASSIGN      11,1,PB                 DEPLANING PAX AT "GATE"          00452000
 DEPLO ASSIGN      1+,1,PH                 FLT TABLE (MH1) ROW SUBSCRIPT    00453000
       TEST L      MH1(PH1,1).0,*+2        MH1(PH1,1)=-1 --> TABLE END (*+2)00454000
       TERMINATE                                                            00455000
       ADVANCE     V$FLT1V                 DELAY TO 1 HR BEFORE SCH ARRV    00456000
       SPLIT       1,DEPLO                                                  00456100
       TEST E      MH1(PH1,1),1,DEPL5      BR FOR ARRIVING FLIGHTS          00457000
       ASSIGN      4,GATE9,PH              FOR GATE LOGIC TO START BOARDING 00459000
       TRANSFER    ,CTRL1                                                   00460000
 DEPL5 ASSIGN      3,MH1(PH1,7),PB         TYPE FLT (1,2,3 = DOM,COM,INT)   00461000
*                                                                           00462000
*      CREATE VISITORS HERE      *****                                      00463000
*                                                                           00464000
       TEST E      PB3,1,DEPL1             TESTS FOR DOMESTIC FLIGHT        00465000
       ASSIGN      1.DDP1F,PB              DEPL DOMESTIC ROUTE FN           00466000
       ASSIGN      14,DBAGF,PB             BAGS - DOMESTIC FLIGHT           00467000
       TRANSFER    ,DEPL3                                                   00468000
 DEPL1 TEST E      PB3,2,DEPL2             TEST FOR COMMUTER FLIGHT         00469000
       ASSIGN      1,DCP1F,PB              DEPL COMMUTER ROUTE FN           00470000
       ASSIGN      14,CBAGF,PB             BAGS - COMMUTER FLIGHT           00471000
       TRANSFER    ,DEPL3                                                   00472000
 DEPL2 ASSIGN      1,DIP1F,PB              DEPL INTERNAT ROUTE FN           00473000
       ASSIGN      14,IBAGF,PB             BAGS - INTERNATIONAL FLIGHT      00474000
 DEPL3 TEST GE     XH$SEQ1H.32000,*+2      XAC SEQUENCER              (*+2) 00475000
       SAVEVALUE   SEQ1H,0,XH                                               00476000
       SPLIT       1,DEP25                 TO CURB W/GREETERS LOGIC         00477000
       SPLIT       1,DEP15                 TO PARK W/GREETERS LOGIC         00478000
       SPLIT       1,DEP17                 TO CURB W/O GREETERS LOGIC       00479000
       TRANSFER    ,DEPL7                                                   00481000
*                                                                           00482000
*   CURB WITH GREETERS LOGIC                                                00483000
*                                                                           00484000
 DEP25 ASSIGN      6,1,PB                  MODE = PARK                      00485000
       ASSIGN      8,V$FLT2V,PH            NO. OF GREETERS                  00485100
       TRANSFER    ,DEP16                                                   00486000
*                                                                           00487000
*   PARK WITH GREETERS LOGIC                                                00488000
*                                                                           00489000
 DEP15 ASSIGN      6,2,PB                  MODE = PARK                      00490000
       ASSIGN      8,V$FLT3V,PH            NO. OF PAX TO BE GREETED         00491000
 DEP16 TEST E      PH8,0,*+2               FT IF NONE                 (*+2) 00492000
       TERMINATE                                                            00493000
       MSAVEVALUE  1+,PH1.5,PH8,MH         ADD TO TOTAL MET                 00494000
       ASSIGN      10,V$DPL3V,PB           STARTING POINT IN R.N. TABLE     00495000
 DEP12 ASSIGN      10+.1,PB                INCREMENT R.N. TABLE POINTER     00496000
       TEST G      PB10,124,*+2                                       (*+2)00497000
       ASSIGN      10,1,PB                 TURN CORNER                      00498000
       ASSIGN      8-,FN$PPPDF,PH          SUBTRACT # OF PAX FROM TOTAL     00499000
       ASSIGN      10+,1,PB                INCR POINTER                     00500000
       ASSIGN      4,FN*PB14,PB            COMPUTE # OF BAGS                00501000
```

```
      ASSIGN     10+,1,PB              INCR POINTER                        00502000
      TEST LE    ML1(1,PB10),XL$GRGXL,*+4 FT FOR GATE MEETING      (*+4)  00503000
      ASSIGN     1,GREGF,PB            MEET AT GATE                        00504000
      ASSIGN     12,1,PB                                                   00505000
      TRANSFER   ,DEP14                                                    00506000
      TEST E     PB4,0,*+3             FT FOR NO BAGS              (*+3)  00507000
      ASSIGN     1,GRELF,PB            MEET IN LOBBY                       00508000
      TRANSFER   ,DEP14                                                    00509000
      ASSIGN     1,GREBF,PB            MEET IN BAG CLAIM                   00510000
DEP14 TEST G     PH8,0,*+2             FT IF NEED MORE PARTIES     (*+2)  00511000
      SPLIT      1,DEP12                                                   00512000
      ASSIGN     14,0,PB               RESET PB14 FOR LATER USE            00513000
      ASSIGN     5,FN$GRPPF,PB         # OF GREETERS                       00514000
      SAVEVALUE  SEQ1H+,1,XH           INCR XAC COUNTER                    00515000
      ASSIGN     10,XH$SEQ1H,PH        XAC SEQUENCE #                      00516000
      ADVANCE    FN$DCA2F              TO AIRPORT                          00517000
      TRANSFER   ,CTRL0                                                    00518000
*                                                                          00519000
*  CURB WITH GREETERS LOGIC (MEET AT CURB)                                 00520000
*                                                                          00521000
DEP17 ASSIGN     6,1,PB                MODE = CURB                         00522000
      ASSIGN     8,V$FLT4V,PH          NO. OF PAX TO BE MET                00523000
      TEST E     PH8,0,*+2             FT IF NONE                          00524000
      TERMINATE                                                            00525000
      MSAVEVALUE 1+,PH1,5,PH8,MH       ADD TO TOTAL MET                    00526000
      ASSIGN     10,V$DPL3V,PB         STARTING POINT IN R.N. TABLE        00527000
      ASSIGN     4,DCAR0,PH            WILL GO TO CURB                     00528000
DEP13 ASSIGN     10+,1,PB              INCREMENT R.N. TABLE POINTER        00529000
      TEST G     PB10,124,*+2                                       (*+2)00530000
      ASSIGN     10,1,PB               TURN CORNER                         00531000
      ASSIGN     8-,FN$PPPDF,PH        SUBTRACT # OF PAX FROM TOTAL        00532000
      ASSIGN     10+,1,PB              INCR POINTER                        00533000
      ASSIGN     4,FN*PB14,PB          COMPUTE # OF BAGS                   00534000
      TEST G     PH8,0,*+2             FT IF NEED MORE PARTIES     (*+2)  00535000
      SPLIT      1,DEP13                                                   00536000
      ASSIGN     14,0,PB               RESET PB14 FOR LATER USE            00537000
      ASSIGN     5,FN$GRPPF,PB         NOT USED UNLESS RECIRC&PARK         00538000
      SAVEVALUE  SEQ1H+,1,XH           INCR XAC COUNTER                    00539000
      ASSIGN     10,XH$SEQ1H,PH        XAC SEQUENCE #                      00540000
      TRANSFER   ,CTRL1                                                    00541000
*                                                                          00542000
*   NOTE: ADVANCE TIME MUST BE DETERMINISTIC                               00543000
DEPL7 ADVANCE    3600                                                      00544000
      GATE LR    DPL1G                 GUARD "BAGS" FM SIMUL ARRV A/C      00546000
      LOGIC S    DPL1G                                                     00547000
      ASSIGN     5,MH1(PH1,9),PH       GATE NO                             00548000
      ASSIGN     7,PH5,PH              LAST MH9 ROW                        00549000
      ASSIGN     2,MH9(PH5,3),PH       POINT NO OF GATE                    00550000
      SAVEVALUE  ACUNL,V$ACUNV,XL      MAX PASSENGER UNLOADING TIME        00551000
      SELECT LR  6PH,CHA1B,CHA2B       PICK FREE BAG CLAIM CHAIN-SWITCH    00552000
      LOGIC S    PH6                   MARK THIS CHAIN IN USE              00553000
      SPLIT      1,DEPL4               COPY TO TERMINATING PAX             00554000
      SPLIT      1,DEPL6               COPY TO TRANSIT PAX                 00555000
      SPLIT      1,DEP10               COPY TO TRANSFER-OUT-OF-SYSTEM      00556000
      TEST E     MH5(1,1),0,DEPL8      FALL THRU IF NO DEP FLTS IN XFER    00557000
      SAVEVALUE  XFRXH+,MH1(PH1,11),XH   ADD XFER PAX TO "HOLDING XH"      00558000
      TERMINATE                                                            00559000
DEPL8 TEST E     MH1(PH1,11),0,*+2     FT IF NO TPAX                (*+2)00560000
      TERMINATE                                                            00561000
      ASSIGN     8,2,PB                MARK AS TRANSFER PAX                00562000
      ASSIGN     13,1,PB               PAX IN PARTY                        00563000
```

```
       ASSIGN     5,1,PB                    TOTAL IN PARTY                   00564000
       SPLIT      MH1(PH1,11),*+2           CREATE TRANSFER PAX              00565000
       TERMINATE                                                             00566000
       HELPA      FORTM,17,1,RN5,PB3,PH5                                     00567000
       TEST E     PB3,3,DEP30               FT IF INTERNATIONAL              00568000
       ASSIGN     1,TIP1F,PB                INT TRANSFER PROCESS FUNCTION    00569000
       ASSIGN     10,V$DPL3V,PB             STARTING POINT IN R.N. TABLE     00570000
       TEST E     MH1(PH1,7),3,*+5          FT IF INT FLIGHT          (*+5)  00570100
       ASSIGN     1,TDP5F,PB                ASSIGN DOM/COM TRANSFER FUNCTION 00570200
       ASSIGN     9,1,PB                    SET PAX AS NOT TICKETED          00570300
       HELP       BAGS,PH1,FN*PB14,4,3,PB8   GENERATE # OF BAGS, MAX RN      00571000
       TRANSFER   ,DEP31                                                     00572000
DEP30  TEST E     MH1(PH1,7),3,*+4          IS OUT GOING FLIGHT INTERNATIONAL00572100
       ASSIGN     1,TDP4F,PB                SKIP IMMIO,BAGCO,CUSTO           00572200
DEP31  ADVANCE    V$ACU2V                   STAGGER PAX OFF A/C              00572300
       TRANSFER   ,DEP11                                                     00573000
       SAVEVALUE  SAVXH,MH1(PH1,9),XH       GATE # OF DEP FLT                00574000
       TEST L     V$XFL4V,45,*+6            FT IF DEP IN <45 MIN      (*+6)  00575000
       TEST E     MH9(PH5,4),MH9(XH$SAVXH,4),*+3 FT IF SAME SEC (*+3)        00576000
       ASSIGN     1,FN$TDSSF,PB             SHORT-STAY/SAME                  00577000
       TRANSFER   ,DEP11                                                     00578000
       ASSIGN     1,FN$TDSDF,PB             SHORT-STAY/DIFF                  00579000
       TRANSFER   ,DEP11                                                     00580000
       TEST E     MH9(PH5,4),MH9(XH$SAVXH,4),*+3 FT IF SAME SEC (*+3)        00581000
       ASSIGN     1,FN$TDLSF,PB             LONG-STAY/SAME                   00582000
       TRANSFER   ,DEP11                                                     00583000
       ASSIGN     1,FN$TDLDF,PB             LONG-STAY/DIFF                   00584000
DEP11  TRANSFER   ,PH8                                                       00585000
*                                                                            00586000
DEPL6  ASSIGN     8,3,PB                    MARK AS TRANSIT                  00587000
       ASSIGN     13,1,PB                   PAX IN PARTY                     00588000
       ASSIGN     5,1,PB                    TOTAL IN PARTY                   00589000
       SPLIT      MH1(PH1,13),*+2           CREATE TRANSIT PAX        (*+2)  00590000
       TERMINATE                                                             00591000
       ADVANCE    V$ACU2V                   STAGGER PAX OFF AIRCRAFT         00592000
       HELPA      FORTM,17,2,PH1            SW,ARR FLT                       00593000
       ASSIGN     1,FN$TDPXF,PB             SELECT A PROCESS FUNCTION        00594000
       TEST E     PB1,0,*+2                 FT IF NEVER DEPLANED      (*+2)  00595000
       TERMINATE                                                             00596000
       TRANSFER   ,PH8                                                       00597000
*                                                                            00598000
DEP10  ASSIGN     8,4,PB                    MARK AS TRANSFER-OUT-OF-SYSTEM   00599000
       ASSIGN     13,1,PB                   PAX IN PARTY                     00600000
       ASSIGN     5,1,PB                    TOTAL IN PARTY                   00601000
       SPLIT      MH1(PH1,16),*+2           CREATE TRANSFERS O.O.S.     (*+2)00602000
       TERMINATE                                                             00603000
       ADVANCE    V$ACU2V                   STAGGER PAX OFF AIRCRAFT         00604000
       ASSIGN     1,TOSDF,PB                PROCESS FN FOR T.O.O.S. (DEPL)   00605000
       TRANSFER   ,CTRL0                                                     00606000
*                                                                            00607000
DEPL4  ASSIGN     8,1,PB                    MARK AS TERMINATING PAX          00608000
       TEST GE    XH$SEQ2H,32000,*+2        XAC SEQUENCER             (*+2)  00609000
       SAVEVALUE  SEQ2H,0,XH                                                 00610000
       SPLIT      1,DEP18                   TO CURB W/GREETERS LOGIC         00611000
       SPLIT      1,DEP19                   TO PARK W/GREETERS LOGIC         00612000
       SPLIT      1,DEP22                   TO CURB W/O GREETERS LOGIC       00613000
       SPLIT      1,DEP26                   TO PARK W/O GREETERS LOGIC       00614000
       SPLIT      1,*+3                     TO ALL OTHER MODES        (*+3)  00615000
       ASSIGN     4,BUNLO,PH                                                 00616000
       TRANSFER   ,CTRL1                                                     00617000
*                                                                            00618000
*  MODES OTHER THAN CURB AND PARK                                             00619000
*                                                                             00620000
        ASSIGN     6,2,PB                  ASSIGN MODE                        00621000
        ASSIGN     8,V$FLT6V,PH  NO. OF PAX IN MODES OTHER THAN CURB/PARK00622000
        TEST E     PH8,0,*+2               FT IF NONE                         00623000
        TERMINATE                                                             00624000
        ASSIGN     10,V$DPL3V,PB           STARTING POINT IN R.N. TABLE       00625000
 DEPL9  ASSIGN     10+,1,PB                INCREMENT R.N. TABLE POINTER       00626000
        TEST G     PB10,124,*+2                                         (*+2)00627000
        ASSIGN     10,1,PB                 TURN CORNER                        00628000
        ASSIGN     13,FN$PPPDF,PB          PAX IN PARTY                       00629000
        ASSIGN     8-,PB13,PH              SUBTRACT FROM TOTAL                00630000
        TEST LE    PH8,0,*+3               FT IF TOTAL <= 0                   00631000
        ASSIGN     13+,PH8,PB              ADJUST LAST PARTY SIZE             00632000
        TRANSFER   ,*+2                    OUT OF LOOP                 (*+2)  00633000
        SPLIT      1,DEPL9                 GO BACK TO CREATE ANOTHER          00634000
        ASSIGN     5,PB13,PB               TOTAL IN PARTY                     00635000
        HELPA      FORTM,5,0,RNS,PB3       PICK GROUND TRANSP. MODE           00636000
        ASSIGN     10+,1,PB                INCR POINTER                       00637000
        HELP       BAGS,PH1,FN*PB14,4,3,PB8  GEN # OF BAGS, MAX RN            00638000
        ASSIGN     14,0,PB                 RESET PB14 FOR LATER USE           00639000
        TEST E     PB6,3,*+3               IF RENT-A-CAR USE DDP1F  (*+3)     00639100
        TEST E     PB4,0,*+2               FT IF NO BAGS               (*+2)  00640000
        ASSIGN     1,DLP1F,PB              CHANGE TO LOBBY PROC FN            00641000
        TRANSFER   ,DEP24                                                     00642000
*                                                                             00643000
*  CURB WITH GREETERS                                                         00644000
*                                                                             00645000
DEP18   ASSIGN     6,1,PB                  MODE = CURB                        00646000
        ASSIGN     8,V$FLT2V,PH NO. OF PRKING GREETERS PROCEEDING TO CURB00646100
        TRANSFER   ,DEP20                                                     00647000
*                                                                             00648000
*  PARK WITH GREETERS                                                         00649000
*                                                                             00650000
DEP19   ASSIGN     6,2,PB                  MODE = PARK                        00651000
        ASSIGN     8,V$FLT3V,PH            NO. OF PAX TO BE GREETED           00652000
DEP20   TEST E     PH8,0,*+2               FT IF NONE                         00653000
        TERMINATE                                                             00654000
        ASSIGN     10,V$DPL3V,PB           STARTING POINT IN R.N. TABLE       00655000
DEP21   ASSIGN     10+,1,PB                INCREMENT R.N. POINTER             00656000
        TEST G     PB10,124,*+2   PREVENT POINTER FROM EXCEEDING 127(*+2)00657000
        ASSIGN     10,1,PB                 RESET POINTER                      00658000
        ASSIGN     13,FN$PPPDF,PB          PAX IN PARTY                       00659000
        ASSIGN     8-,PB13,PH              SUBTRACT GROUP FROM TOTAL          00660000
        ASSIGN     12,2,PB                 MEET AT BAGCLAIM                   00661000
        ASSIGN     10+,1,PB                INCREMENT R.N. POINTER             00662000
        HELP       BAGS,PH1,FN*PB14,4,3,PB8  GEN # OF BAGS, MAX RN            00663000
        ASSIGN     10+,1,PB                INCREMENT R.N. POINTER             00664000
        TEST LE    ML1(1,PB10),XL$GRGXL,*+2 FT FOR GATE MEETING   (*+2)       00665000
        ASSIGN     12,1,PB                 MEET AT GATE                       00666000
        TEST LE    PH8,0,*+3               IS LAST PARTY SIZE TOO LARGE(*+3)00670000
        ASSIGN     13+,PH8,PB              ADJUST LAST PARTY SIZE             00671000
        TRANSFER   ,*+2                                            (*+2)      00672000
        SPLIT      1,DEP21                                                    00673000
        TEST NE    PB3,3,*+4               FT IF NOT INTERNATIONAL    (*+4)   00673050
        TEST E     PB4,0,*+3               FT IF NO BAGS          (*+3)       00673100
        ASSIGN     1,DLP1F,PB              CHANGE TO LOBBY PROC FN            00673200
        ASSIGN     12,3,PB                 MEET AT TICKETING                  00673300
        ASSIGN     14,0,PB                 RESET PB14 FOR LATER USE           00674000
        ASSIGN     5,PB13,PB               TOTAL IN PARTY                     00675000
        SAVEVALUE  SEQ2H+,1,XH             INCR XAC COUNTER                   00676000
```

```
       ASSIGN     10,XH$SEQ2H,PH        XAC SEQUENCE #                      00677000
       TRANSFER   ,DEP24                                                    00676000
*                                                                           00679000
*  CURB WITHOUT GREETERS                                                    00680000
*                                                                           00681000
 DEP22 ASSIGN     6,1,PB                MODE = CURB                         00682000
       ASSIGN     8,V$FLT4V,PH          NO. OF PAX TO BE MET                00683000
       TEST E     PH8.0,*+2             FT IF NONE                          00684000
       TERMINATE                                                            00685000
       ASSIGN     10,V$DPL3V,PB         STARTING POINT IN R.N. TABLE        00686000
 DEP23 ASSIGN     10+,1,PB              INCREMENT R.N. POINTER              00687000
       TEST G     PB10,124,*+2   PREVENT POINTER FROM EXCEEDING 127(*+2)00688000
       ASSIGN     10,1,PB               RESET POINTER                       00689000
       ASSIGN     13,FN$PPPDF,PB        PAX IN PARTY                        00690000
       ASSIGN     8-,PB13,PH            SUBTRACT GROUP FROM TOTAL           00691000
       ASSIGN     10+,1,PB              INCREMENT R.N. POINTER              00692000
       HELP       BAGS,PH1,FN*PB14,4,3,PB8  GEN # OF BAGS, MAX RN           00593000
       TEST LE    PH8,0,*+3             IS LAST PARTY SIZE TOO LARGE(*+3)00696000
       ASSIGN     13+,PH8,PB            ADJUST LAST PARTY SIZE              00697000
       TRANSFER   ,*+2                                                      00698000
       SPLIT      1,DEP23                                                   00699000
       ASSIGN     14,0,PB               RESET PB14 FOR LATER USE            00700000
       ASSIGN     5,PB13,PB             TOTAL IN PARTY                      00701000
       SAVEVALUE  SEQ2H+,1,XH           INCR XAC COUNTER                    00702000
       ASSIGN     10,XH$SEQ2H,PH        XAC SEQUENCE #                      00703000
       TRANSFER   ,DEP24                                                    00704000
*                                                                           00705000
*  PARK WITHOUT GREETERS                                                    00706000
*                                                                           00707000
 DEP26 ASSIGN     6.2,PB                MODE = PARK                         00708000
       ASSIGN     8,V$FLT5V,PH          NO. OF PAX TO BE CREATED            00709000
       TEST E     PH8.0,*+2             FT IF NONE                          00710000
       TERMINATE                                                            00711000
       ASSIGN     10,V$DPL3V,PB         STARTING POINT IN R.N. TABLE        00712000
 DEP27 ASSIGN     10+,1,PB              INCREMENT R.N. POINTER              00713000
       TEST G     PB10,124,*+2   PREVENT POINTER FROM EXCEEDING 127(*+2)00714000
       ASSIGN     10,1,PB               RESET POINTER                       00715000
       ASSIGN     13,FN$PPPDF,PB        PAX IN PARTY                        00716000
       ASSIGN     8-,PB13,PH            SUBTRACT GROUP FROM TOTAL           00717000
       TEST LE    PH8,0,*+3             IS LAST PARTY SIZE TOO LARGE(*+3)00718000
       ASSIGN     13+,PH8,PB            ADJUST LAST PARTY SIZE              00719000
       TRANSFER   ,*+2                                              (*+2)  00720000
       SPLIT      1,DEP27                                                   00721000
       ASSIGN     5,PB13,PB             TOTAL IN PARTY                      00722000
       ASSIGN     10+.1,PB              INCREMENT R.N. POINTER              00723000
       HELP       BAGS,PH1,FN*PB14,4,3,PB8  GEN # OF BAGS, MAX RN           00724000
       ASSIGN     14,0,PB               RESET PB14 FOR LATER USE            00725000
       TEST E     PB4,0,*+2             FT IF NO BAGS                 (*+2) 00726000
       ASSIGN     1,DLP1F,PB            CHANGE TO LOBBY PROC FN             00727000
*                                                                           00728000
 DEP24 ADVANCE    V$ACU2V               STAGGER PAX OFF AIRCRAFT            00729000
       TEST NE    PB12,1,*+2            FT IF NOT MEETING AT GATE   (*+2)   00730000
       TRANSFER   ,CTRLO                                                    00731000
       UNLINK     GREGC,CTRL1,1,10PH,PH10,DEP28 TRY TO UNLINK GREETER       00732000
 DEP29 GATE LS    PAS3L                 WAIT FOR INFO FROM GREETER          00733000
       ASSIGN     5+,XB$PAS32,PB        ADD GREETERS TO PARTY               00734000
       ASSIGN     14,XB$PAS33,PB        PICK UP PARKING LOT NUMBER          00735000
       LOGIC R    PAS3L                 PERMIT NEXT USE OF PAS3 S'VALUES    00736000
       TRANSFER   ,CTRLO                                                    00737000
 DEP28 ASSIGN     4,DEP29,PH            WILL GO TO DEP29 WHEN UNLINKED      00738000
       LINK       GREGC,FIFO            WAIT FOR GREETERS                   00739000
```

```
*                                                                       00740000
*                                                                       00741000
*...E N P L A N I N G   P A S S E N G E R   L O G I C                   00742000
*                                                                       00743000
 ENPL0 TRANSFER   ,ENPL9                                                00744000
 ENPL1 TEST E     PB8,4,*+4            FT IF TRANSFER FROM OUT OF SYS.  00745000
       ASSIGN     2,1,PH               ETTR.FACIL.FOR OUT OF SYS.TRANSF.00745100
       ASSIGN     1,TOSEF,PB                                            00746000
       TRANSFER   ,CTRL0                                                00747000
       ASSIGN     1,EDP1F,PB           ENPLANING DOMESTIC PASSENGERS    00748000
       ASSIGN     10,V$RND2V,PB        GET RANDOM INDEX FOR BAG FUNCTION00749000
       ASSIGN     4,FN$DBAGF,PB        NO OF BAGS - DOMESTIC            00750000
       TRANSFER   ,CTRL0                                                00752000
 ENPL2 ASSIGN     1,ECP1F,PB           ENPLANING COMMUTER PAX           00753000
       ASSIGN     10,V$RND2V,PB        GET RANDOM INDEX FOR BAG FUNCTION00754000
       ASSIGN     4,FN$CBAGF,PB        NO OF BAGS - COMMUTER            00755000
       TRANSFER   ,CTRL0                                                00757000
 ENPL3 ASSIGN     1,EIP1F,PB           ENPLANING INTERNATIONAL PAX      00758000
       ASSIGN     10,V$RND2V,PB        GET RANDOM INDEX FOR BAG FUNCTION00759000
       ASSIGN     4,FN$IBAGF,PB        NO OF BAGS - INTERNATIONAL       00760000
       TRANSFER   ,CTRL0                                                00762000
 ENPL9 CHANGE     ENPL0,CHNG0                                           00763000
       TERMINATE                                                        00764000
 CHNG0 TRANSFER   FN,ENP1F                                              00765000
*                                                                       00766000
*                                                                       00767000
*...B A G C L A I M                                                     00768000
*                                                                       00769000
 BAGC0 TEST G     PB4,0,CTRL0          RETURN TO CONTROL IF NO BAGS     00770000
       HELPA      FORTM,3,PH2,PH1      LOC,MH1 ROW                      00771000
       ADVANCE    XH$TRVXH             TRAVEL TIME TO BAG CLAIM AREA    00772000
       TEST E     PB13,0,BAGC1         FT IF GREETER                    00773000
       UNLINK     GREBC,CTRL1,1,10PH,PH10,BAGC2 TRY TO UNLINK PAX       00774000
 BAGC3 GATE LR    PAS4L                ONLY ONE AT A TIME               00775000
       SAVEVALUE  PAS42,PB5,XB         NUMBER OF GREETERS               00776000
       SAVEVALUE  PAS43,PB14,XB        PARKING LOT NUMBER               00777000
       LOGIC S    PAS4L                LET PAX PICK UP INFO             00778000
       TEST NE    PB6,1,*+2            BR IF MODE IS CURB          (*+2)00779000
       TERMINATE                                                        00780000
       ASSIGN     5,0,PB               NO-ONE IN PARTY                  00781000
       ASSIGN     4,GRT03,PH           ROUTE TO GREETER LEAVING PARKING 00782000
       TRANSFER   ,CTRL1                                                00783000
*                                                                       00784000
 BAGC2 ASSIGN     4,BAGC3,PH           WILL GO TO BAGC3 WHEN UNLINKED   00785000
       LINK       GREBC,FIFO                                            00786000
*                                                                       00787000
 BAGC1 SAVEVALUE  PH2+,PB5,XH          CONGESTION                       00788000
       TEST E     MH1(PH1,1),0,BAGC8   BR IF ALL BAGS AT BAGCLAIM       00789000
       ASSIGN     4,BAGC7,PH                                            00790000
       MARK                            MARK PAX FOR WAITING TIME        00790100
       LINK       PH6,FIFO             WAIT FOR BAGS (UNLK IN BUNL*)    00791000
 BAGC7 ASSIGN     11+,M1,PH            ADD WAITING TIME TO PH11         00791100
 BAGC8 SAVEVALUE  PH2-,PB5,XH          CONGESTION                       00792000
       TEST E     PB12,2,BAGC4         FT IF TO BE GREETED AT BAGCLAIM  00793000
       UNLINK     GREBC,CTRL1,1,10PH,PH10,BAGC5 TRY TO UNLINK GREETER   00794000
 BAGC6 GATE LS    PAS4L                WAIT FOR INFO FROM GREETER       00795000
       ASSIGN     5+,XB$PAS42,PB       ADD GREETERS TO PARTY            00796000
       ASSIGN     14,XB$PAS43,PB       PICK UP PARKING LOT NUMBER       00797000
       LOGIC R    PAS4L                PERMIT NEXT USE OF PAS4 S'VALUES 00798000
       TRANSFER   ,BAGC4                                                00799000
*                                                                       00800000
```

```
 BAGC5 ASSIGN     4,BAGC6,PH               WILL GO TO BAGC6 WEN UNLINKED     00801000
       LINK       GREBC,FIFO                                                 00802000
*                                                                            00803000
 BAGC4 TRANSFER   ,CTRLO                                                     00804000
*                                                                            00805000
*                                                                            00806000
*...B A G G A G E   U N L O A D I N G   L O G I C                            00807000
*                                                                            00808000
 BUNLO PRIORITY   5,BUFFER                 ALLOW PAX XAC TO EXECUTE "BAGS"   00809000
       SPLIT      1,*+2,,9PH,40PB                                   (*+2)    00810000
       TERMINATE                                                             00811000
       HELPA      FORTM,2,10               BAGGAGE UNLOADING LOGIC           00812000
       ASSIGN     5,MH1(PH1,6),PH          A/C TYPE                          00813000
       ASSIGN     5,FN$BUN1F,PH            UNLOAD TIME FUN F(A/C TYPE)       00814000
       LOGIC R    DPL1G                                                      00815000
*                                                                            00817000
       ADVANCE    FN*PH5                   BAGGAGE UNLOADING TIME            00819000
*                                                                            00820000
       ASSIGN     5,V$BUN4V,PH             BAGCLAIM INDEX + AREA NO          00821000
       ASSIGN     5,MH9(PH5,3),PH          POINT NO OF BAG CLAIM             00822000
       ASSIGN     4,40,PH                  LOOP COUNTER                      00823000
 BUNL2 TEST G     PB*PH4,0,BUNL3           BR IF ALL BAGS UNLOADED           00824000
       ADVANCE    30                       BAG UNLOADING TIME AT BAGCLAIM.   00825000
       UNLINK LE  PH6,CTRL1,ALL,3PH,PB*PH4   BAGS TO WAITING PAX             00826000
       LOOP       4PH,BUNL2                                                  00827000
 BUNL3 LOGIC R    PH6                      FREE CHAIN/SWITCH PAIR            00828000
       MSAVEVALUE 1.PH1.1,2,MH             MARK ALL BAGS AT CLAIM AREA       00829000
       TERMINATE                                                             00830000
*                                                                            00831000
*                                                                            00832000
*...T I C K E T I N G   &   C H E C K I N                                    00833000
*                                                                            00834000
 CHEKO TEST E     BV$CHK1B,1,CTRLO         BR IF ENPL, TICKETED AND DIRECT   00835000
       HELPA      FORTM,13.PH2,MH1(PH1,3),PB9,RN4,PB13 LOC,AIRLINE,          00836000
*                                          TICKETED,EXPCHK RN,# PAX          00837000
       ADVANCE    XH$TRVXH                                                   00838000
       TEST E     PB13,0,CHEK9             FT IF GREETER                     00839000
       UNLINK     GRELC,CTRL1,1,10PH,PH10,CHE10 TRY TO UNLINK PAX            00840000
 CHE11 GATE LR    PAS2L                    ONLY ONE AT A TIME                00841000
       SAVEVALUE  PAS22,PB5,XB             NUMBER OF GREETERS                00842000
       SAVEVALUE  PAS23,PB14,XB            PARKING LOT NUMBER                00843000
       LOGIC S    PAS2L                    LET PAX PICK UP INFO              00844000
       TEST NE    PB6,1,*+2                BR IF MODE IS CURB           (*+2)00845000
       TERMINATE                                                             00846000
       ASSIGN     5,0,PB                   NO-ONE IN PARTY                   00847000
       ASSIGN     4,GRT03,PH               ROUTE TO GREETER LEAVING PARKING  00848000
       TRANSFER   ,CTRL1                                                     00849000
*                                                                            00850000
 CHE10 ASSIGN     4,CHE11,PH               WILL GO TO CHE11 WHEN UNLINKED    00851000
       LINK       GRELC,FIFO                                                 00852000
*                                                                            00853000
 CHEK9 TEST E     PB12,3,CHEK6             FT IF TO BE GREETED AT TICKETING  00854000
       UNLINK     GRELC,CTRL1,1,10PH,PH10,CHEK7 TRY TO UNLINK GREETER        00855000
 CHEK8 GATE LS    PAS2L                    WAIT FOR INFO FROM GREETER        00856000
       ASSIGN     5+,XB$PAS22,PB           ADD GREETERS TO PARTY             00857000
       ASSIGN     14,XB$PAS23,PB           PICK UP PARKING LOT NUMBER        00858000
       LOGIC R    PAS2L                    PERMIT NEXT USE OF PAS2 S'VALUES  00859000
       TRANSFER   ,CTRLO                                                     00860000
*                                                                            00861000
 CHEK7 ASSIGN     4,CHEK8,PH               WILL GO TO CHEK8 WHEN UNLINKED    00862000
       LINK       GRELC,FIFO                                                 00863000
```

```
*                                                                         00864000
 CHEK6 TEST NE    PB8,1,CTRLO              BR IF TERMINATING              00865000
       SAVEVALUE  PH2+,PB5,XH              CONGESTION                     00866000
*                                                                         00867000
*                                                                         00868000
*                                                                         00869000
 CHEK1 QUEUE      PH5,PB5                   WAIT FOR FREE SEVER           00870000
       MARK                                MARK PAX FOR WAITING TIME      00870100
       ENTER      PH5                      SERVICE                        00871000
       DEPART     PH5,PB5                                                 00872000
       ASSIGN     11+,M1,PH                ADD WAITING TIME TO PH11       00872100
       TRANSFER   ,PH4                     BR TO CHEK2 OR CHEK3;SET IN FORT.00873000
 CHEK2 ADVANCE    V$CHK1V                  CHECKIN TIME FOR TICKETED PAX  00874000
       LEAVE      PH5                      SERVICE                        00875000
       SAVEVALUE  PH2-,PB5,XH              CONGESTION                     00876000
       TEST E     MH1(PH1,3),1,*+3                                        00876100
       SAVEVALUE  EALXH+,PB13,XH                                          00876200
       TRANSFER   ,CHE12                                                  00876300
       TEST E     MH1(PH1,3),8,*+3                                        00876400
       SAVEVALUE  AMAXH+,PB13,XH                                          00876500
       TRANSFER   ,CHE12                                                  00876600
       TEST E     MH1(PH1,3),6,*+2                                        00876700
       SAVEVALUE  UNTXH+,PB13,XH                                          00876800
 CHE12 TRANSFER   ,CTRLO                                                  00877000
 CHEK3 SAVEVALUE  CHKXH,MH1(PH1,3),XH                                     00878000
       ADVANCE    FN$ATK1F   TICKET/CHECKIN TIME                          00878100
       LEAVE      PH5                                                     00880000
       SAVEVALUE  PH2-,PB5,XH              CONGESTION                     00881000
       MSAVEVALUE 13+,V$TCKTN,1,PB5,MH                                    00881100
       TRANSFER   ,CTRLO                                                  00882000
*                                                                         00882020
*...I M M I G R A T I O N                                                 00882040
*                                                                         00882060
*IMMIO HELPA      FORTM,8,PH2,MH1(PH1,9)      PT. NO.,GATE NO.            00882080
       ADVANCE    XH$TRVXH                                                00882100
       SAVEVALUE  PH2+,PB5,XH              CONGESTION                     00882120
       GATE SNF   PH5,IMMI1                                               00882140
       ENTER      PH5                      SERVICE                        00882160
       TRANSFER   ,IMMI2                                                  00882180
 IMMI1 QUEUE      PH5,PB5                  SERVICE                        00882200
       ENTER      PH5                                                     00882220
       DEPART     PH5,PB5                                                 00882240
 IMMI2 ADVANCE    V$IMM1V                  IMMIGRATION PROCESS TIME       00882260
       LEAVE      PH5                                                     00882280
       SAVEVALUE  PH2-,PB5,XH              CONGESTION                     00882300
       SAVEVALUE  IMIG+,PB13,XH                                           00882310
       TRANSFER   ,CTRLO                                                  00882320
*                                                                         00882400
*...C U S T O M S                                                         00882420
*                                                                         00882440
*CUSTO HELPA      FORTM,4,PH2,PH8          LOC,MH9 ROW                    00882460
       ADVANCE    XH$TRVXH                                                00882480
       SAVEVALUE  PH2+,PB5,XH              CONGESTION                     00882500
 CUST3 GATE SNF   PH5,CUST2                                               00882520
       ENTER      PH5                                                     00882540
       TRANSFER   ,CUST4                                                  00882560
 CUST2 QUEUE      PH5,PB5                                                 00882580
       ENTER      PH5                                                     00882600
       DEPART     PH5,PB5                                                 00882620
 CUST4 ADVANCE    V$CUS1V                  CUSTOMS PROCESS TIME           00882640
       LEAVE      PH5                      SERVICE                        00882660
       SAVEVALUE  PH2-,PB5,XH          CONGESTION                       00882680
       SAVEVALUE  CSTM+,PB13,XH                                         00882690
       TRANSFER   ,CTRLO                                                00882700
*                                                                       00885000
*                                                                       00886000
*...C O N C O U R S E   E X I T  -  T E R M I N A T I N G   P A X       00887000
*                                                                       00888000
 CNCRO HELPA      FORTM,25,PH2,MH1(PH1,9)        LOC, GATE              00889000
       ADVANCE    XH$TRVXH         WALKING TM TO CONCOURSE FROM GATE    00890000
       MSAVEVALUE 11+,PH5,1,PB5,MH      NO OF PAX LEAVING CONCOURSE     00891000
       SAVEVALUE  CONXH+,PB5,XH DUMMY TO KEEP FUNCT 1 FOR MNLINK HAPPY  00892000
       TRANSFER   ,CTRLO                                                00893000
*                                                                       00894000
*                                                                       00895000
*...D E P L A N I N G   C U R B   ( C A R S )                           00896000
*                                                                       00897000
 DCARO ADVANCE    FN$DCA1F             ARRIVAL DISTRIBUTION FOR CARS    00898000
       SAVEVALUE  ARDXH+,1,XH          COUNT OF VEH ON ARRIVING ROAD    00899000
 DCAR1 SAVEVALUE  DEPXH+,1,XH                                           00899100
       HELPA      FORTM,10,MH1(PH1,3),PH1,PB4,0 AIRLINE,FLT,# OF BAGS   00900000
       TEST NE    PB4,0,DCARD          FALL THROUGH IF DEPL             00901000
       ASSIGN     8,V$DBPKD,PH  1 LESS THAN 1ST DEPL CURB STO NO        00901100
       ASSIGN     4,PH8,PH             SAVE CURB STO NO MINUS 1         00901105
       ASSIGN     5,V$DPCD8,PH 1 LESS THN 1ST DBL PRKING STO NO FOR DEPL00901110
       TEST NE    PB10,4,DCARB   IF CIR VEH, TRANSFER TO DCARB          00901120
       ASSIGN     8,V$DEPLS,PH   1 LESS THAN CORRESPONDING STO NO       00901140
       TRANSFER   ,DCARB                                                00901150
*                                      GOING TO ENPLANING SECTION       00901155
 DCARD ASSIGN     8,V$DBPK1,PH   1 LESS THAN ENPL CURB STO NO           00901160
       ASSIGN     4,PH8,PH   SAVE CURB STO NO MINUS 1                   00901163
       ASSIGN     5,V$EPCD8,PH         DBL PARKING STO NO FOR ENPL      00901165
       TEST NE    PB10,4,DCARB   IF CIR VEH, TRANSFER TO DCARB          00901170
       ASSIGN     8,V$ENPLS,PH   1 LESS THAN CORRESPONDING ENPL STO NO  00901200
 DCARB ASSIGN     8+,1,PH              INCREMENT STO NO                 00902020
       TEST NE    PB10,4,*+2       IF CIR VEH, SKIP NEXT TEST           00902030
       TEST NE    PH6,PH8,DCARA  IF STO NO SAME AS DESTINATION          00902040
*                                      STO NO, GO TO DCARA              00902050
       ASSIGN     4+,1,PH              INCREMENT CURB STO NO            00902051
       ASSIGN     5+,1,PH              INCREMENT DBL PARKING STO NO     00902055
       QUEUE      PH4                  ENTER QUEUE FOR BLOCKED LANE     00902057
       TEST E     BV$LNFLD,0  STOP TRAFFIC IF DBL PRKING & NO OF PASSING00902060
*                                      LANES AVAILABLE IS 1             00902070
       DEPART     PH4                  DEPART QUEUE FOR BLOCKED LANE    00902075
       TEST E     S*PH5,0,DCARC    FT IF CURRENT CONTENTS OF DBL PARK   00902080
*                                      STO IS ZERO                      00902090
       SAVEVALUE  NOLAN,V$NLNMX,XH  SET NO OF PASSING LANES TO MAX AVAIL00902100
       SAVEVALUE  TOTL1+,1,XH          SAVE COUNT                       00902110
       TRANSFER   ,*+3                                                  00902120
 DCARC SAVEVALUE  NOLAN,VSNLNM1,XH  SET NO OF PASSING LANES TO 1 LESS   00902130
*                                      THAN MAX AVAILABLE               00902140
       SAVEVALUE  TOTL2+,1,XH          SAVE COUNT                       00902150
       SAVEVALUE  TOTL3+,V$LNDLY,XH    SAVE TIME                        00902160
       ADVANCE    V$LNDLY   DELAY VEH ACCORDING TO NO OF LANES AVAIL    00902170
       TEST NE    PB4,0,*+3            FALL THROUGH IF DEPL             00902175
       TEST E     BV$CIRVD,0,DCARA     IF CIR VEH  IS AT LAST           00902180
*                                      DEPL STO, TRANSFER TO DCARA      00902190
       TRANSFER   ,DCARB               MOVE TO NEXT DEPL STO            00902200
       TEST E     BVSCIRVH,0,DCARA     IF CIR VEH IS AT LAST            00902204
*                                      ENPL STO, TRANSFER TO DCARA      00902206
       TRANSFER   ,DCARB               MOVE TO NEXT ENPL STO            00902208
 DCARA SAVEVALUE  ADPLC+,1,XH          COUNT OF VEH. ON DEPL. CURB      00902210
```

```
       ASSIGN     8,V$TMODV,PH            FOR TIME LIMIT ENFORCEMENT          00902220
       TEST G     PB10,1,DCAR5            BR TO ENTER CURB SLOT               00903000
       TEST G     PB10,2,DCAR4            BR TO ENTER DP SLOT                 00904000
       TEST G     PB10,3,*+3              BR TO ENTER QUEUE SLOT    (*+3)     00905000
       ASSIGN     4,CIRCO,PH              SEND TO RECIRCULATION ROAD          00906000
       TRANSFER   ,CTRL1                                                      00907000
       ENTER      PH6                                                         00908000
       PRIORITY   12                      TO ASSURE PRIORITY FOR DP SLOT      00909000
       ASSIGN     5,RN2,PH                                                    00910000
       LINK       PH6,5PH                 LINK ON QUEUE IN RANDOM ORDER       00911000
*                                                                             00912000
DCAR2  LEAVE      PH6                     LEAVE DP STORAGE                    00913000
       UNLINK     V$DPDQV,DCAR3,1         UNLINK 1 FROM CORR. QUEUE CHAIN     00914000
       ASSIGN     6,V$DPDCV,PH            CHANGE PH6 TO CURB STORE #          00915000
       ASSIGN     10,1,PB                 NOW AT CURB                         00916000
       PRIORITY   10                      DROP TO NORMAL PRIORITY             00917000
       ENTER      PH6                     ENTER CURB STORE                    00918000
       ASSIGN     5,RN2,PH                                                    00919000
       LINK       DPL1C,5PH               RELINK IN RANDOM ORDER              00920000
*                                                                             00921000
DCAR3  TEST E     PB7,0,ENPC8             BR IF VEH UNLINKED IS ENPLANING     00922000
       LEAVE      PH6                     LEAVE QUEUE STORAGE                 00923000
       ASSIGN     6,V$DPQDV,PH            CHANGE PH6 TO DP STORE #            00924000
       ASSIGN     10,2,PB                 NOW USING DP                        00925000
DCAR4  PRIORITY   11                      SET DP PRIORITY                     00926000
DCAR5  ENTER      PH6                     CURB OR DP STORE                    00927000
       UNLINK E   DPL2C,CTRL1,1,10PH,PH10,*+2  MATCH CAR WITH PAX (*+2)00928000
       TRANSFER   ,DPLC9                                                      00929000
       ASSIGN     5,RN2,PH                                                    00930000
       LINK       DPL1C,5PH               LINK IN RANDOM ORDER                00931000
*                                                                             00932000
*...TIME LIMIT ENFORCEMENT                                                    00933000
*                                                                             00934000
       GENERATE   ,,,1,9,2PH              "POLICE OFFICER"                    00935000
DCAR6  ADVANCE    V$DENFV                 FREQUENCY OF ENFORCEMENT            00936000
       UNLINK     DPLIC,DCAR8,ALL,BV$DLIMB UNLINK ANY VEHICLES WHICH          00937000
*                                         HAVE BEEN DOUBLE-PARKED LONGER      00938000
*                                         THAN LIMIT                          00939000
       ASSIGN     1,MH8(12,1),PH          # OF DEPLANING CURBS                00940000
       ASSIGN     1+,MH8(8,1),PH          ADD # OF ENPLANING CURBS.           00941000
       ASSIGN     2,DPQCS                 FIRST QUEUE CHAIN                   00942000
DCAR7  UNLINK     PH2,DCAR9,ALL,BV$DLIMB UNLINK ANY VEHICLES IN QUEUE         00943000
*                                         LONGER THAN LIMIT                   00944000
       ASSIGN     2+,1,PH                                                     00945000
       LOOP       1PH,DCAR7                                                   00946000
       TRANSFER   ,DCAR6                                                      00947000
DCAR8  UNLINK     V$DPDQV,DCAR3,1         UNLINK 1 FROM CORR. QUEUE CHAIN     00948000
DCAR9  LEAVE      PH6                     DP OR QUEUE STORAGE                 00949000
       PRIORITY   10                      DROP TO NORMAL PRIORITY             00950000
       TEST E     PB7,1,*+2               IF ENPLANING, FALL THROUGH          00950500
       ASSIGN     11+,M1,PH               ADD WAITING TIME TO PH11            00950600
       TEST E     PB7,0,CIRC1             BR IF UNLINKED VEH IS ENPLANING     00951000
       TRANSFER   ,CIRCO                  RECIRCULATE                         00952000
*                                                                             00953000
*                                                                             00954000
*...DEPLANING   CURB   (PAX)                                                  00955000
*                                                                             00956000
DPLCO  HELPA      FORTM,9,PH2,PB11,PH7,PH1   LOC,FACTYP,MH9ROW,MH1ROW         00957000
       ADVANCE    XH$TRVXH                                                    00958000
*      SAVEVALUE  PH2+,PB5,XH             CONGESTION                          00959000
       TRANSFER   FN,DPL1F                BRANCH BY TRANSP MODE               00960000
```

```
*                                                                       00961000
*   PRIVATE CAR                                                         00962000
*                                                                       00963000
 DPLC3 UNLINK E  DPL1C,DPLC9,1,10PH,PH10,*+2  MATCH PAX WITH CAR (*+2) 00964000
       TRANSFER  ,DPLC1                BR IF MATCH SUCCESSFUL          00965000
       UNLINK E  DPL3C,DPCG2,1,10PH,PH10,DPLC2 MATCH WITH GREETER      00966000
       TRANSFER  ,DPCG1                BR IF MATCH SUCCESSFUL          00967000
 DPLC1 GATE LS   PASSL                 WAIT FOR CAR LOCATION INFO      00968000
       ASSIGN    6,XH$PASS1,PH         PICK UP...                      00969000
       ASSIGN    10,XB$PASS2,PB        CAR LOCATION INFO               00970000
       LOGIC R   PASSL                                                 00971000
       ADVANCE   V$DPL1V               LOADING TIME                    00972000
       TEST E    PB10,1,DPLC7          FT IF LEAVING CURB              00973000
       SAVEVALUE TMPXF,V$DPCDV,XF      INDIRECT ADDR. USING VARIABLE   00974000
       MARK                            MARK PAX FOR WAITING TIME       00974100
       ADVANCE   V$DPL2V               RANDOM PULLOUT DELAY            00975000
       ASSIGN    11+,M1,PH             ADD WAITING TIME TO PH11        00975100
       LEAVE     PH6                                                   00976000
       UNLINK E  DPL1C,DCAR2,1,6PH,V$DPCDV UNLINK 1 FROM CHAIN WITH    00977000
*                                      PH6=CORR. DP STORE              00978000
       TRANSFER  ,DPLC8                                                00979000
*                                                                      00980000
 DPLC7 LEAVE     PH6                                                   00981000
       UNLINK    V$DPDQV,DCAR3,1       UNLINK 1 FROM CORR. QUEUE CHAIN 00982000
*      SAVEVALUE PH2-,PB5,XH           CONGESTION                      00983000
 DPLC8 SAVEVALUE DRDXH+,1,XH           COUNT OF VEH ON DEPARTING ROAD  00984000
       TRANSFER  ,CTRL0                                                00985000
*                                                                      00986000
 DPLC2 ASSIGN    4,DPLC1,PH                                            00987000
       LINK      DPL2C,FIFO            PAX WAITING FOR CARS            00988000
*                                                                      00989000
 DPLC9 GATE LR   PASSL                 ONLY ONE CAR AT A TIME          00990000
       SAVEVALUE PASS1,PH6,XH          PASS CAR LOCATION...            00991000
       SAVEVALUE PASS2,PB10,XB         INFO TO PASSENGER XAC           00992000
       LOGIC S   PASSL                                                 00993000
       TERMINATE                                                       00994000
*                                                                      00995000
*   BUS/LIMO - PAX                                                     00996000
*                                                                      00997000
 DPLC4 SAVEVALUE DPCXH+,1,XH           DEPLANING PAX WAITING ON BUS/LIMO00998000
*      SAVEVALUE PH2+,PB5,XH           CONGESTION                      00999000
       TRANSFER  ,CTRL0                                                01000000
*                                                                      01001000
*   TAXI                                                               01002000
*                                                                      01003000
 DPLC5 ADVANCE   V$DPL1V               LOADING TIME                    01004000
*      SAVEVALUE PH2-,PB5,XH           CONGESTION                      01005000
       SAVEVALUE DRDXH+,1,XH           COUNT OF VEH ON DEPARTING ROAD  01006000
       TRANSFER  ,CTRL0                                                01007000
*                                                                      01008000
*   BUS/LIMO - VEHICLE                                                 01009000
*                                                                      01010000
 DPLC6 SAVEVALUE ARDXH+,1,XH           COUNT OF VEH ON ARRIVING ROAD   01011000
*      SAVEVALUE PH2-,PB5,XH           CONGESTION                      01012000
       SAVEVALUE DPCXH,0,XH            REMOVE PAX WAITING FOR BUS/LIMO 01013000
       SAVEVALUE DRDXH+,1,XH           COUNT OF VEH ON DEPARTING ROAD  01014000
       TERMINATE                                                       01015000
*                                                                      01016000
*   DEPLANING CURB GREETERS (AFTER RECIRCULATE AND PARK)               01017000
*                                                                      01018000
 DPCG0 HELPA     FORTM,10,MH1(PH1,3),PH1,PB4,1 A/L,FLT,# OF BAGS,GREET 01019000
```

```
       ADVANCE     XH$TRVXH                                             01020000
       SAVEVALUE   PH2+,PB5,XH           CONGESTION                     01021000
       UNLINK E    DPL2C,DPCG1,1,10PH,PH10,*+2 MATCH PAX AT CURB        01022000
       TRANSFER    ,DPCG2                BR IF SUCCESSFUL               01023000
       LINK        DPL3C,FIFO            OTHERWISE LINK TO SPECIAL CHAIN 01024000
*                                                                       01025000
 DPCG1 GATE LS     PAS5L                 WAIT FOR INFO FROM GREETER     01026000
       ASSIGN      5+,XB$PAS52,PB        ADD GREETERS TO PARTY          01027000
       ASSIGN      14,XB$PAS53,PB        PICK UP PARKING LOT NUMBER     01028000
       LOGIC R     PAS5L                 PERMIT NEXT USE OF PAS5 S'VALUES 01029000
       ASSIGN      6,2,PB                CHANGE MODE TO PARKING         01030000
       ASSIGN      2,1,PB                PROC. FN. POINTER              01031000
       ASSIGN      1,GRCPF,PB            NEW PROC. FN.                  01032000
       TRANSFER    ,CTRLO                                               01033000
*                                                                       01034000
 DPCG2 GATE LR     PAS5L                 ONLY ONE AT A TIME             01035000
       SAVEVALUE   PAS52,PB5,XB          NUMBER OF GREETERS             01036000
       SAVEVALUE   PAS53,PB14,XB         PARKING LOT NUMBER             01037000
       LOGIC S     PAS5L                 LET PAX PICK UP INFO           01038000
       TERMINATE                                                        01039000
*                                                                       01040000
*                                                                       01041000
*...E N P L A N I N G   C U R B                                         01042000
*                                                                       01043000
*   PVTCAR/TAXI                                                         01044000
*                                                                       01045000
 ENPCO SAVEVALUE   ARDXH+,1,XH           COUNT OF VEH ON ARRIVING ROAD  01046000
 ENPCR SAVEVALUE   ENPXH+,1,XH                                          01046100
       HELPA       FORTM,11,MH1(PH1,3),PB6  AIRLINE,MODE                01047000
       ASSIGN      8,V$DBPK1,PH   1 LESS THAN 1ST ENPL CURB STO NO      01047003
       ASSIGN      4,PH8,PH              SAVE CURB STO NO MINUS 1       01047004
       ASSIGN      5,V$EPCD8,PH 1 LESS THN 1ST DBL PRKING STO NO FOR ENPL01047006
       TEST NE     PB10,4,ENPCA          IF CIR VEH, TRANSFER TO ENPCA  01047009
       ASSIGN      8,V$ENPLS,PH 1 LESS THN CORR 1ST ENPL STO NO         01047012
 ENPCA ASSIGN      8+,1,PH               INCREMENT ENPL STO NO          01047020
       TEST NE     PB10,4,*+2   IF CIR VEH SKIP NEXT TEST               01047030
       TEST NE     PH6,PH8,ENPC1 IS ENPL STO NO SAME AS DESTINATION STO N01047040
       ASSIGN      4+,1,PH               INCREMENT ENPL CURB STO NO     01047043
       ASSIGN      5+,1,PH               INCREMENT ENPL DBL PRKING STO NO 01047045
       QUEUE       PH4                   ENTER QUEUE FOR BLOCKED LANE   01047047
       MARK                              MARK PAX FOR WAITING TIME      01047048
       TEST E      BV$LNFUL,0  STOP TRAFFIC IF DBL PRKING & NO OF PASSING01047050
       ASSIGN      11+,M1,PH             ADD WAITING TIME TO PH11       01047055
*                                        LANES AVAILABLE IS 1           01047060
       DEPART      PH4                   LEAVE QUEUE FOR BLOCKED LANE   01047065
       TEST E      S*PH5,0,ENPCB   IF CURRENT CONTENTS OF DBL PARKING STO01047070
*                                        IS ZERO FALL THROUGH           01047080
       SAVEVALUE   NOLAN,V$NLNMX,XH SET NO OF PASSING LANES TO MAX AVAIL 01047090
       SAVEVALUE   TOTL4+,1,XH           SAVE COUNT                     01047100
       TRANSFER    ,*+3                                                 01047110
 ENPCB SAVEVALUE   NOLAN,V$NLNM1,XH  SET NO OF PASSING LANES TO 1 LESS  01047120
*                                        THAN MAX AVAILABLE             01047130
       SAVEVALUE   TOTL5+,1,XH           SAVE COUNT                     01047140
       SAVEVALUE   TOTL6+,V$LNDLY,XH     SAVE TIME                      01047150
       MARK                              MARK PAX FOR WAITING TIME      01047155
       ADVANCE     V$LNDLY  DELAY VEH ACCORDING TO NO OF LANES AVAILABLE 01047160
       ASSIGN      11+,M1,PH             ADD WAITING TIME TO PH11       01047165
       TEST E      BV$CIRVH,0,ENPC1  IF CIR VEH IS AT LAST ENPL CUR STO 01047170
*                                        TRANSFER TO ENPC1              01047180
       TRANSFER    ,ENPCA                MOVE TO NEXT ENPL STO NO       01047190
 ENPC1 SAVEVALUE   AENPC+,1,XH           COUNT OF VEH ON ENPL CURB      01048000
```

```
       ASSIGN     8,VSTMODV,PH           FOR TIME LIMIT ENFORCEMENT          01049000
       TEST G     PB10,1,ENPC5           BR TO ENTER CURB SLOT               01050000
       TEST G     PB10,2,ENPC4           BR TO ENTER DP SLOT                 01051000
       TEST G     PB10,3,*+3             BR TO ENTER QUEUE SLOT              01052000
       ASSIGN     4,CIRC1,PH             SEND TO RECIRCULATION ROAD          01053000
       TRANSFER   ,CTRL1                                                     01054000
       ENTER      PH6                                                        01055000
       PRIORITY   12                     TO ASSURE PRIORITY FOR DP SLOT      01056000
       ASSIGN     5,RN2,PH                                                   01057000
       MARK                              MARK PAX FOR WAITING TIME           01057100
       LINK       PH6,5PH                LINK ON QUEUE IN RANDOM ORDER       01058000
*                                                                            01059000
 ENPC8 TEST E     PB7,1,DCAR3            BR IF VEH UNLINKED IS DEPLANING     01059100
       ASSIGN     11+,M1,PH              ADD WAITING TIME TO PH11            01060000
       LEAVE      PH6                    LEAVE QUEUE STORAGE                 01061000
       ASSIGN     6,V$EPQCV,PH           CHANGE PH6 TO CURB STORE #          01062000
       ASSIGN     10,1,PB                USING CURB                          01063000
       TEST E     R*PH6,0,ENPC5          FT IF STORAGE FILLED                01064000
       ASSIGN     6,V$DPCDV,PH           CHANGE PH6 TO DP STORE              01065000
       ASSIGN     10,2,PB                USING DP                            01066000
 ENPC4 PRIORITY   11                     DP PRIORITY                         01067000
 ENPC5 ENTER      PH6                                                        01068000
       ADVANCE    V$ENP1V                ENPL CURB UNLOAD TIME--CAR/TAXI     01069000
       SPLIT      1,*+5                                                      01069100
       TEST E     PB9,0,*+3              FT IF PRETICKETED                   01069200
       TRANSFER   .XH$CRBXH,*+2,*+1      FT IF USING CURB SIDE CHECK IN      01069300
       ADVANCE    FN$CSCKF               CURB SIDE CHECK IN TIME             01069400
       TRANSFER   ,CTRL0                                                     01069450
       ADVANCE    FN$ENP3F               EMPTY CAR PARKING TIME              01069500
       TEST E     PB10,1,ENPC7           FT IF LEAVING CURB                  01070000
       SAVEVALUE  TMPXF,VSDPCDV,XF       INDIRECT ADDR. USING VARIABLE       01071000
       ADVANCE    V$DPL2V                RANDOM PULLOUT DELAY                01072000
       LEAVE      PH6                    CURB STORE                          01073000
       UNLINK     V$EPCQV,ENPC8,1        UNLINK 1 FROM QUEUE                 01074000
       TRANSFER   ,ENPC9                                                     01075000
 ENPC7 LEAVE      PH5                    DP STORE                            01076000
       UNLINK     V$DPDQV,ENPC8,1        UNLINK 1 FROM QUEUE                 01077000
 ENPC9 SAVEVALUE  DRDXH+,1,XH            COUNT OF VEH ON DEPARTING ROAD      01078000
       TEST E     PB5.PB13,*+2           FT IF NO WELL-WISHERS       (*+2)   01079000
       TERMINATE                                                             01080000
       HELPA      FORTM,16,PH2,PB6,1,0,1 LOC,MODE,ENPL,0,LOTNO ONLY          01081000
       ADVANCE    300                    VEHICLE TO LOT                      01083000
       SAVEVALUE  RERXH+,1,XH            RECIRC. RDWY COUNT                  01083100
       SAVEVALUE  PKLXH+,1,XH            ENTER CAR IN PARKING LOT            01084000
       SAVEVALUE  PLIXH+,1,XH            VEH COUNT IN PARKING LOT            01085000
       TERMINATE                                                             01086000
*                                                                            01087000
*   BUS/LIMO                                                                 01088000
*                                                                            01089000
 ENPC2 HELPA      FORTM,11,MH1(PH1,3),PB6   AIRLINE,MODE                     01090000
       ASSIGN     4,ENPC3,PH                                                 01091000
       LINK       EBUSC,FIFO             LINK TO ENPLANING BUS USER CHAIN    01092000
 ENPC3 TRANSFER   ,CTRL0                                                     01093000
*                                                                            01094000
*   BUS/LIMO (VEHICLE)                                                       01095000
*                                                                            01096000
 ENPC6 SAVEVALUE  ARDXH+,1,XH            COUNT OF VEH ON ARRIVING ROAD       01097000
       ADVANCE    V$ENP2V                BUS/LIMO LOAD TIME                  01098000
       UNLINK     EBUSC,CTRL1,ALL                                            01099000
       SAVEVALUE  DRDXH+,1,XH            COUNT OF VEH ON DEPARTING ROAD      01100000
       TERMINATE                                                             01101000
```

```
*                                                                    01102000
*...RECIRCULATION ROAD                                               01103000
*                                                                    01104000
 CIRC0 ADVANCE    V$CIRCV               RECIRCULATE                  01105000
       SAVEVALUE  RCDXH+,1,XH           RECIRCULATION TO DEPLANING   01106000
       SAVEVALUE  RERXH+,1,XH           RECIRC. ROADWAY COUNT        01106100
       TEST E     PB12,0,DCAR1          BR IF GREETER WHO HAS ALREADY MET01107000
       TRANSFER   .XH$CPKXH,DCAR1,*+1   FT TO ENTER PK OR BR TO DEPL ROAD01108000
       ASSIGN     6,2,PB                CHANGE MODE TO PARKING       01109000
       TEST E     PB2,4,*+2                                          01109100
       TERMINATE                                                     01109200
       ASSIGN     1,GRECF,PB            PROC. FN. TO CURB            01110000
       TRANSFER   ,CTRL0                                             01111000
 CIRC1 ADVANCE    V$CIRCV               RECIRCULATE                  01112000
       SAVEVALUE  RERXH+,1,XH           RECIRC. ROADWAY COUNT        01112100
       SAVEVALUE  RCEXH+,1,XH           RECIRCULATION TO ENPLANING   01113000
       TRANSFER   ,ENPCR                RETURN TO ENPLANING ROAD     01114000
*                                                                    01115000
*                                                                    01116000
*...E N T R A N C E                                                  01117000
*                                                                    01118000
 ENTR0 HELPA      FORTM,12,PH2          LOC                          01119000
       ADVANCE    XH$TRVXH                                           01120000
       TEST E     DPDIN,1,*+3                                        01120100
       SAVEVALUE  DPLIN+,PB5,XH                                      01120200
       TRANSFER   ,*+2                                               01120300
       SAVEVALUE  ENDIN+,PB5,XH                                      01120400
       SAVEVALUE  ENDOR+,PB5,XH                                      01120500
       TRANSFER   ,CTRL0                                             01121000
*                                                                    01122000
*                                                                    01123000
*...E X I T                                                          01124000
*                                                                    01125000
 EXIT0 HELPA      FORTM,7,PH2,PB11,FN*PB1,PH7   LOC,CUR PROC,NXTADR,MH9R01126000
       ADVANCE    XH$TRVXH              TRAVEL TIME TO EXIT          01127000
       TEST E     DDOUT,1,*+3                                        01127100
       SAVEVALUE  DPOUT+,PB5,XH                                      01127200
       TRANSFER   ,*+2                                               01127300
       SAVEVALUE  EPOUT+,PB5,XH                                      01127400
       SAVEVALUE  EXDOR+,PB5,XH                                      01127500
       TRANSFER   ,CTRL0                                             01128000
*                                                                    01129000
*                                                                    01130000
*...G A T E   ( E N P L A N I N G   P A X )                          01131000
*                                                                    01132000
 GATE0 HELPA      FORTM,15,PH2,MH1(PH1,9)   LOC,GATE                 01133000
       ADVANCE    XH$TRVXH                                           01134000
       TEST E     PB13,0,GATE7          FT IF NO PAX                 01135000
       UNLINK     GREGC,CTRL1,1,10PH,PH10,GAT10 MATCH WITH PAX       01136000
 GATE8 GATE LR    PAS3L                 ONLY ONE AT A TIME           01137000
       SAVEVALUE  PAS32,PB5,XB          NUMBER OF GREETERS           01138000
       SAVEVALUE  PAS33,PB14,XB         PARKING LOT NUMBER           01139000
       LOGIC S    PAS3L                 LET PAX PICK UP INFO         01140000
       TEST NE    PB6,1,*+2             BR IF MODE IS CURB      (*+2)01141000
       TERMINATE                                                     01142000
       ASSIGN     5,0,PB                NO-ONE IN PARTY              01143000
       ASSIGN     4,GRT03,PH            ROUTE TO GREETER LEAVING PARKING 01144000
       TRANSFER   ,CTRL1                                             01145000
 GAT10 ASSIGN     4,GATE8,PH            WILL GO TO GATE8 WHEN UNLINKED  01146000
       LINK       GREGC,FIFO                                         01147000
 GATE7 TEST NE    PB5,PB13,GATE3        FT IF HAVE WELL-WISHERS      01148000
```

```
      SPLIT      1,*+3               SPLIT WW OFF             (*+3)  01149000
      ASSIGN     5,PB13,PB           DROP WW                         01150000
      TRANSFER   ,GATE3                                              01151000
      ASSIGN     5-,PB13,PB          SUBTRACT OUT PAX                01152000
      ASSIGN     13,0,PB             SET PAX TO 0                    01153000
      ASSIGN     1,WWG1F,PB          WW LEAVING FROM GATE PROC FN    01154000
      ASSIGN     2,1,PB              PROC FN POINTER                 01155000
      TRANSFER   ,CTRLO                                              01156000
GATE3 SAVEVALUE  PH2+,PB5,XH         CONGESTION                      01157000
GATE1 QUEUE      PH5,PB5                                             01161000
      SAVEVALUE  GATXH+,PB13,XH      PASS. COUNT AT THE GATE         01161100
      MARK                           MARK PAX FOR WAITING TIME       01161110
      ENTER      PH5                 SERVICE                         01162000
      DEPART     PH5,PB5                                             01163000
      ASSIGN     11+,M1,PH           ADD WAITING TIME TO PH11        01163100
      ADVANCE    V$GAT3V             GATE SERVICE TIME               01165000
GATE6 LEAVE      PH5                                                 01166000
      GATE LS    PH5,GATE2           GATE SET WHEN BOARDING BEGINS   01167000
      SAVEVALUE  PH2-,PB5,XH         CONGESTION                      01168000
      TRANSFER   ,CTRLO                                              01169000
GATE2 MSAVEVALUE 1+,PH1,12,PB5,MH    PAX WAITING AT LOUNGE TO BOARD  01172000
      TRANSFER   ,CTRLO                                              01173000
*                                                                    01174000
*   START BOARDING OPERATIONS                                        01175000
*                                                                    01176000
GATE9 ADVANCE    V$GAT1V             ADVANCE TO BOARDING TIME        01177000
      ASSIGN     5,V$GAT2V,PH                                        01178000
      LOGIC S    PH5                 GATE BOARDING SWITCH            01179000
      ADVANCE    XH$BDTXH                                            01180000
      LOGIC R    PH5                 GATE BOARDING SWITCH            01181000
      ASSIGN     5,MH1(PH1,9),PH     GATE NO.                        01182000
      ASSIGN     5.MH9(PH5,3),PH     POINT NO                        01183000
      SAVEVALUE  PH5-,MH1(PH1,12),XH REMOVE PAX BOARDING FROM POINT  01184000
      MSAVEVALUE 1,PH1;12,0,MH       RESET TO ZERO FOR LATE PAX COUNT 01184100
      TERMINATE                                                      01185000
*                                                                    01186000
*                                                                    01187000
*...G R O U N D   T R A N S P O R T A T I O N   ( M I S C . )        01188000
*                                                                    01189000
*                GRT00 - DEPLANING PAX - SELF                        01190000
*                GRT01 - ENPLANING PAX - SELF                        01191000
*                                                                    01192000
*   DEPLANING PAX - SELF                                             01193000
*                                                                    01194000
GRT00 TEST E     MH1(PH1,3),10,*+2   TEST FOR SHUTTLE PAX (*+2)      01195000
      ASSIGN     14,3,PB             ASSIGN TO PARK. FACIL 3         01195100
      HELPA      FORTM,10,PH2,PB6,0  LOC,MODE,DEP                    01195200
      ADVANCE    XH$TRVXH                                            01196000
      TRANSFER   ,CTRL3                                              01197000
*                                                                    01198000
*   ENPLANING PAX - SELF                                             01199000
*                                                                    01200000
GRT01 TEST E     MH1(PH1,3),10,*+2   TEST FOR SHUTTLE PAX (*+2)      01201000
      ASSIGN     14,3,PB             ASSIGN TO PARK. FACIL 3         01201100
      HELPA      FORTM,16,0,PB6,1    *SPARE*,MODE,ENP                01201200
      SAVEVALUE  ARDXH+,1,XH         COUNT OF VEH ON ARRIVING ROAD   01202000
      SAVEVALUE  PKLXH+,1,XH         ENTER CAR IN PARKING LOT        01203000
      SAVEVALUE  PLIXH+,1,XH         VEH. COUNT IN PARK. LOT         01204000
      TRANSFER   ,CTRLO                                              01205000
*                                                                    01206000
*   GREETERS                                                         01207000
*                                                                          01208000
 GRT02 TEST E      MH1(PH1,3),10,*+2      TEST FOR SHUTTLE PAX (*+2)      01209000
       ASSIGN      14,3,PB                ASSIGN TO PARK FACIL 3          01209100
       HELPA       FORTM,16,0,PB6,0       LOC,MODE,DEPL                   01209200
       SAVEVALUE   ARDXH+,1,XH           COUNT OF VEH ON ARRIVING ROAD    01210000
       SAVEVALUE   PKLXH+,1,XH           ENTER CAR IN PARKING LOT         01211000
       SAVEVALUE   PLIXH+,1,XH           VEH COUNT IN PARKING LOT         01212000
       TRANSFER    ,CTRL0                                                 01213000
*                                                                          01214000
*   GREETERS TAKING CAR FROM PARKING TO CURB                              01215000
*                                                                          01216000
 GRT03 HELPA       FORTM,16,PH2,PB6,0    LOC,MODE,DEPL                    01217000
       ADVANCE     XH$TRVXH                                               01218000
       QUEUE       PH5                                                    01219000
       ENTER       PH5                                                    01220000
       DEPART      PH5                                                    01221000
       ADVANCE     V$PAR1V                                                01222000
       LEAVE       PH5                                                    01223000
       SAVEVALUE   PKLXH-,1,XH           EXIT CAR FROM PARKING LOT        01224000
       SAVEVALUE   RERXH+,1,XH           RECIRC. ROADWAY COUNT            01224100
       SAVEVALUE   PKCXH+,1,XH           PARKING TO CURB RECIRCULATION    01225000
       ASSIGN      4,DCAR1,PH            ROUTE TO DEPLANING CURB          01226000
       TRANSFER    ,CTRL1                                                 01227000
*                                                                          01228000
*                                                                          01229000
*...PARKING  (DEPL  PAX - CARS)                                           01230000
*                                                                          01231000
 PARK0 QUEUE       PH5                   SERVICE                          01232000
       MARK                              MARK PAX FOR WAITING TIME        01232100
       ENTER       PH5                                                    01233000
       DEPART      PH5                                                    01234000
       ASSIGN      11+,M1,PH             ADD WAITING TIME TO PH11         01234100
       ADVANCE     V$PAR1V               PARKING EXIT SERVICE TIME        01235000
       LEAVE       PH5                                                    01236000
       SAVEVALUE   PKLXH-,1,XH           EXIT CAR FROM PARKING LOT        01237000
       SAVEVALUE   DRDXH+,1,XH           COUNT OF VEH ON DEPARTING ROAD   01238000
       TRANSFER    ,CTRL0                                                 01239000
*                                                                          01240000
*                                                                          01241000
*...RENTACAR                                                              01242000
*                                                                          01243000
*   FOLLOWING FOR DEPL PAX RENTING A CAR                                  01244000
*                                                                          01245000
 RCAR0 TEST E      PB6,3,CTRL0           RETURN TO CONTROL IF NOT RENTACAR01246000
       ASSIGN      10,FN$RCA2F,PB        CAR RENTAL AGENCY SELECT         01247000
       HELPA       FORTM,6,PH2,PB10      LOC,AGENCY                       01248000
       ADVANCE     XH$TRVXH              TRAVEL TIME TO CAR RENTAL AREA   01249000
       SAVEVALUE   PH2+,PB5,XH           CONGESTION                       01250000
 RCAR1 QUEUE       PH5,PB5                WAIT FOR FREE AGENT             01254000
       MARK                              MARK PAX FOR WAITING TIME        01254100
       ENTER       PH5                   SERVICE                          01255000
       DEPART      PH5,PB5                                                01256000
       ASSIGN      11+,M1,PH             ADD WAITING TIME TO PH11         01256100
 RCAR2 ADVANCE     V$RCA1V               CAR RENTAL PROCESSING TIME       01257000
       LEAVE       PH5                   SERVICE                          01258000
       SAVEVALUE   PH2-,PB5,XH           CONGESTION                       01259000
       TRANSFER    ,CTRL0                                                 01260000
*                                                                          01261000
*   DEPL PAX IN GROUND TRANSPORT WHO HAVE ALREADY RENTED CAR.             01262000
*   NOTE: CURRENT LOGIC ASSUMES PAX PICKS UP CAR AT AGENCY PARKING LOT.   01263000
*                                                                          01264000
```

```
 RCAR9 HELPA       FORTM,16,PH2,PB6,0,PB10   LOC,MODE,DEP/ENP,AGENCY      01265000
       ADVANCE     XHSTRVXH                                               01266000
       SAVEVALUE   DRDXH+,1,XH          COUNT OF CARS ON DEPARTING ROAD   01267000
       TRANSFER    ,CTRLO                                                 01268000
*                                                                         01269000
*   ENPLANING PAX - RENTACAR                                              01270000
*                                                                         01271000
*             NOTE: CURRENT LOGIC ASSUMES RENTAL CAR RETURNED TO A        01272000
*                   PARKING LOT (GENERAL OR AGENCY LOT).   PROCESSING,    01273000
*                   IF ANY, IS DONE IN THE TERMINAL.                      01274000
*                                                                         01275000
 RCAR8 HELPA       FORTM,16,0,PB6,1,PB10    ***,MODE.DEP/ENP,AGENCY       01276000
       SAVEVALUE   ARDXH+,1,XH          COUNT OF VEH ON ARRIVING ROAD     01277000
       TRANSFER    ,CTRLO                                                 01278000
*                                                                         01279000
*                                                                         01280000
*...S E C U R I T Y                                                       01281000
*                                                                         01282000
 SECU0 HELPA       FORTM,14,PH2,MH1(PH1,9)         LOC, GATE              01283000
       ADVANCE     XHSTRVXH                                               01284000
       TEST NE     PB5,PB13,SECU3       FT IF HAVE WW                     01285000
       TRANSFER    .XHSWWCXH,*+1,SECU3  FT IF LEAVING WW AT SECURITY      01286000
       SPLIT       1,*+3                SPLIT WW OFF               (*+3)  01287000
       ASSIGN      5,PB13,PB            DROP WW                           01288000
       TRANSFER    ,SECU3                                                 01289000
       ASSIGN      5-,PB13,PB           SUBTRACT OUT PAX                  01290000
       ASSIGN      13,0,PB              SET PAX TO 0                      01291000
       ASSIGN      1,WWS1F,PB           WW LEAVING FROM SEC PROC FN       01292000
       ASSIGN      2,1,PB               PROC FN POINTER                   01293000
       TRANSFER    ,CTRLO                                                 01294000
 SECU3 SAVEVALUE   PH2+,PB5,XH          CONGESTION                        01295000
 SECU1 QUEUE       PH5,PB5                                                01299000
       MARK                             MARK PAX FOR WAITING TIME         01299050
       ENTER       PH5                  SERVICE                           01300000
       DEPART      PH5,PB5                                                01301000
       ASSIGN      11+,M1,PH            ADD WAITING TIME TO PH11          01302000
 SECU2 ADVANCE     V$SEC1V              SECURITY CHECK TIME               01303000
       LEAVE       PH5                  SERVICE                           01304000
       SAVEVALUE   PH2-,PB5,XH          CONGESTION                        01305000
       MSAVEVALUE  12+,V$SECNN,1,PB5,MH  NO. OF PAX LEAVING SECURITY      01305100
       TRANSFER    ,CTRLO                                                 01306000
*                                                                         01307000
*                                                                         01308000
*...C O N C E S S I O N S                                                 01309000
*                                                                         01310000
 LOBC0 HELPA       FORTM,24,PH2,PH1,RN5,C1,1 LOC,FLT,RN,CLOCK,SWITCH      01311000
       ADVANCE     XHSTRVXH                                               01312000
       SAVEVALUE   PH2+,PB5,XH          CONGESTION                        01313000
       ADVANCE     PH5                  WASTE TIME AT CONCESSION          01314000
       SAVEVALUE   PH2-,PB5,XH          CONGESTION                        01315000
       TRANSFER    ,CTRLO                                                 01316000
 CONC0 HELPA       FORTM,24,PH2,PH1,RN5,C1,2 LOC,FLT,RN,CLOCK,SWITCH      01317000
       ADVANCE     XHSTRVXH                                               01318000
       SAVEVALUE   PH2+,PB5,XH          CONGESTION                        01319000
       ADVANCE     PH5                  WASTE TIME AT CONCESSION          01320000
       SAVEVALUE   PH2-,PB5,XH          CONGESTION                        01321000
       TRANSFER    ,CTRLO                                                 01322000
*                                                                         01323000
*                                                                         01324000
*...T R A N S F E R   F L I G H T S                                       01325000
*                                                                         01326000
```

```
XFLT0 ASSIGN     1,TDP1F,PB           ASSIGN PROCESS FN
      ASSIGN     2,2,PB      SKIP CONCOURSE FOR TDP1F FUNCTION
      ASSIGN     13,1,PB              PAX IN PARTY
      ASSIGN     5,1,PB               TOTAL IN PARTY
      HELPA      FORTM,18,0,0         INITIALIZE XFER FLT TABLE MH5
      TEST L     MH1(PH1,1),0,*+3     FT IF AT END OF FLT TABLE
      TEST E     MH5(1,1),0,XFLT3     FT IF NO XFER FLTS
      TERMINATE
      SPLIT      1,XFLT3
*
*   ADD FLT TO XFR FLT TABLE
*
      TEST G     MH1(PH1,11),0,XFLT2  BR IF NO TPAX FOR NEXT FLT
XFLT1 ADVANCE    V$XFL1V
      HELPA      FORTM,18,PH1,2
XFLT2 ASSIGN     1+,1,PH
      TEST L     MH1(PH1,1),0,XFLT5   TEST FOR END OF TABLE
      TERMINATE
XFLT5 TEST E     BV$XFL1B,1,XFLT2     BR IF ARRV FLT OR NO TPAX
      TRANSFER   ,XFLT1
*
*   DELETE FLT FROM XFR FLT TABLE
*
XFLT3 TEST G     MH5(1,1),0,XFLT9     BR IF XFER FLT TABLE EMPTY
      ASSIGN     1,MH5(1,1),PH        FIRST FLT TO DELETE
XFLT7 ADVANCE    V$XFL2V
      ASSIGN     5,MH1(PH1,9),PH      GATE NO
      ASSIGN     3,MH1(PH1,7),PB      TYPE FLT (DOM,COM,INT)
      HELPA      FORTM,18,PH1,3       MH1 ROW NO., FLAG FOR TICK CNTER
      SPLIT      MH1(PH1,11),XFLT8    CREATE REMAINING XFER PAX
      HELPA      FORTM,18,PH1,1       DELETE FLT FROM TABLE
XFLT4 ASSIGN     1+,1,PH
XFLT9 TEST L     MH1(PH1,1),0,XFLT6   TEST FOR END OF TABLE
      TERMINATE
XFLT6 TEST E     BV$XFL1B,1,XFLT4     BR IF ARRV FLT OR NO TPAX
      TRANSFER   ,XFLT7
XFLT8 ADVANCE    V$XFL3V              STAGGER PAX TO SECURITY
      TRANSFER   ,CTRL0
*
*
*
*...E R R O R   C H A I N
*
*                COLLECTS XACS CAUSING SELECTED ERRORS.   PH4 WILL
*                NORMALLY GIVE CLUE TO NATURE OF PROBLEM.
*
ERROR LINK       ERRCH,FIFO
*
*
*   WHEN REMOVING ANY MODULE, UNCOMMENT ENTRY POINT CARD & OTHERS NOTED:
*   NOTE: TO REMOVE "CHECKIN", UNCOMMENT CHEK0, CHEK1, AND CHEK2.
*         TO REMOVE "DEPLCURB", UNCOMMENT DPCL0 AND DPCL9.
*         TO REMOVE "GATE (ENPLANING PAX)", UNCOMMENT GATE AND GATE9.
*         TO REMOVE "RENTACAR", UNCOMMENT RCAR0, RCAR8 AND RCAR9.
*
*CHEK0 TRANSFER  ,CTRL0
*CHEK1 TRANSFER  ,CTRL0
*CHEK2 TRANSFER  ,CTRL0
 CUST0 TRANSFER  ,CTRL0
*DCAR0 TRANSFER  ,CTRL0
*DPLC0 TRANSFER  ,CTRL0
```

```
*DPLC9 TRANSFER    ,CTRLO
*ENPCO TRANSFER    ,CTRLO
*EXITO TRANSFER    ,CTRLO
*GATEO TRANSFER    ,CTRLO
*GATE3 TRANSFER    ,CTRLO
*GATE9 TERMINATE
 IMMIO TRANSFER    ,CTRLO
*PARKO TRANSFER    ,CTRLO
*RCARO TRANSFER    ,CTRLO
*RCAR8 TRANSFER    ,CTRLO
*RCAR9 TRANSFER    ,CTRLO
*SECUO TRANSFER    ,CTRLO
*
*
*...CHANGE   CARD   READER
*
       GENERATE    ,,,1,120,2PH
 CHG00 HELPA       FORTM,23,1            PERFORM PREVIOUS AND READ NEXT
*                                        CHANGE CARD
       TEST G      XH$NSCXH,0,*+3        IF NO STORAGE CHANGES,     (*+3)
*                                        DON'T SPLIT
       ASSIGN      1-,1,PH
       SPLIT       XH$NSCXH,CHG01,1PH    SPLIT 1 XAC TO CHANGE EACH
*                                        STORAGE, SEQUENCED IN PH1
       ADVANCE     XF$CHGXF              WAIT TILL NEXT CHANGE
       TEST E      XH$NSCXH,0            IF WAIT WAS 0, WAIT TILL MH7 USED
       TRANSFER    ,CHG00
 CHG01 ASSIGN      2,PH1,PH
       SAVEVALUE   SAVXH,PH1,XH          SAVE MH7 ROW POINTER
       ASSIGN      2+,30,PH              PH2 POINTS TO SECOND PORTION
*                                        OF MH7
       ASSIGN      1,MH7(PH1,1),PH       CHANGE PH1 FROM MH7 ROW REFERENCE
*                                        TO STORAGE SUBSCRIPT (SET IN FORT
       ASSIGN      2,MH7(PH2,1),PH       CHANGE PH2 TO REQUIRED STORAGE
*                                        CAPACITY (SET IN FORTM)
       MSAVEVALUE  7.XH$SAVXH,1,0,MH     ZERO OUT MH7
       SAVEVALUE   SAVXH+,30,XH
       MSAVEVALUE  7,XH$SAVXH,1,0,MH
       SAVEVALUE   NSCXH-,1,XH           COUNT DOWN OF XACS USING MH7
       TEST E      PH2,0,CHG02           FT IF NO LOWERING NEEDED
       GATE SNF    PH1,*+3               IF NEW CAPACITY HIGHER ... (*+3)
       ENTER       PH1                   ALLOW XACS WAITING ON DELAY
       LEAVE       PH1                   CHAIN TO START MOVING
       TERMINATE
 CHG02 GATE SNF    PH1                   WAIT TILL SOMEONE LEAVES
       HELPA       FORTM,23,2,PH1,PH2    PUSH STORAGE CAPACITY DOWN TO
*                                        CURRENT CONTENTS
       TEST E      XH$SLCXH,1,CHG02      STORAGE LOWERING COMPLETE
       SAVEVALUE   SLCXH,0,XH
       TERMINATE
*
*
*   TIMER  -  INITIAL   FORTRAN   CALL
*
*   INITIAL CALL TO FORTRAN HELP BLOCK TO:
*                  READ FLIGHT SCHEDULE AND GEOMETRY DATA,
*                  INITIALIZE MATRIX BASE ADDRESSES,
*                 EXECUTE CLINK AND MNLINK.
*
       GENERATE    ,,,1,127,1PH
       ADVANCE     XH$CLKXH              NEEDED TO DEFINE "CLKXH" AS XH
       SAVEVALUE  INCXH,60,XH           SET "CLOCK" INCREMENT              01440000
       HELPC      CLINK,1                                                  01450000
       HELPA      FORTM,1,1                                                01451000
       TEST G     PH1,0,*+3                                  (*+3)         01452000
 STOPO SPLIT      1,STOPO                                                  01453000
       TERMINATE  PH1                                                      01454000
       SPLIT      1,*+4                                      (*+4)         01455000
       GATE LS    JOBLS             SET IF ERROR COUNT OVER LIMIT          01456000
 STOP1 SPLIT      1,STOP1                                                  01457000
       TERMINATE  100                                                      01458000
       SPLIT      1,*+5                                      (*+5)         01459000
       ADVANCE    V$INC1V               CLOCK INCREMENT                    01460000
       HELPA      FORTM,21,XH$INCXH                                        01461000
       ADVANCE    XH$INCXH              CLOCK INCREMENT                    01462000
       TRANSFER   ,*-2                                       (*-2)         01463000
       ASSIGN     1,127,PH              LOOP COUNTER                       01464000
       MSAVEVALUE 1,1,PH1,FN$RANDF,ML   FILL ML1 WITH RANDOM NUMBERS       01465000
       LOOP       1PH,*-1                                                  01466000
       PRIORITY   126                   ALLOWS "CLOCK" TO OCCUR 1ST        01467000
       ADVANCE    XF$ENDXF              ADVANCE TO END OF RUN              01468000
       HELPA      FORTM,20,C1                                              01469000
*      PRINT      ,,C,C                 CLOCK TIME                         01469010
       PRINT      ,,W,C                 BLOCK COUNTS                       01469020
       PRINT      ,,S,C                 STORAGE STATISTICS                 01469030
       PRINT      ,,Q,C                 QUEUE STATISTICS                   01469040
       PRINT      ,,U,C                 USER CHAIN STATISTICS              01469060
       PRINT      ,,T,C                 TABLE STATISTICS                   01469070
*      PRINT      ,,X,C                 CONTENTS OF FULLWORD SAVEVALUES    01469080
*      PRINT      ,,XH,C                CONTENTS OF HALFWORD SAVEVALUES    01469090
*      PRINT      ,,XB,C                CONTENTS OF BYTE SAVEVALUES        01469100
*      PRINT      ,,XL,C                CONTENTS OF FLOATING POINT SAVE    01469110
*      PRINT      ,,LG,C                STATUS OF LOGIC SWITCHES           01469160
       TERMINATE  1                                                        01470000
*                                                                          01471000
       GENERATE   11700,,,1                                                01472000
       HELPA      FORTM,22                                                 01473000
       ADVANCE    300                                                      01474000
       TRANSFER   ,*-2                                                     01475000
*                                                                          01476000
*      GENERATE   12300,,,1                                                01477000
       TERMINATE  1                                                        01478000
*                                                                          01478100
*             TIMER FOR CHANGING HOURLY TABLE FOR PAX WAITING TIME         01478200
       GENERATE   ,,,1,,0               GENERATE TIMER TRANSACTION         01478300
       SAVEVALUE  PXTBN+,1,XB           INCREMENT NUMBER OF TABLE          01478400
       ADVANCE    3600                  ADVANCE 1 HOUR                     01478500
       TRANSFER   ,*-2                                                     01478600
*                                                                          01479000
 1     FUNCTION   PH1,L57       MNEMONIC LINK FUNCTION-SEE FORTRAN CALL    01480000
,CMH01/,CMH02/,CMH03/,CMH04/,CMH06/,CMH07/,CMH08/,CMH09                    01481000
,CML02                                                                     01482000
,ENDXF/,TRVXH/,BDTXH/,ABUXH/,DBUXH/,XFRXH/,XFAXH/,XFDXH/,SCLXH/,CLKXH      01483000
,CUSQS/,RCRQS/,DPCBS/,EPCBS/,CHKQS/,SECQS/,GAQSL/,PARQS/,IMMQS/,TICQS      01484000
,RCAR0/,BAGC0/,DPLC0/,CHEK2/,CHEK3/,CGTR0/,ERROR/,SECU0/,CTRL0/,CTRL1      01485000
,TRX99/,CONXH/,CHGXF/,NSCXH/,SLCXH/,DPDPS/,DPQCS/,WWGXH/,GRGXL/,EPOPS      01486000
,EPQCS/,GRT00/,GRTXL/,CPKXH/,CRBXH/,CGTXL/,PCBXL                           01487000
,JOBLS                                                                     01488000
*                                                                          01489000
*                                                                          01490000
       SIMULATE   9                                                        01491000
*      START      1,NP                                                     01492000
*      RESET                                                                01492002
       START      1,NP                                                      01492004
        NOXREF                                                              01492010
        REPORT                                                              01492015
*HMS    TITLE      1,FLIGHT SCHEDULE                                        01492020
 HMS    TITLE      2,AIRLINE INFORMATION TABLE                              01492030
*HMS    TITLE      3,TABLE OF POINTS                                        01492040
*HMS    TITLE      4,% ENPLANING PAX TICKETED                               01492050
*HMS    TITLE      5,TRANSFER FLIGHT TABLE                                  01492060
 HMS    TITLE      6,WALKING TIME BETWEEN POINTS                            01492070
 HMS    TITLE      7,RANDOM NUMBERS FOR BAGS                                01492080
*HMS    TITLE      8,USED IN MATRIX 9                                       01492090
*HMS    TITLE      9,FACILITY TABLE                                         01492100
 HMS    TITLE      11, COUNTER FORPAX LEAVING CONCOURSE                     01492110
 HMS    TITLE      12, COUNTER FOR PAX LEAVING SECURITY                     01492120
 HMS    TITLE      13, COUNTER FOR PAX LEAVING AIRLINES                     01492130
*LMS    TITLE      1,RANDOM NUMBER TABLE                                    01492140
*LMS    TITLE      2,GROUND TRANSACTION MODAL CHOICE                        01492150
*       OUTPUT     REMOVE * TO GET ALL STATICS                              01492160
*                                                                           01493000
*                                                                           01494000
        END                                                                 01495000
```