METROPOLITAN WASHINGTON COUNCIL OF GOVERNMENTS
777 North Capitol Street, N.E.
Suite 300
Washington, D.C. 20002-4239

MEMORANDUM

TO: Files

FROM: Ronald Milone

DATE: October 1, 2004

SUBJECT: Mobile Emissions Post–Processor Description and Results

1.0 Introduction

This memorandum describes the operation of TPB's post-processor used in the 2004 CLRP/FY-2005-2010 TIP Air Quality Conformity Determination. The post-processor calculates emissions by combining travel demand outputs from the Version 2.1 D Draft #50 model and emission rates from the EPA mandated MOBILE 6.2 (final) emission rate model.

The rationale behind the use of a post-processor is addressed in federal guidance:

> Since emissions are extremely sensitive to vehicle speed, EPA and DOT recommend that speeds be estimated in a separate step after traffic assignment (also known as "post-processing"), using refined speed-volume relationships and final assigned traffic volumes. Post-processed speeds estimated in the validation year should be compared with speeds empirically observed during peak- and off-peak periods…. Based on these comparisons, speed-volume relationships used for speed post-processing should be adjusted to obtain reasonable agreement with observed speeds. *Regardless of the specific analytical technique, every effort must be made to ensure that speed estimates are credible and based on a reproducible and logical analytical procedure*.[1]

Although highway volume and speed data are provided at the network link level as an output of the travel model traffic assignment step, speeds resulting from the assignment process are generally considered to be unsuitable for direct use in computing emissions. The primary focus of the traffic assignment is to provide accurate volumes for each time period, and so speed accuracy at the desired temporal level of detail is not addressed by the travel model. The post-processor serves to improve and refine link speeds by developing reasonable hourly volume estimates combined with speed-flow curves developed from observed speed surveys in the Washington, D.C. area. At the same time the post-processor attempts to maintain consistency with the demand information provided by the travel model.

Highway speeds developed from regional travel demand models are subject to inherent limitations. Operational considerations such as queuing delay at intersections are not well

[1] Transportation Conformity Reference Guide, FHWA, Revised 7/31/2001. Page D-6-9.

reflected in the regional model. Behavioral considerations relating to how travelers respond to congestion are also not fully understood, let alone well modeled. Despite these limitations the post–processor serves to develop hourly speeds in a sound and reasonable way for computing mobile emissions.

The post-processor is used to develop estimates of mobile emissions attributable to *modeled* trips and VMT, which do not account for the entire universe of mobile source emissions. "Off-network" components include vehicle-related (diurnal and resting loss) emissions as well as emissions relating to local road, bus, and park-and-ride travel. The off-network emission components are developed using an assortment of off-line procedures which are not addressed in this memorandum.

The next section describes the updates that have been recently implemented to TPB's post–processor. Most of the updates are based on recommendations suggested by a recent expert panel that reviewed the TPB's travel model and mobile emissions models. A more detailed review of the post-processor follows.

2.0 Recent Updates to the Post-Processor

The post-processor used in the current (FY-2005-2010) conformity study includes updates that were implemented to the emissions process used in the last (FY-2004-2009) study. The updates include:

1) The MOBILE model software was upgraded from MOBILE 6.0 to MOBILE 6.2(final). The upgraded MOBILE model effectively reduced CO emission rates (both trip-end and running) associated with light-duty trucks. VOC and NOx emission rates were unaffected by the upgrade.

2) The development of initial (pre-spread) hourly traffic volumes in the post-processor now includes a step that ensures that the initial hourly volumes will sum exactly to the traffic volumes produced by the traffic assignment for each time period (AM peak, PM peak, and off-peak). Previously, such consistency was maintained only for 24-hour volumes. Furthermore, the hourly distributions used to generate the initial volumes have been adjusted to replicate peaking assumptions that are used the traffic assignment process.

3) Congested freeway capacities are now adjusted to reflect the fact that 'through-put' volume declines during periods of extreme congestion. Previously the freeway LOS 'E' capacity was preserved irrespective of computed V/C levels.

4) The relationships between speed and flow have been refined to better reflect realistic conditions as observed in TPB freeway and arterial travel monitoring studies. The speed-flow assumptions now include speed 'floors' across all facility types. Freeway speeds do not decline beyond 17 mph. Arterial speeds include 'floors' of between 8 and 16 mph depending on the area type of the link. Arterial and collector speed-flows assumptions are more detailed than before, and also reflect declining speeds beyond a

V/C of 1.0. Final hourly volumes are set to be consistent with final speeds based on observed data from TPB travel monitoring studies.

3.0 Post-Processor Overview

The mobile emission post-processor produces regional estimates of daily air pollution generated by motor vehicles operating on the regional highway network. The emission calculation is based on vehicle-specific rates produced by the MOBILE model. The pollutants modeled are volatile organic compounds/hydrocarbons (VOC/HC), carbon monoxide (CO), and oxides of nitrogen (NOx). Mobile emissions are computed essentially by multiplying a unit of travel by an associated emission rate. Mobile emissions in most metropolitan areas are typically computed using a single *per VMT* rate which reflects all facets of the vehicle trip cycle (i.e., the starting, hot-stabilized, soaking, and evaporative stages). The TPB emissions calculation is not based on the 'single rate' approach. Instead, emissions are computed separately for each stage of the trip cycle. Specifically, *per trip* rates are developed to compute starting and soaking emissions, while *per VMT* rates are developed to compute hot-stabilized (or running) emissions. Evaporative (or diurnal) rates are also computed on a *per vehicle* basis. This more detailed computation approach is sometimes referred to as the 'hybrid' emission estimation method.

Emission rates used in COG/TPB's post processor are formulated on a county-by-county basis as inspection policy and fleet characteristics inputs to the MOBILE model vary by jurisdiction. These include the vehicle registration distributions, diesel sales fraction distributions, vehicle 'mix' distributions, and parameters relating to the jurisdiction's participation in state and federal air quality programs (e.g., I/M procedures and the use of reformulated gasoline). Table 1 indicates the 27 counties and external station groups for which mobile emission rates are prepared. The table indicates that specific emission rates are explicitly developed for 16 of the 27 jurisdictions. 'Nearest-neighbor' emission rates are used for the remaining 11 areas. A map of the emission areas is shown as Exhibit 1. The exhibit also indicates the extent of the modeled study area, which, for the most part, extends beyond the Washington Metropolitan Statistical Area (MSA). The non-attainment area is defined as the Washington MSA.

An overview of the mobile emissions post-processor is shown as Exhibit 2. The exhibit graphically shows how the post-processor relates to the MOBILE 6.2 model and the travel demand model. The MOBILE 6.2 model is executed 16 times as described above. Each execution produces 134 separate 'scenarios' representing the following conditions:

MOBILE6 'Scenarios'	Operating Mode	Facility Type	Speed Specifications
1-65	Stabilized	Arterial	1 to 65 mph in 1 mph increments
66-130	Stabilized	Freeway, Non-Ramp	1 to 65 mph in 1 mph increments
131	Stabilized	Freeway Ramp	-
132	Cold	Local	-
133	Hot	Local	-
134	Stabilized	Local	-

Running emission rates and soak rates are developed from the results of scenarios 1-131. As shown, the running rates are developed on a speed and facility type basis. Scenarios 132 and 133

are used to develop the starting emission rates. Starting rates are developed specifically for cold and hot starting conditions. The local rates developed in scenario 134 are used to support off-network emission calculations.

Exhibit 2 also indicates that a processing step named *M6RATES* follows immediately after MOBILE processing. The step is necessary for three reasons. First, the M6RATES program converts vehicle-specific rates into composite rates. Composite emission rates are computed by weighting the vehicle-specific rates by the associated VMT proportions assumed in the MOBILE program. Secondly, the M6RATES program specially computes emission rates suitable for trip-end and vehicle-related emission calculations. Because the MOBILE 6.2 model produces all rates on a *per mile* basis, transformations are necessary to produce *per trip* rates to compute start and soak emissions and *per vehicle* rates to compute diurnal emissions. The transformations are made using regional estimates of trip starts and miles driven by specific vehicle classifications. The estimates are extracted from the database output of the MOBILE 6.2 model and are entered into the M6RATES program as user-defined parameters.[2] Finally, the M6RATES program produces the composite rates in a fixed file format which facilitates subsequent computer processing. The program is executed in a 'batch' fashion for each jurisdiction. It ultimately produces 64 ASCII files, four rate types (freeway rates, arterial rates, ramp rates, and start-up) for each of the 16 jurisdictions. These ASCII files are accessed directly by the post-processor.

Example listings of the four emission rate files for a single jurisdiction are provided in Attachment 2. The arterial and freeway running rate files consist of four columns of numbers, the speed and associated VOC, CO, and NOx emission rates in gm/mile. The remaining two files each consist of one line of data, and contain start-up emission rates (hot transient VOC, CO, and NOx and cold start VOC, CO, and NOx rates expressed as gm/trip) and freeway non-ramp soak emission rates (ramp VOC, CO, and NOx rates in gm/mi and VOC soak rates in gm/trip).

Exhibit 2 indicates that trip tables and loaded link volumes produced by the travel demand model are inputs to the mobile emissions post-processor. The Version 2.1 D Draft #50 model produces trip tables and loaded link traffic volumes on the basis of three discrete time periods: the AM peak period (6:00 AM-9:00 AM), the PM peak period (4:00 PM-7:00 PM) and the off-peak period hours (i.e., the remaining 18 hours of the day). The trips and link volumes are those resulting from the final (i.e., the sixth) iteration of the travel demand model.

[2] A detailed description of the transformations is given in a December 12, 2002 Memorandum from Ronald Milone to the files, Subject: Development of Composite Emission Rates from MOBILE6 Listings.

Table 1
Jurisdictional Emission Areas
(Locations for which emission rates are calculated)

Emission Area System Number	Jurisdiction / External Station Group	TAZ Range	'Nearest Neighbor' Emission Rates Used? (Y/N)	'Nearest Neighbor' Jurisdiction
1	Washington, DC	1 – 319	N	*N/A*
2	Montgomery	320 – 627	N	*N/A*
3	Prince George's	640 – 1020	N	*N/A*
4	Howard	1080 – 1099	Y	Prince George's
5	Anne Arundel	1110 – 1142	Y	Prince George's
6	Carroll	1060 – 1073	Y	Prince George's
7	Baltimore Externals	2172 – 2191	Y	Prince George's
8	Calvert	1150 – 1163	N	*N/A*
9	Charles	1200 – 1223	N	*N/A*
10	Frederick	1030 – 1053	N	*N/A*
11	Frederick Externals	2169 – 2171	Y	Frederick
12	Arlington	1230 – 1311	N	*N/A*
13	Fairfax	1400 – 1755	N	*N/A*
14	Loudoun	1780 – 1905	N	*N/A*
15	Prince William	1920 – 2061	N	*N/A*
16	Stafford	2080 – 2093	N	*N/A*
17	Alexandria	1330 – 1389	N	*N/A*
18	St. Mary's	1170 – 1190	N	*N/A*
19	Washington Co. Externals	2164 – 2168	N	*N/A*
20	Clarke	2130 – 2132	N	*N/A*
21	Fauquier	2115 – 2125	Y	Clarke
22	Jefferson W. Virginia	2135 – 2141	Y	Clarke
23	Western Externals	2154 – 2163	Y	Clarke
24	Spotsylvania	2105 – 2110	N	*N/A*
25	King George	2070 – 2074	Y	Spotsylvania
26	City of Fredericksburg	2100 – 2101	Y	Spotsylvania
27	Southern Externals	2145 – 2153	Y	Spotsylvania

N/A – Not applicable

- MSA Indicator

Exhibit 1
Mobile Emission Factor Area System (1 -27)

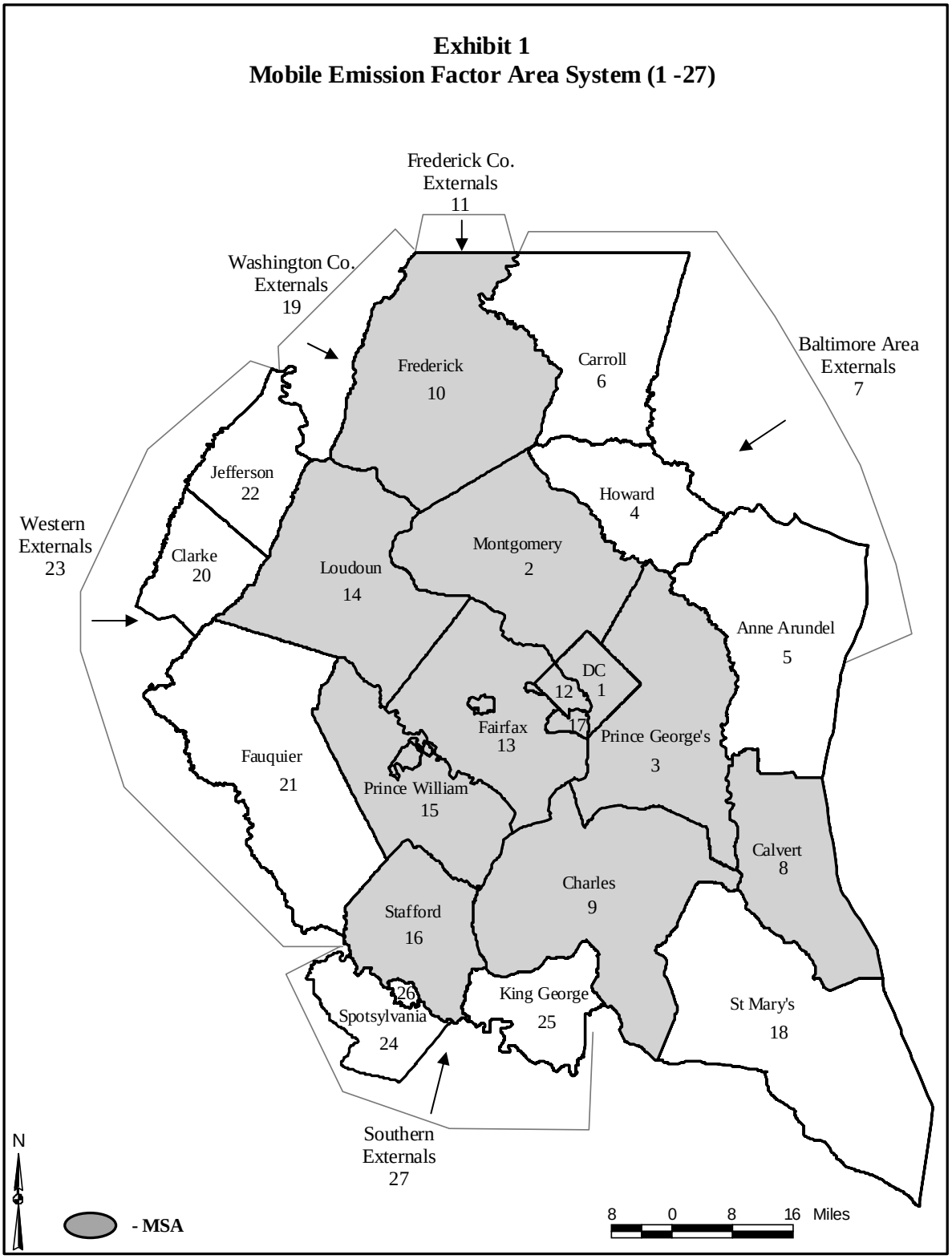
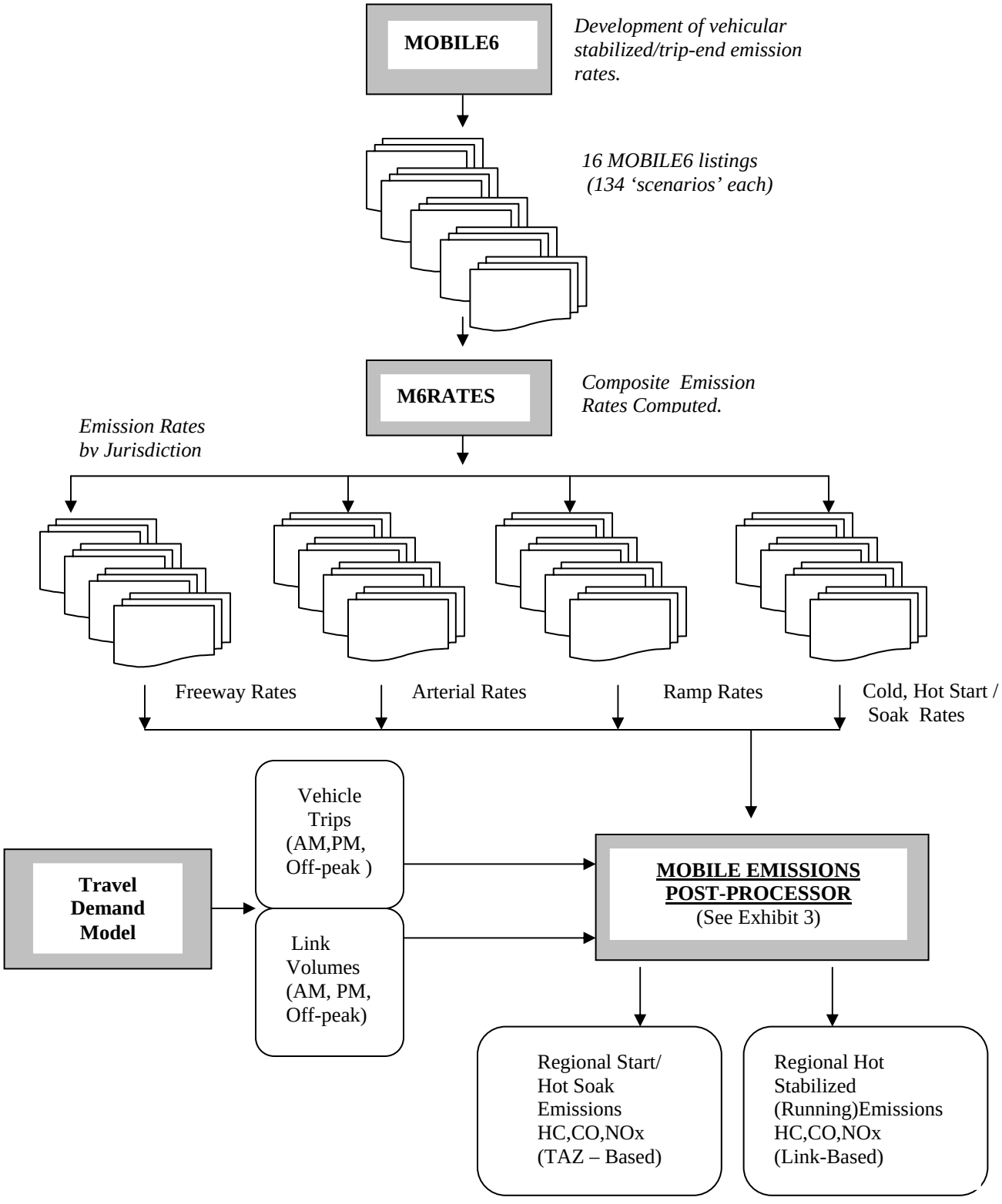


Exhibit 2
Mobile Emissions Development Process



4.0 Post-Processor Program Steps

The post-processor consists of a series of five program steps graphically shown as Exhibit 3. The steps are executed with TP+ programs. The five steps are summarized below. (Associated TP+ script names are in parenthesis):

1) Trip Table Formatting (AQTRIPS.S): AM, PM, and off-peak trip tables produced by the travel demand model are read. The program produces zonal trip-ends for each of the three time periods. It also produces a file containing the proportion of daily vehicle trips from/to each of the 27 emission areas. Since the trip proportions are developed with daily trips, the proportion in the *i/j* direction is generally the same as that in the *j/i* direction.

2) Time-of-Day Trip-Ends Program (ZONESPRD.S): The program reads the zonal origins and destinations, described above, and apportions them among discrete hourly periods.

3) Start/Soak Emissions Program (STRT_SKR.S): The program applies emission rates to the trip- ends to compute start-up and soaking emissions on a zone-by-zone and hour-by-hour basis. The program reads 1) hourly trip-ends, 2) the MOBILE6-generated cold/hot starting rates and soak rates, and 3) the county-level trip proportions file. HC, CO, and NOx starting emissions and HC soak emissions result from the program.

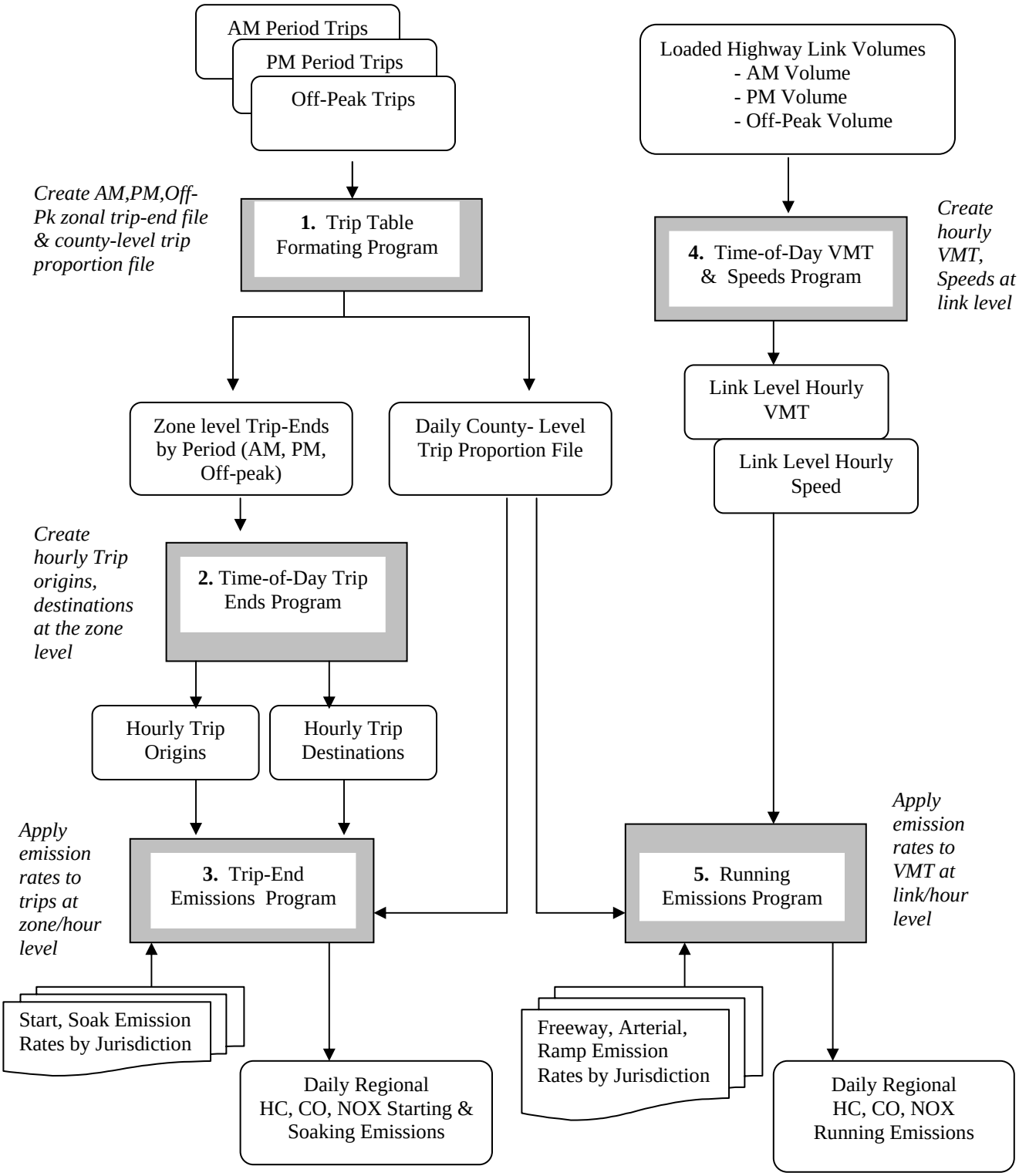
4) Time-of-Day VMT and Speeds Program (PEAK_SPREAD.S): The program reads the AM, PM, and off-peak network link volumes produced by the travel demand model. It produces hourly volumes, VMT, and restrained speed for each highway link.

5) Running Emissions Program (RUNNING.S): The program computes hot stabilized emissions on a link-by-link and hour-by-hour basis. It reads 1) the hourly link VMT and highway speed files developed above, 2) MOBILE6-based running emission rates which are provided on the basis of speed, and 3) the county level trip proportions file. HC, CO, and NOx running emissions result from the program

Listings of the above scripts are provided in Attachment 1. A batch file is normally used to facilitate the execution the above scripts in a 'command prompt' window. An example listing of the batch file is also contained in Attachment 1. The batch file defines information on the location of travel demand / emission rate inputs as environmental variables that are referenced in the scripts.

Details concerning the above steps are presented below.

Exhibit 3
TPB Mobile Emissions Post-Processor Steps



5.0 Post–Processor Computations

Starting emissions are computed by applying per-trip emission rates to vehicle trips at the zone level and are computed on an hour-by-hour basis. Starting pollutant rates are associated with VOC, CO, and NOx emissions, and are expressed in terms of *cold* and *hot transient* types. A cold start relates to those auto trips with fully cooled engines (i.e., engines that have been turned off for at least one hour prior to the trip starting time). Alternatively, hot transient starts are those auto trips with warm engines (i.e., engines turned off less than one hour prior to the trip start time). An hourly allocation of trip origins is necessary for the starting emission calculation since the proportion of cold and hot starts vary by the time of day. The assumed hourly distribution of AM, PM, and Off-peak vehicle trips is shown on Table 2. The distribution shown was derived from the 1994 Household Travel Survey (HTS). The assumed hourly distribution for cold and hot transient starts is shown on Table 3. This table was also derived from the 1994 HTS. The table logically indicates that the share of hot vehicle starts is low in the early morning hours and generally higher later in the day.

Soaking emissions are associated with the evaporative VOC emissions that result when the engine is turned off. The soak emissions consist of a single emission rate that is applied to trip destinations. There is no temporal component to the soaking emission computation.

It was stated earlier that emission rates are developed on a county-by-county basis. An averaged emission rate is used in the post processor, as opposed to a single county-specific rate, because the vehicle starts in any given jurisdiction are realistically made by residents of that jurisdiction as well as by residents of many other jurisdictions. For example, the emission rate used within the District of Columbia is the average of all emission rates weighted by the proportion of daily vehicle trips from each jurisdiction to the District. The general equation for computing starting emissions for a specific TAZ and hour of the day is as follows:

$$StartEm_{ih} = Starts_h * \sum_{j=1}^{27} ((CSR_j * CPCT_h + HSR_j * HPCT_h) * Tprop_{ij})$$

Where:

$StartEm_{ih}$ = Zonal starting-up emissions (in grams) at hour h in jurisdiction i
$Starts_h$ = Zonal vehicle starts at hour h
CSR_j = Cold Start rate (gm/trip) for jurisdiction j
$CPCT_h$ = Cold start proportion at hour h
HSR_j = Hot Start rate (gm/trip) for jurisdiction j
$HPCT_h$ = Hot start proportion at hour h
$Tprop_{ij}$ = Proportion of daily trips between jurisdiction i/j

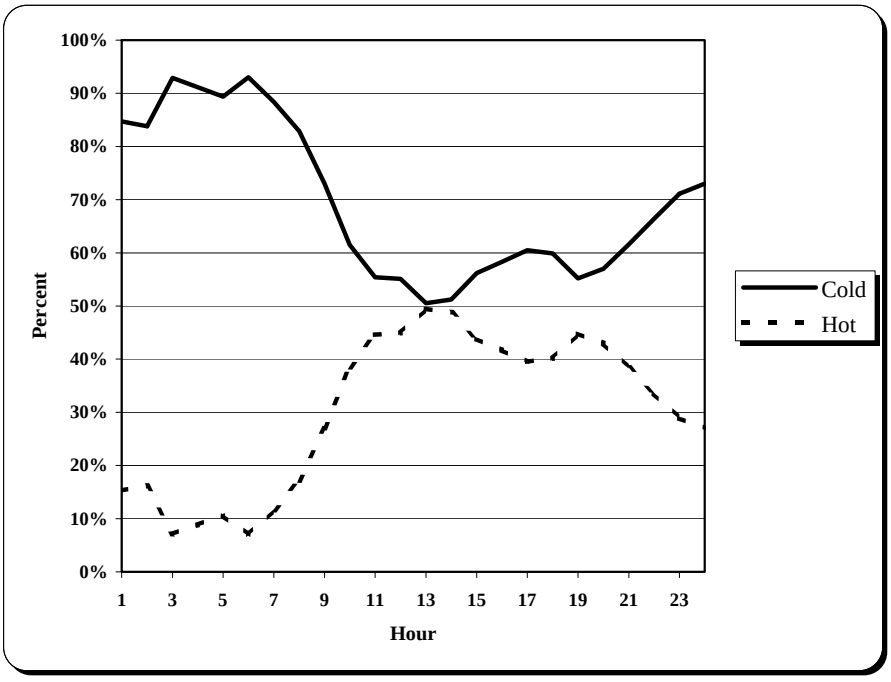
Table 2

Distribution of AM, PM, and Off-Peak Period Auto Driver Trips
Among
Hourly Periods

Hour No.		% AM	% PM	Off-Peak
1	12mid - 12:59AM			0.30%
2	1:00AM - 1:59AM			0.40%
3	2:00AM - 2:59AM			0.30%
4	3:00AM - 3:59AM			0.30%
5	4:00AM - 4:59AM			0.50%
6	5:00AM - 5:59AM			2.20%
7	6:00AM - 6:59AM	20.10%		
8	7:00AM - 7:59AM	39.80%		
9	8:00AM - 8:59AM	40.10%		
10	9:00AM - 9:59AM			9.70%
11	10:00AM - 10:59AM			8.20%
12	11:00AM - 11:59AM			9.20%
13	12noon - 12:59PM			10.10%
14	1:00PM - 1:59PM			8.90%
15	2:00PM - 2:59PM			9.00%
16	3:00PM - 3:59PM			11.60%
17	4:00PM - 4:59PM		31.40%	
18	5:00PM - 5:59PM		37.30%	
19	6:00PM - 6:59PM		31.30%	
20	7:00PM - 7:59PM			10.80%
21	8:00PM - 8:59PM			7.70%
22	9:00PM - 9:59PM			5.80%
23	10:00PM - 10:59PM			3.40%
24	11:00PM - 11:59PM			1.60%
Total		100.00%	100.00%	100.00%

Table 3
Distribution of Cold / Hot Transient Vehicle Starts by Hour

Hour No.		% Cold	% Hot	Total
1	12mid - 12:59AM	84.70%	15.30%	100.00%
2	1:00AM - 1:59AM	83.80%	16.20%	100.00%
3	2:00AM - 2:59AM	92.90%	7.10%	100.00%
4	3:00AM - 3:59AM	91.20%	8.80%	100.00%
5	4:00AM - 4:59AM	89.40%	10.60%	100.00%
6	5:00AM - 5:59AM	93.00%	7.00%	100.00%
7	6:00AM - 6:59AM	88.40%	11.60%	100.00%
8	7:00AM - 7:59AM	82.90%	17.10%	100.00%
9	8:00AM - 8:59AM	73.00%	27.00%	100.00%
10	9:00AM - 9:59AM	61.50%	38.50%	100.00%
11	10:00AM - 10:59AM	55.40%	44.60%	100.00%
12	11:00AM - 11:59AM	55.10%	44.90%	100.00%
13	12noon - 12:59PM	50.50%	49.50%	100.00%
14	1:00PM - 1:59PM	51.20%	48.80%	100.00%
15	2:00PM - 2:59PM	56.20%	43.80%	100.00%
16	3:00PM - 3:59PM	58.30%	41.70%	100.00%
17	4:00PM - 4:59PM	60.50%	39.50%	100.00%
18	5:00PM - 5:59PM	59.90%	40.10%	100.00%
19	6:00PM - 6:59PM	55.20%	44.80%	100.00%
20	7:00PM - 7:59PM	57.00%	43.00%	100.00%
21	8:00PM - 8:59PM	61.60%	38.40%	100.00%
22	9:00PM - 9:59PM	66.40%	33.60%	100.00%
23	10:00PM - 10:59PM	71.10%	28.90%	100.00%
24	11:00PM - 11:59PM	73.00%	27.00%	100.00%



Similarly, the equation for computing hot soak emissions is as follows:

$$SoakEm_{ih} = Stops_h * \sum_{j=1}^{27} (HSR_j * Tprop_{ij})$$

Where:

$SoakEm_{ih}$ = Zonal hot soak emissions (in grams) at hour h in jurisdiction i
$Stops_h$ = Vehicle stops at hour h
HSR_j = Hot Soak rate (gm/trip) for jurisdiction j
$Tprop_{ij}$ = Proportion of daily trips between jurisdiction i and jurisdiction j

The regional total of starting/soaking emissions is, therefore, based on the result of the above equations accumulated over all TAZ's, over all hours of the day. Regional emissions in grams are converted to tons using a conversion factor of 907,184.74 gm/ton.

6.0 Running (Hot Stabilized) Emissions

Running emissions relate to HC, CO, and NOx pollutants. They are computed by applying per mile emission rates to VMT at the network link level, on an hour-by-hour basis. The calculation is applied on an hourly basis because the running emission rates are provided as a function of highway speed, which varies with congestion throughout the day. As with the trip-end emission calculation, the running emission rate for a given link is a weighted average of all jurisdictional rates based on the proportion of daily vehicle trips from each county to the specific county associated with the link.

The allocation of link volumes among hourly periods is done in a two-step manner. First, an initial hourly distribution based on observed data for the Washington region is applied to the daily link volume, based on the facility class and *peaking* classification of the link. Facility classifications are defined as freeway, arterial, or local. COG has established three peaking types, AM-oriented, PM-oriented, and Even, based on the following *peaking percentage*[3]:

Peaking Percentage = ((AM Volume* PM scale factor) – PM Volume) / Daily Link Volume

Where:
Peaking Percentage > 7.5% indicates AM oriented class
Peaking Percentage < - 7.5% indicates PM oriented class
Peaking Percentage - 7.5% to 7.5% indicates Even oriented class

The PM scale factor shown is applied to all AM period volumes so that the sum of regional AM link volumes will equal the sum of regional PM volumes. The scaled volume is used *only* for the purpose of computing the peaking index, and is necessary to ensure that a reasonable regional balance of AM and PM oriented links are attained. Default hourly distributions associated with specific facility and peaking classifications are shown on Table 4. The distribution selected for a

[3] See August 27, 2002 Memorandum from Michael Freeman to File, Subject: Development and Recommendations of Hourly Distributions of Daily Traffic Volumes.

Table 4
Hourly Distribution of Daily Traffic by Orientation and Facility Type

Hour		AM			PM			EVEN		
No.		Freeway	Arterial	Collector	Freeway	Arterial	Collector	Freeway	Arterial	Collector
1	12mid - 12:59AM	0.77	0.49	0.34	1.11	0.76	0.62	1.07	0.67	0.52
2	1:00AM - 1:59AM	0.55	0.30	0.20	0.64	0.41	0.32	0.73	0.40	0.31
3	2:00AM - 2:59AM	0.52	0.25	0.18	0.48	0.28	0.24	0.61	0.30	0.24
4	3:00AM - 3:59AM	0.72	0.37	0.29	0.42	0.24	0.20	0.68	0.33	0.30
5	4:00AM - 4:59AM	1.88	1.09	0.96	0.58	0.38	0.32	1.24	0.72	0.70
6	5:00AM - 5:59AM	6.20	4.05	3.80	1.38	1.08	0.96	3.60	2.27	2.37
7	6:00AM - 6:59AM	8.66	8.75	9.19	3.24	2.70	2.58	4.99	4.58	4.83
8	7:00AM - 7:59AM	11.13	12.38	13.40	4.63	4.62	4.67	6.96	7.65	8.06
9	8:00AM - 8:59AM	8.04	9.82	10.92	4.71	5.15	5.07	5.44	6.90	7.27
10	9:00AM - 9:59AM	6.94	6.39	6.10	3.84	4.38	4.10	5.93	6.11	5.80
11	10:00AM - 10:59AM	5.14	4.71	4.50	3.90	4.19	3.94	5.18	5.15	4.80
12	11:00AM - 11:59AM	4.68	4.53	4.51	4.21	4.67	4.54	5.15	5.40	5.14
13	12noon - 12:59PM	4.65	4.72	4.81	4.61	5.25	5.25	5.34	5.80	5.50
14	1:00PM - 1:59PM	4.58	4.64	4.64	4.83	5.21	5.01	5.45	5.68	5.34
15	2:00PM - 2:59PM	4.66	4.80	4.85	5.95	5.87	5.76	6.10	5.97	5.89
16	3:00PM - 3:59PM	4.70	5.09	5.17	7.32	7.14	7.03	6.80	6.62	6.68
17	4:00PM - 4:59PM	4.56	5.26	5.24	9.95	9.58	10.06	5.94	6.26	6.61
18	5:00PM - 5:59PM	4.76	5.55	5.58	10.87	10.93	11.57	6.63	7.15	7.66
19	6:00PM - 6:59PM	4.32	4.98	4.92	8.55	9.03	9.65	5.35	5.92	6.44
20	7:00PM - 7:59PM	3.66	3.90	3.72	5.61	6.01	6.17	4.99	5.29	5.45
21	8:00PM - 8:59PM	2.95	2.97	2.70	4.25	4.44	4.60	3.89	4.05	4.09
22	9:00PM - 9:59PM	2.64	2.40	2.01	3.68	3.58	3.52	3.44	3.27	3.06
23	10:00PM - 10:59PM	2.06	1.64	1.30	2.80	2.41	2.20	2.70	2.21	1.90
24	11:00PM - 11:59PM	1.23	0.92	0.72	2.45	1.71	1.62	1.81	1.29	1.05
Total		100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00

given link is applied to the *daily* link volume to arrive at initial hourly volume estimates. However, the hourly estimates are subsequently scaled so that the link volumes produced by the traffic assignment process are maintained in each time period (i.e., the AM peak, PM peak, and off-peak periods). The hourly link speed is developed from the volume–to-capacity ratio developed at this point based on the speed flow relationship shown on Table 6. The functions shown on Table 6 are based on observed speed and density data collected in the Washington region.

In the second step, the initial hourly volume is compared to the hourly link capacity (Level-of-Service 'E') and adjusted if necessary. The adjustment procedure (outlined on Table 5) begins with the comparison of AM peak hour traffic and PM peak hour traffic with the available capacity. If the initial peak hour volume exceeds capacity, then the peak hour volume is adjusted to equal the capacity and the portion of volume exceeding capacity is then apportioned in equal parts to the hour before and the hour after the peak hour. In the case of overly congested freeways, the capacity is moderated to reflect the fact that the 'through-put' volumes cannot be sustained at LOS 'E' service levels when the V/C ratio exceeds 1.0. Table 7 shows the assumed relationship between freeway capacities and congested V/C ratios. Because this adjustment could potentially cause the 'shoulder' hour volumes to exceed capacity, added steps are undertaken to compare the resulting volumes in each successive shoulder hour with the capacity. If a given shoulder hour volume exceeds capacity, then the volume is similarly adjusted to equal capacity and the 'overflow' volume is added to the volume of the adjacent hourly period. Traffic assignments on rare occasions produce severely overloaded link volumes to the point where a given link volume could exceed the capacity over *all* hours of the day. Because of this possibility, volume adjustments are *not* made for the first, noon, and last hours (hours 1, 13, and 24), even if a given link volume is determined to exceed capacity in those particular hours.

Subsequent to the development of restrained speeds, the general equation for computing running emissions is:

$$RunningEm_{ih} = VMT_h * \sum_{j=1}^{27} (RRate_j * Tprop_{ij})$$

Where:

$RunningEm_{ih}$ = Running link emissions at hour h in jurisdiction i
VMT_h = Vehicle Miles Travel (after peak-spreading) at hour h
$RRate_j$ = Running rate (gm/mi) as a function of highway speed for jurisdiction j
$Tprop_{ij}$ = Proportion of daily trips between jurisdiction i/j

The regional running emissions are the accumulation of calculated hourly emissions over all network links in the study area. Emissions in grams are converted to tons using a conversion factor of 907,184.74 gm/ton.

Table 5	
Peak Spreading Procedure ***Adjustment Process for Spreading Hourly Volumes When Initial Volumes Exceed Capacity***	
Step 1:	The AM peak hour (hour 8) initial volume is compared to the link capacity. If the initial hour 8 volume exceeds capacity, then the hour 8 volume is set to capacity (or a moderated capacity value in the case of freeways) and the excess volume portion is added to the volume in periods occurring before *and* after the AM peak hour (hours 7 and 9) on a 50/50 basis.
Step 2:	The PM peak hour (hour 18) initial volume is compared to the link capacity. If the initial volume exceeds capacity, then the hour 18 volume is set to capacity (or a moderated capacity value in the case of freeways) and the excess volume portion is added to the volume in periods occurring before *and* after the PM peak hour (hours 17 and 19) on a 50/50 basis.
Step 3:	The volume occurring during pre-AM peak hours (hours 1 to 7) are sequentially checked against the link capacity as in steps 1 and 2, and adjusted (if necessary) in a backward-moving fashion. If the volume occurring in hour 7 exceeds capacity then the hour 7 volume is set to capacity and the excess volume portion is added to the volume of hour 6 volume, and so on. There is no volume spreading at hour 1, even for rare cases where the resulting hour 1 volume exceeds capacity.
Step 4:	The volume occurring during post-AM peak hours (hours 9 to 13) are sequentially checked against the link capacity as in steps 1 and 2, and adjusted (if necessary) in a forward-moving fashion. If the volume occurring in hour 9 exceeds capacity then the hour 9 volume is set to capacity and the excess volume portion is added to the volume of hour 10 volume, and so on. There is no volume spreading at hour 13 (the midday hour), even for rare cases where the resulting hour 13 volume exceeds capacity.
Step 5:	The volume occurring during pre-PM peak hours (hours 13 to 17) are sequentially checked against the link capacity as in steps 1 and 2, and adjusted (if necessary) in a backward-moving fashion. If the volume occurring in hour 17 exceeds capacity then the hour 17 volume is set to capacity and the excess volume portion is added to the volume of hour 16 volume, and so on. There is no volume spreading at hour 13 (the midday hour), even for rare cases where the resulting hour 13 volume exceeds capacity.
Step 6:	The volume occurring during post-PM peak hours (hours 19 to 24) are sequentially checked against the link capacity as in steps 1 and 2, and adjusted (if necessary) in a forward-moving fashion. If the volume occurring in hour 19 exceeds capacity then the hour 19 volume is set to capacity and the excess volume portion is added to the volume of hour 20 volume, and so on. There is no volume spreading at hour 24, even for rare cases where the resulting hour 24 volume exceeds capacity.

Table 6
Speed Delay Functions Used in the MWCOG Mobile Emissions Post Processor
By
Facility Type and Area Type (1-7)

V/C	Freeway			Major Arterial				Minor Arterial				Collector				Expressway		
Atp->	1-2	3-4	5-7	1-2	3-4	5	6-7	1-2	3-4	5	6-7	1-2	3-4	5	6-7	1-2	3-5	6-7
0.00	55.000	60.000	67.000	25.000	35.000	40.000	45.000	20.000	30.000	35.000	40.000	15.000	20.000	25.000	30.000	45.000	50.000	55.000
0.10	54.945	59.945	66.940	24.300	33.600	38.600	43.600	19.400	28.900	33.800	39.200	14.500	19.300	24.300	28.800	44.945	49.939	54.933
0.20	54.890	59.890	66.880	23.600	32.200	37.200	42.200	18.800	27.800	32.600	38.400	14.000	18.600	23.600	27.600	44.890	49.878	54.866
0.30	54.810	59.800	66.790	22.900	30.800	35.800	40.800	18.200	26.700	31.400	37.600	13.500	17.900	22.900	26.400	44.820	49.800	54.780
0.40	54.710	59.690	66.670	22.200	29.400	34.400	39.400	17.600	25.600	30.200	36.800	13.000	17.200	22.200	25.200	44.730	49.700	54.670
0.50	54.570	59.540	66.490	21.500	28.000	33.000	38.000	17.000	24.500	29.000	36.000	12.500	16.500	21.500	24.000	44.620	49.578	54.536
0.60	54.370	59.300	66.180	20.800	27.000	31.600	36.400	16.400	23.400	27.800	35.000	12.000	16.000	20.800	23.000	44.470	49.411	54.352
0.70	54.060	58.910	65.600	20.100	26.000	30.200	34.800	15.800	22.300	26.600	34.100	11.500	15.500	20.100	22.000	44.260	49.178	54.096
0.80	53.540	58.170	64.260	19.400	25.000	28.800	33.200	15.200	21.200	25.400	31.400	11.000	15.000	19.400	21.000	43.970	48.856	53.741
0.90	52.560	56.560	60.840	18.700	24.000	27.400	31.600	14.600	20.100	24.200	28.700	10.500	14.500	18.700	20.000	43.530	48.367	53.203
1.00	50.580	53.220	55.280	18.000	23.000	26.000	30.000	14.000	19.000	23.000	26.000	10.000	14.000	18.000	19.000	42.820	47.578	52.336
1.10	46.860	48.550	49.875	16.600	20.800	23.400	27.200	12.800	17.600	21.000	23.600	9.200	12.800	16.600	17.600	41.250	45.833	50.417
1.17	44.256	45.281	46.092	15.620	19.260	21.580	25.240	11.960	16.620	19.600	21.920	8.640	11.960	15.620	16.620	40.151	44.612	49.073
1.20	43.140	43.880	44.470	15.200	18.600	20.800	24.400	11.600	16.200	19.000	21.200	8.400	11.600	15.200	16.200	39.680	44.089	48.498
1.30	39.335	39.870	40.315	13.800	16.400	18.200	21.600	10.400	14.800	17.000	18.800	7.600	10.400	13.800	14.800	36.925	41.028	45.131
1.40	35.530	35.860	36.160	12.400	14.200	15.600	18.800	9.200	13.400	15.000	16.400	6.800	9.200	12.400	13.400	34.170	37.967	41.763
1.50	32.470	32.740	32.990	11.000	12.000	13.000	16.000	8.000	12.000	13.000	14.000	6.000	8.000	11.000	12.000	31.420	34.911	38.402
1.60	29.410	29.620	29.820	11.000	12.000	13.000	16.000	8.000	12.000	13.000	14.000	6.000	8.000	11.000	12.000	28.670	31.856	35.041
1.80	24.550	24.700	24.850	11.000	12.000	13.000	16.000	8.000	12.000	13.000	14.000	6.000	8.000	11.000	12.000	24.050	26.722	29.394
2.00	20.610	20.730	20.860	11.000	12.000	13.000	16.000	8.000	12.000	13.000	14.000	6.000	8.000	11.000	12.000	20.230	22.478	24.726
2.25	16.650	16.750	16.850	11.000	12.000	13.000	16.000	8.000	12.000	13.000	14.000	6.000	8.000	11.000	12.000	16.350	18.167	19.983
99.99	16.650	16.750	16.850	11.000	12.000	13.000	16.000	8.000	12.000	13.000	14.000	6.000	8.000	11.000	12.000	16.350	18.167	19.983

Table 7
Freeway Through-Put Capacities Under Congested Conditions

V/C	Fwy AT1	Fwy AT2	Fwy AT3	Fwy AT4	Fwy AT5	FWY AT6	FWY AT7
1.00	1500	1600	1800	1800	2000	2000	2100
1.20	1433	1528	1719	1719	1911	1911	2006
1.40	1366	1457	1639	1639	1821	1821	1912
1.60	1299	1385	1559	1559	1732	1732	1818
1.80	1214	1295	1457	1457	1619	1619	1699
2.00	1128	1204	1355	1355	1505	1505	1580
2.25	1017	1085	1221	1221	1356	1356	1424
99.99	1017	1085	1221	1221	1356	1356	1424

7.0 Emission Results

The table below shows the MSA level emission and travel demand results for the post-processor in column (3). Column (3) reflects the Draft #50 travel model and the October 2004 post-processor (documented in this memorandum). Other post-processor results are shown as well for comparison purposes. Column (1) shows emission results of the previous conformity work (FY-2004-2009 TIP), based on the Version 2.1C travel model and the March 2003 post-processor. Column (2) shows results of a test run, where the March 2003 post-processor was executed in combination with the Draft #50 model output. The (3)-(1) column shows the updated Draft # 50 travel model and October 2004 post-processor produce lower emission results in comparison with the previous conformity results, by about 6 tons and 16 tons for VOC and NOx, respectively. The VMT decline of about 9 million between the Version 2.1C and the Draft #50 travel models is also notable. In contrast, the (3)-(2) difference column, intended to isolate the effect of the March 2003 and October 2004 post-processors, indicates that the updated post-processor also yields lower emissions, but the magnitude is much less than the difference shown in the (3)-(1) difference column. The updated post-processor produces 1.7 tons less for VOC and 0.85 tons less for NOx in comparison with the earlier March 2003 post-processor. These results indicate that both the Draft #50 travel model and the October 2004 post-processor in combination produce lower emissions in comparison with earlier conformity work using the Version 2.1/C travel model and the March 2003 post-processor, but the decline is primarily attributable to the updated Draft #50 travel model.

MSA Mobile Emissions -- Comparison of 2005 Emission Results

		(1)	(2)	(3)	Difference (3) - (1)	Difference (3) - (2)
Year		**2005**	**2005**	**2005**		
AQC Version		**2004-2009-TIP**	**2005-2010 TIP**	**2005-2010 TIP**		
Mobile Emissions Post-Processor:		**March 2003**	**March 2003**	**October 2004**		
Mobile Model		**M6.0**	**M6.2 final**	**M6.2 final**		
Travel Model:		**V. 2.1/TP+,C**	**V. 2.1D #50**	**V. 2.1D #50**		
Land Use		**Rnd 6.3**	**Rnd 6.4A**	**Rnd 6.4A**		
	Veh.Trip Starts	15,515,817	15,790,764	15,790,764	274,947	0
	Veh.Trip Stops	15,514,948	15,790,088	15,790,088	275,139	0
Trip-End Emissions (tons)	Start-Up / VOC	17.317	17.641	17.641	0.324	0.000
	Start-Up / Nx	10.890	11.092	11.092	0.202	0.000
	Hot Soaks / VOC	11.227	11.447	11.447	0.220	0.000
Avg Trip-End Rates (gm/trip)	Start-Up / VOC	1.012	1.013	1.013	0.001	0.000
	Start-Up / Nx	0.637	0.637	0.637	0.001	0.000
	Hot Soaks / VOC	0.656	0.658	0.658	0.001	0.000
VMT		126,453,600	117,389,600	117,389,600	-9,064,000	0
Average Highway Speed (mph)		39.3	39.6	40.9	1.6	1.3
Running Emissions (tons)	VOC	46.813	42.617	40.854	-5.959	-1.763
	Nx	202.118	186.958	186.108	-16.010	-0.850
Avg Running Rates (gm/mi)	VOC	0.336	0.329	0.316	-0.020	-0.014
	Nx	1.450	1.445	1.438	-0.012	-0.007

Table 8 is a summary of the final emission results produced by the Draft #50 travel model and the October 2004 post-processor over time. The years 2015 through 2030 include two ICC alternative alignments designated as "MP" for the master plan alignment and "NA" for the northern alignment.

Table 8

MSA Mobile Emissions Summary -- 2004 CLRP and FY 2005 to 2010 TIP Results

AQC Version	**2005-2010 TIP**
Mobile Emissions Post-Processor:	**Oct. 2004**
Mobile Model	**M6.2 final**
Travel Model:	**Version 2.1D #50**
Land Use	**Rnd 6.4A**

Year / Alternative		2005	2015/MP	2015/NA	2025/MP	2025/NA	2030/MP	2030/NA
	Veh.Trip Starts	15,790,800	17,776,800	17,791,900	19,270,900	19,288,300	19,797,500	19,819,000
	Veh.Trip Stops	15,790,100	17,775,100	17,790,200	19,269,100	19,286,200	19,795,100	19,817,600
Trip-End Emissions (tons)	Start-Up / VOC	17.641	8.137	8.144	6.555	6.561	6.493	6.500
	Start-Up / CO	257.661	171.662	171.806	168.848	168.996	170.313	170.501
	Start-Up / Nx	11.092	4.694	4.698	2.608	2.610	2.404	2.407
	Hot Soaks / VOC	11.447	5.966	5.971	3.845	3.848	3.948	3.952
Avg Trip-End Rates (gm/trip)	Start-Up / VOC	1.013	0.415	0.415	0.309	0.309	0.298	0.298
	Start-Up / CO	14.803	8.760	8.760	7.949	7.948	7.804	7.804
	Start-Up / Nx	0.637	0.240	0.240	0.123	0.123	0.110	0.110
	Hot Soaks / VOC	0.658	0.304	0.304	0.181	0.181	0.181	0.181
VMT		117,389,600	135,171,800	135,004,900	147,212,500	147,070,300	152,202,300	152,131,200
Average Highway Speed (mph)		40.9	40.9	40.8	40.3	40.3	39.9	39.9
Running Emissions (tons)	VOC	40.854	20.275	20.263	18.507	18.503	19.052	19.055
	CO	868.435	440.677	440.176	414.962	414.665	420.679	420.616
	Nx	186.108	60.344	60.278	31.065	31.044	29.436	29.433
Avg Running Rates (gm/mi)	VOC	0.316	0.136	0.136	0.114	0.114	0.114	0.114
	CO	6.711	2.958	2.958	2.557	2.558	2.507	2.508
	Nx	1.438	0.405	0.405	0.191	0.191	0.175	0.176

Attachment 1

Mobile Emission Post-Processor TP+ Scripts and Mobile 6.2 rates

Used in the

2004 CLRP/ FY 2005-2010 TIP

EMISS.BAT

```
REM  ================================================================================
REM    Mobile Emission Estimation -   Version 2.1/MOBILE6 cogv2.1C                    =
REM    Year:          2005                                                            =
REM    Description:  Travel Model:   V21D#50                                         =
REM                  Land Use:       Rnd 6.4 LU w/ ICC/ CTPP Emp. adjustements       =
REM                  Emission Rates: Mobile 6.2 Case 7 Rates                         =
REM                  Networks:       TIP 2005-10                                     =
REM  ================================================================================

REM  ================================================================
REM  =====   Input File Specification Section                  ========
REM  ================================================================

REM ---'Final' Loaded Network & AM, PM, Off-Peak Trip Tables-----------
REM ---Note:  Include subdirdectory if necessary            -----------

set _inpnet_=I:\CGV2_1D_Conformity2005\2005\i6hwy.net
set _inpamt_=I:\CGV2_1D_Conformity2005\2005\i6am.vtt
set _inppmt_=I:\CGV2_1D_Conformity2005\2005\i6pm.vtt
set _inpopt_=I:\CGV2_1D_Conformity2005\2005\i6op.vtt


REM  MOBILE6 Emission File subdirectory (if needed) & Naming 'Prefix'
set _M6Sub_=I:\CGV2_1D_Conformity2005\Emissions\M6Rates\2005c7\
set _M6Pre_=m605c7

REM  ================================================================
REM  =====   End of Input File Specifications                  ========
REM  ================================================================



REM  ================================================================
REM  =====   Start TP+ program executions                      ========
REM  ================================================================

runtpp AQTRIP.S
runtpp ZONESPRD.S
runtpp Peak_Spread.S
runtpp RUNNING.S
runtpp STRT_SKR.S

REM  ================================================================
REM  =====   End of program executions; clear env. variables   ========
REM  ================================================================

set _inpnet_=
set _inpamt_=
set _inppmt_=
set _inpopt_=
set _M6Sub_=
set _M6Pre_=
```

AQTRIP.S

```
*del tppl*.PRN
; ====================================================================
;  AQTRip.S
;
;  Prepare Vehicle trips tables for AQ emission estimation process
;
; ====================================================================
;---------------------------------------------
; Define Inputs/ Outputs / Global Variable Names/Parameters     -
;---------------------------------------------


;Input Files :
AMTRP      = '%_inpamt_%';  Iter 2 AM Vehicle Trips(spec'd in batch)
PMTRP      = '%_inppmt_%';  Iter 2 PM Vehicle Trips(spec'd in batch)
OPTRP      = '%_inpopt_%';  Iter 2 OP Vehicle Trips(spec'd in batch)

;Output Files :
daytrips   = 'i224hrve.tt'                  ; Balanced 24Hr Veh Trips
tripends   = 'trip.end'                     ; 24hr,am,pm,op Trip-Ends
tripdst    = 'trip.dst'                     ; 27x27 Trip distribution

;Number Of Air Quality Juris./Extl Station TAZ Groups
aqd = '27'

;-------------------------------------------------
;Define the 30 Air Quality Juris-Extl/TAZ Equiv -
;-------------------------------------------------

COPY FILE=AQD_TAZ.EQV
;               ; Jur    Jur Rate Set
;               ; ----   -------------
D  1 =    1- 319 ; DC         DC               1
D  2 =  320- 627 ; Mtg        Mtg              2
D  3 =  640-1020 ; PG         PG               3
D  4 = 1080-1099 ; How        PG               3
D  5 = 1110-1142 ; AA         PG               3
D  6 = 1060-1073 ; Car        PG               3
D  7 = 2172-2191 ; Balt Extls PG               3
D  8 = 1150-1163 ; Cal        Cal              4
D  9 = 1200-1223 ; Chs        Chs              5
D 10 = 1030-1053 ; Frd        Frd              6
D 11 = 2169-2171 ; Frd Extls  Frd              6
D 12 = 1230-1311 ; Arl        Arl              7
D 13 = 1400-1755 ; Ffx        Ffx              8
D 14 = 1780-1905 ; Ldn        Ldn              9
D 15 = 1920-2061 ; PW         PW              10
D 16 = 2080-2093 ; Sta        Sta             11
D 17 = 1330-1389 ; Alx        Alx             12
D 18 = 1170-1190 ; Stm        Stm             13
D 19 = 2164-2168 ; Wash Co Extls  Wash Co     14
D 20 = 2130-2132 ; Clk        Clk             15
D 21 = 2115-2125 ; Fau        Clk             15
D 22 = 2135-2141 ; Jef        Clk             15
D 23 = 2154-2163 ; Clk Extls  Clk             15
D 24 = 2105-2110 ; Spts       Spts            16
D 25 = 2070-2074 ; KG         Spts            16
D 26 = 2100-2101 ; Frdbg      Spts            16
D 27 = 2145-2153 ; Spts Extls Spts            16
;
ENDCOPY
```

```
;-------------------------------------------------------------------
; Program script follows:
;-------------------------------------------------------------------
; Step 1.
; First read in Final Iter 2. Time per.-specific trip tables
; (they are segmented by mode)
; consolidate into daily, AM, PM, and Off-Peak trip tables, write out
;-------------------------------------------------------------------
;
;
RUN PGM=MATRIX                                    ; - - - - - - - - -
  MATI[1]  = @AMTRP@                              ;    AM Trip file
  MATI[2]  = @PMTRP@                              ;    PM Trip file
  MATI[3]  = @OPTRP@                              ;    OP Trip file
  MATO = totdayve.tem, MO=30,31,32,33             ;output trip file
;                                                  4 -tables
;                                                     1) daily(unbalanced)
;                                                     2) AM Total Trips
;                                                     3) PM Total Trips
;                                                     4) OP Total Trips
;     AM             PM             Off-Pk
;--------------  --------------  --------------
 MW[1] = MI.1.1  MW[11] = MI.2.1 MW[21] = MI.3.1  ; <- 1-Occ Vehs
 MW[2] = MI.1.2  MW[12] = MI.2.2 MW[22] = MI.3.2  ; <- 2-Occ Vehs
 MW[3] = MI.1.3  MW[13] = MI.2.3 MW[23] = MI.3.3  ; <- 3+Occ Vehs
 MW[4] = MI.1.4  MW[14] = MI.2.4 MW[24] = MI.3.4  ; <- Medium Heavy Trks
 MW[5] = MI.1.5  MW[15] = MI.2.5 MW[25] = MI.3.5  ; <- Airport Pax ADrs

;
; Consolidate daily, AM,PM,OPk trips here (in mtxs 30,31,32,33)
;
 MW[30] = MW[1] + MW[11] + MW[21] +               ; Total Veh Trips
          MW[2] + MW[12] + MW[22] +               ; in mw[30]
          MW[3] + MW[13] + MW[23] +               ;
          MW[4] + MW[14] + MW[24] +               ;
          MW[5] + MW[15] + MW[25]                 ;

 MW[31] = MW[01] + MW[02] + MW[03] + MW[04] + MW[05] ; Total AM  Trips
 MW[32] = MW[11] + MW[12] + MW[13] + MW[14] + MW[15] ; Total PM  Trips
 MW[33] = MW[21] + MW[22] + MW[23] + MW[24] + MW[25] ; Total OPk Trips

endrun
;============================================================

;-------------------------------------------------------------------
; Step 2.
; Read in Daily(unbalanced),AM,PM,OP veh. trip tab & get X-pose of each.
; write out ASCII zone file listing Daily(balanced), AM,PM,&OP Os/Ds.
; Write out daily balanced trip table
;-------------------------------------------------------------------

RUN PGM=MATRIX
  MATI[1]  = totdayve.tem       ; 24 Hr Veh Trips/ Unbalanced
  MATO     = @daytrips@, MO=10  ; 24 Hr Veh Trips/ Balanced
;
  MW[1] = MI.1.1    ; Total Daily Veh.Trips/Unbalanced
  MW[2] = MI.1.1.T  ; Total Daily Veh.Trips/Unbalanced - Transposed
;
;
  MW[11] = MI.1.2    ;       AM Veh.Trips/Unbalanced
  MW[12] = MI.1.2.T  ;       AM Veh.Trips/Unbalanced - Transposed
;
;
  MW[21] = MI.1.3    ;       PM Veh.Trips/Unbalanced
  MW[22] = MI.1.3.T  ;       PM Veh.Trips/Unbalanced - Transposed
;
;
```

```
  MW[31] = MI.1.4    ;       OP Veh.Trips/Unbalanced
  MW[32] = MI.1.4.T  ; Total OP Veh.Trips/Unbalanced - Transposed
;
;-------------------------------------------------------------------------
; Now, balance the total daily vehicle trip table here                   -
; and then get the row and col totals of the daily, AM, PM, and Off-     -
; peak trips                                                             -
;-------------------------------------------------------------------------
;
  MW[10]= (MW[1] + MW[2]) * 0.50

  RSUM10 = ROWSUM(10) ; Zonal Row Total of daily BALANCED Veh. trips
  RSUM11 = ROWSUM(11) ; Zonal Row Total of AM             Veh. trips
  RSUM12 = ROWSUM(12) ; Zonal Col Total of AM             Veh. trips
  RSUM21 = ROWSUM(21) ; Zonal Row Total of PM             Veh. trips
  RSUM22 = ROWSUM(22) ; Zonal Col Total of PM             Veh. trips
  RSUM31 = ROWSUM(31) ; Zonal Row Total of OffPk          Veh. trips
  RSUM32 = ROWSUM(32) ; Zonal Col Total of OffPk          Veh. trips

  List   = I(5),  rsum10(10.2), rsum10(10.2), ;  Daily Bal O/Ds
                  rsum11(10.2), rsum12(10.2), ;  AM Os/Ds
                  rsum21(10.2), rsum22(10.2), ;  PM Os/Ds
                  rsum31(10.2), rsum32(10.2), ;  OP Os/Ds
                  file=@tripends@

ENDRUN         ; all done
;
;-------------------------------------------------------------------------
; Step 3.
; Squeeze zonal daily balanced trip table to Air Quality Districts
;-------------------------------------------------------------------------

RUN PGM=MATRIX
  MATI[1]    = @daytrips@       ;   24 Hr Veh Trips/ Balanced
  MATO       = sqz.tem,MO=1     ;   24 Hr Veh Trips/ Balanced squeezed
  MW[1] = MI.1.1    ; Total Daily Veh.Trips/Balanced
  RENUMBER FILE=AQD_TAZ.EQV, MISSINGZI=M, MISSINGZO=W
ENDRUN

;
;-------------------------------------------------------------------------
; Step 4.
; Read squeezed district level daily balanced veh. trip table
;-------------------------------------------------------------------------

RUN PGM=MATRIX
ZONES=@AQD@
ARRAY ROWCHKI=@AQD@, ROWADJ=@AQD@, ROWCHKF=@AQD@
  MATI[1]    = sqz.tem          ;
  MATO       = sqzftrxp.tem,mo=3,dec=4
  MW[1] = MI.1.1        ; Total Daily Veh.Trips/Balanced/Squeezed
  ROWSM1     = ROWSUM(1) ; MATRIX ROW Total
JLOOP
  MW[2][J]   = Round(MW[1][J] / ROWSM1 * 1000.00)
ENDJLOOP

  ROWCHKI[i] = 0
  ROWADJ[I]  = 0
JLOOP
  ROWchkI[i] = rowchkI[i] + MW[2][J] ; Initial Check
ENDJLOOP

  ROWADJ[I]     = 1000.0 - ROWCHKI[i]

  ROWCHKF[i]    = 0
JLOOP
   IF (I=J)
      MW[2][J] = MW[2][J] + ROWADJ[I]
   ENDIF
ENDJLOOP

JLOOP
  ROWchkF[i]  = rowchkF[i] + MW[2][J] ; Final   Check
     MW[3][J] = MW[2][J] /1000.0
ENDJLOOP

JLOOP
   LIST =I(5),J(5),MW[1](10),MW[2](10),MW[3](10.4),
         rowchkI[i](6),ROWADJ[i](6),ROWCHKF[i](6),file=distchk.asc
ENDJLOOP


RUN PGM=MATRIX
  MATI  = sqzftrxp.tem
  MW[1] = mi.1.1.T
  PRINT FORM=6.3,
  LIST  =i(5),' ',
  mw[1][01],' ',mw[1][02],' ',mw[1][03],' ',mw[1][04],' ',
  mw[1][05],' ',mw[1][06],' ',mw[1][07],' ',mw[1][08],' ',
  mw[1][09],' ',mw[1][10],' ',mw[1][11],' ',mw[1][12],' ',
  mw[1][13],' ',mw[1][14],' ',mw[1][15],' ',mw[1][16],' ',
  mw[1][17],' ',mw[1][18],' ',mw[1][19],' ',mw[1][20],' ',
  mw[1][21],' ',mw[1][22],' ',mw[1][23],' ',mw[1][24],' ',
  mw[1][25],' ',mw[1][26],' ',mw[1][27],
  file=@tripdst@
ENDRUN

*copy tppl*.PRN aqtrip.rpt
```

Peak_Spread.s

```
*del tppl*.prn
;--------------------------------------------------------------------
;Peak_Spread.s - Post-traffic assignment process to develop hourly
;               speeds and volumes.  The hourly speeds will be used
;               to recompute AM, PM and Off-Peak Period Speeds which
;               will be recycled back to the trip distribution process
;
;                The program was built of of PEAKTST4.S which was one
;                of many V2.1C post-processor alternative tests explored by
;                Milone&Kirby in the Fall of 2003 for the TRB Model Review
;                committee.
;
;               This run includes special freeway volume overides based
;               on skycomp-based volume flow volumes for congested links
;
;  ----------
;       Changes per V2.1 D Draft 50 Model:
;         1.   Speed is NOT the minimum of pre-/post- spreaded hrly volumes, as before.
;              Speed is based on the initial volume only.
;         2.   Frwy speed-flow curves now have w/ 16 mph floor for freeways (not 8mph as
before).
;         3.   Original 'initial' hrly distributions adjusted so that
;              peaking assumptions used in traffic assignment are maintained.
;         4.   The time period volumes from the traffic assignment are 'conserved'.
;              i.e., The initial hourly volumes are adjusted to match assignment loads.
;         5.   Non-Freeway Speed flow curves updated.
;--------------------------------------------------------------------

; Define Inputs/Outputs:


; Step 1. Accumulate regional AM&PM highway volume.
;         The AM/PM totals will be stored in temporary variables
;         _AMVOL and _PMVOL.

run pgm=hwynet
 neti     = %_inpnet_%
 neto     = dummy.net,include = a,b,i6amvol,i6pmvol

 if (a>2191 &&  b>2191)
 _AMVOL = _AMVOL + i6amvol
 _PMVOL = _PMVOL + i6pmvol
endif

 LOG PREFIX=HWYNET , VAR =_AMVOL, _PMVOL ; pass regional AM/PM volumes figures
endrun                                   ; out to TP MAIN as HWYNET._AMVOL & HWYNET._PMVOL
*del dummy.net

;-----------------------------------------------------------------
; Step 2. (TP_MAIN)
;      -  Calculated ratio of regional PM volume to AM volume
;         in variable: ftram
;

amtot=hwynet._amvol
pmtot=hwynet._pmvol
ftram=pmtot/amtot

; Step 3 (TP+/hwynet)
;      -  Apply ratio (ftram) to AM volume to develop scaled AM Volume
;      -  Identify AM/PM HOV link using a 0/1 'flags'(AMHOVFLG/PMHOVFLG)
;         These links must be identified so that they will be allocated
;         to hourly periods using special hourly distributions.

run pgm=hwynet

 neti = %_inpnet_%
 neto = temp.net,include = a,b,distance,spdclass,capclass,ftype,amhtime,
        amlane,pmlane,oplane,i6amvol,i6pmvol,i6opvol,i624vol,jur,atype,
        i6amspd,i6pmspd,i6opspd,
        amhovflg,pmhovflg,ftramvol

        amhovflg = 0
        pmhovflg = 0
         if (i6amvol = i624vol) amhovflg = 1
         if (i6pmvol = i624vol) pmhovflg = 1

        ftramvol = i6amvol * @ftram@ ; ftramvol is the scaled am vol so that the regional
total
endrun                                   ; equals the pm total.  The scaled vol is used to
develop a balanced
                                         ; distribution of am,pm peaking class codes for
the system below.
                                         ; It is not used for any other purpose
;-----------------------------------------------------------------
; Step 4. (TP+/hwynet)
;         Now develop final hourly VMT and Speeds
;

run pgm=hwyload
  neti = temp.net


; Set up arrays

ARRAY,
CAPA=24,   ; hourly link capacities
IVOL=24,   ; initial hrly volumes based on table lookup distributions
IVCR=24,   ; Initial V/C ratio
IVMT=24,   ; Initial VMT
ISPD=24,   ; Initial Restrained Hrly Speed

XVOL=24,   ; excess hrly volume- portion of initial volume exceeding capacity
NVOL=24,   ; 'New'  hrly volume- initial volume+excess from adjacent hr(s)
FVOL=24,   ; Final  hrly volume- final hourly volume (after volume spreading)
NVCR=24,   ; New    V/C ratio
NSPD=24,   ; New    Restrained Hrly Speed

FSPD=24,   ; Final Restrained Hrly Speed
DSPD=24,   ; Difference between Initial and Final Speed
LANE=24,   ; lanes
FVMT=24    ; Final  VMT


;--------------------------------------------------------------------------
; NOTE: The hourly arrays are indexed from 1 to 24, as follows:
; 1-  12mid - 12:59AM       13- 12noon - 12:59PM
; 2- 1:00AM -  1:59AM       14- 1:00PM -  1:59PM
; 3- 2:00AM -  2:59AM       15- 2:00PM -  2:59PM
; 4- 3:00AM -  3:59AM       16- 3:00PM -  3:59PM
; 5- 4:00AM -  4:59AM       17- 4:00PM -  4:59PM
; 6- 5:00AM -  5:59AM       18- 5:00PM -  5:59PM
; 7- 6:00AM -  6:59AM       19- 6:00PM -  6:59PM
; 8- 7:00AM -  7:59AM       20- 7:00PM -  7:59PM
; 9- 8:00AM -  8:59AM       21- 8:00PM -  8:59PM
;10- 9:00AM -  9:59AM       22- 9:00PM -  9:59PM
;11-10:00AM - 10:59AM       23-10:00PM - 10:59PM
;12-11:00AM - 11:59AM       24-11:00PM - 11:59PM
;--------------------------------------------------------------------------
```

```
;
;---------------------------------------------------------------------
; Initial Hourly traffic distribution LOOKUP for NON-HOV Links        -
;---------------------------------------------------------------------

;Establish 9 lookup Hrly distributions
;Curves taken from M.Freeman's 8/27/02 Memo to File (Figure 5)
;Note: Hours range from 1-24, instead of 0-23
;      Pcts of last hour (24) may be adjusted slightly to force normalization
; Updated 8/25 by milone, moran
; so that assignment&peak spreader use consistent peaking assumptions
LOOKUP NAME=HR_DIST,           ; <<-- HRLY distributions by peak type&Ft
       LOOKUP[1]=1, RESULT=2,  ; AM / Freeway
       LOOKUP[2]=1, RESULT=3,  ; AM / Arterial
       LOOKUP[3]=1, RESULT=4,  ; AM / Local/Collector

       LOOKUP[4]=1, RESULT=5,  ; PM / Freeway
       LOOKUP[5]=1, RESULT=6,  ; PM / Arterial
       LOOKUP[6]=1, RESULT=7,  ; PM / Local/Collector

       LOOKUP[7]=1, RESULT=8,  ; Even / Freeway
       LOOKUP[8]=1, RESULT=9,  ; Even / Arterial
       LOOKUP[9]=1, RESULT=10, ; Even / Local/Collector
       INTERPOLATE = N,LIST=T,
R= " 1  0.767  0.490  0.337  1.113  0.760  0.621  1.073  0.671  0.520",
   " 2  0.548  0.301  0.195  0.644  0.408  0.323  0.728  0.396  0.305",
   " 3  0.517  0.250  0.176  0.478  0.281  0.242  0.606  0.301  0.238",
   " 4  0.717  0.374  0.285  0.420  0.238  0.199  0.680  0.330  0.301",
   " 5  1.879  1.092  0.958  0.581  0.377  0.321  1.235  0.719  0.703",
   " 6  6.199  4.052  3.801  1.379  1.075  0.962  3.602  2.272  2.369",
   " 7  8.662  8.748  9.188  3.238  2.699  2.583  4.991  4.578  4.832",
   " 8 11.133 12.380 13.403  4.629  4.618  4.667  6.955  7.651  8.064",
   " 9  8.038  9.822 10.917  4.711  5.148  5.068  5.442  6.899  7.265",
   "10  6.937  6.386  6.095  3.836  4.376  4.096  5.927  6.112  5.800",
   "11  5.138  4.707  4.495  3.898  4.192  3.938  5.176  5.153  4.802",
   "12  4.679  4.531  4.509  4.208  4.674  4.543  5.148  5.404  5.141",
   "13  4.653  4.717  4.809  4.611  5.252  5.247  5.340  5.803  5.496",
   "14  4.584  4.639  4.635  4.832  5.210  5.007  5.450  5.678  5.336",
   "15  4.662  4.802  4.845  5.950  5.874  5.763  6.102  5.971  5.890",
   "16  4.703  5.092  5.167  7.319  7.139  7.034  6.801  6.619  6.679",
   "17  4.558  5.259  5.238  9.950  9.580 10.056  5.944  6.258  6.610",
   "18  4.763  5.545  5.584 10.866 10.929 11.574  6.631  7.153  7.661",
   "19  4.324  4.983  4.919  8.551  9.029  9.651  5.347  5.921  6.435",
   "20  3.655  3.903  3.716  5.613  6.012  6.167  4.994  5.294  5.452",
   "21  2.950  2.965  2.698  4.246  4.436  4.597  3.885  4.047  4.088",
   "22  2.644  2.403  2.010  3.682  3.579  3.518  3.441  3.269  3.061",
   "23  2.058  1.639  1.297  2.798  2.409  2.203  2.696  2.212  1.901",
   "24  1.232  0.920  0.723  2.447  1.705  1.620  1.806  1.289  1.052"

;;; Original / Fall 2003 Hourly Curves - Superseded -
;; " 1, 0.758, 0.490, 0.337, 1.169, 0.780, 0.645, 1.022, 0.642, 0.507",
;; " 2, 0.541, 0.301, 0.195, 0.677, 0.419, 0.336, 0.693, 0.379, 0.297",
;; " 3, 0.511, 0.250, 0.176, 0.502, 0.288, 0.251, 0.577, 0.288, 0.232",
;; " 4, 0.708, 0.374, 0.285, 0.441, 0.244, 0.207, 0.648, 0.316, 0.294",
;; " 5, 1.856, 1.091, 0.959, 0.610, 0.387, 0.334, 1.176, 0.688, 0.686",
;; " 6, 6.123, 4.048, 3.804, 1.449, 1.103, 1.000, 3.431, 2.175, 2.310",
;; " 7, 9.488, 9.020, 9.206, 3.011, 2.603, 2.452, 5.805, 5.069, 5.153",
;; " 8,10.022,11.834,13.334, 4.305, 4.454, 4.430, 6.769, 7.724, 7.985",
;; " 9, 8.804,10.128,10.938, 4.381, 4.965, 4.811, 6.330, 7.639, 7.748",
;; "10, 6.735, 6.373, 6.106, 4.030, 4.489, 4.257, 5.645, 5.852, 5.656",
;; "11, 5.075, 4.703, 4.499, 4.095, 4.300, 4.093, 4.930, 4.933, 4.683",
;; "12, 4.622, 4.527, 4.513, 4.421, 4.794, 4.721, 4.903, 5.174, 5.013",
;; "13, 4.596, 4.713, 4.813, 4.844, 5.387, 5.453, 5.086, 5.556, 5.359",
;; "14, 4.528, 4.635, 4.639, 5.076, 5.344, 5.204, 5.191, 5.436, 5.203",
;; "15, 4.605, 4.798, 4.849, 6.251, 6.025, 5.989, 5.812, 5.717, 5.743",
;; "16, 4.645, 5.087, 5.171, 8.268, 7.593, 7.701, 6.477, 6.337, 6.513",
;; "17, 4.637, 5.265, 5.233, 9.493, 9.303, 9.537, 6.702, 6.859, 7.033",
;; "18, 4.845, 5.551, 5.579, 9.659,10.417,11.003, 6.753, 7.301, 7.571",
;; "19, 4.399, 4.988, 4.915, 8.159, 8.768, 9.153, 6.029, 6.490, 6.847",
;; "20, 3.610, 3.899, 3.719, 5.897, 6.167, 6.409, 4.756, 5.068, 5.316",
;; "21, 2.914, 2.962, 2.700, 4.461, 4.550, 4.778, 3.700, 3.875, 3.986",
;; "22, 2.612, 2.401, 2.012, 3.868, 3.671, 3.656, 3.277, 3.130, 2.985",
;; "23, 2.033, 1.637, 1.298, 2.939, 2.471, 2.290, 2.568, 2.118, 1.854",
;; "24, 1.333, 0.925, 0.720, 1.994, 1.478, 1.290, 1.720, 1.234, 1.026"
;;;

;---------------------------------------------------------------------
;  End of Hourly non-HOV traffic distribution LOOKUP                  -
;---------------------------------------------------------------------
;---------------------------------------------------------------------
; Initial Hourly traffic distribution LOOKUP for Special HOV Links    -
;---------------------------------------------------------------------

;Establish 9 lookup Hrly distributions
;Curves taken from Fall 2002 HOV traffic counts provided by Pat Zilliacus
; provide in Mar 2003 (undocumented).
;
;

LOOKUP NAME=HR_HOVD,           ; <<-- HOV HRLY distributions by type
       LOOKUP[1]=1, RESULT=2,  ; AM prd HOV Operation on Freeways
       LOOKUP[2]=1, RESULT=3,  ; PM prd HOV Operation on Freeway
       LOOKUP[3]=1, RESULT=4,  ; unused / for future use
       LOOKUP[4]=1, RESULT=5,  ; unused / for future use
       LOOKUP[5]=1, RESULT=6,  ; unused / for future use
       LOOKUP[6]=1, RESULT=7,  ; unused / for future use
       LOOKUP[7]=1, RESULT=8,  ; unused / for future use
       LOOKUP[8]=1, RESULT=9,  ; unused / for future use
       LOOKUP[9]=1, RESULT=10, ; unused / for future use
       INTERPOLATE = N,LIST=T,

R= " 1, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   " 2, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   " 3, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   " 4, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   " 5, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   " 6, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   " 7,29.700, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   " 8,43.300, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   " 9,27.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "10, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "11, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "12, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "13, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "14, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "15, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "16, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "17, 0.000,37.300, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "18, 0.000,41.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "19, 0.000,21.700, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "20, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "21, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "22, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "23, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000",
   "24, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000"
;---------------------------------------------------------------------
;  End of HOV Hourly traffic distribution LOOKUP                      -
;---------------------------------------------------------------------

;---------------------------------------------------------------------
; Speed Lookup tables - Speed is a function of V/C                    -

LOOKUP NAME = SPDA,
       LOOKUP[1] = 1,RESULT=2,   ; FWY/RMP AT=1-2
       LOOKUP[2] = 1,RESULT=3,   ; FWY/RMP AT=3-4
       LOOKUP[3] = 1,RESULT=4,   ; FWY/RMP AT=5-7
```

```
      LOOKUP[4] = 1,RESULT=5,   ; MAJ AT=1-2
      LOOKUP[5] = 1,RESULT=6,   ; MAJ AT=3-4
      LOOKUP[6] = 1,RESULT=7,   ; MAJ AT=5
      LOOKUP[7] = 1,RESULT=8,   ; MAJ AT=6-7
      LOOKUP[8] = 1,RESULT=9,   ; MIN AT=1-2
      LOOKUP[9] = 1,RESULT=10,  ; MIN AT=3-4
      LOOKUP[10]= 1,RESULT=11,  ; MIN AT=5
      LOOKUP[11]= 1,RESULT=12,  ; MIN AT=6-7
      LOOKUP[12]= 1,RESULT=13,  ; COL AT=1-2
      LOOKUP[13]= 1,RESULT=14,  ; COL AT=3-4
      LOOKUP[14]= 1,RESULT=15,  ; COL AT=5
      LOOKUP[15]= 1,RESULT=16,  ; COL AT=6-7
      LOOKUP[16]= 1,RESULT=17,  ; EXP AT=1-2
      LOOKUP[17]= 1,RESULT=18,  ; EXP AT=3-5
      LOOKUP[18]= 1,RESULT=19,  ; EXP AT=6-7
  INTERPOLATE = Y, LIST=T,
; UPDATED 9/23/04 Speed Flow Curves - sivasailam/milone
;
; V/C     |----   Freeway  ---||-----    Maj Art    ------||------   Minor Art  -----||-----
-  Collector ---------||----    Expr  -----|
```

```
;   Atp->  1-2    3-4    5-7    1-2    3-4    5      6-7    1-2    3-4      5      6-7    1-2   3-4     5      6-7   1-2   3-5    6-7
;          -----  ------ ------ ------ ------ ------ ------ ------ ------ ------ ------ ------ ------ ------ ------  -----  ------ ------
R=" 0.00  55.000 60.000 67.000 25.000 35.000 40.000 45.000 20.000 30.000 35.000 40.000 15.000 20.000 25.000 30.000 45.000 50.000 55.000",
  " 0.10  54.945 59.945 66.940 24.300 33.600 38.600 43.600 19.400 28.900 33.800 39.200 14.500 19.300 24.300 28.800 44.945 49.939 54.933",
  " 0.20  54.890 59.890 66.880 23.600 32.200 37.200 42.200 18.800 27.800 32.600 38.400 14.000 18.600 23.600 27.600 44.890 49.878 54.866",
  " 0.30  54.810 59.800 66.790 22.900 30.800 35.800 40.800 18.200 26.700 31.400 37.600 13.500 17.900 22.900 26.400 44.820 49.800 54.780",
  " 0.40  54.710 59.690 66.670 22.200 29.400 34.400 39.400 17.600 25.600 30.200 36.800 13.000 17.200 22.200 25.200 44.730 49.700 54.670",
  " 0.50  54.570 59.540 66.490 21.500 28.000 33.000 38.000 17.000 24.500 29.000 36.000 12.500 16.500 21.500 24.000 44.620 49.578 54.536",
  " 0.60  54.370 59.300 66.180 20.800 27.000 31.600 36.400 16.400 23.400 27.800 35.000 12.000 16.000 20.800 23.000 44.470 49.411 54.352",
  " 0.70  54.060 58.910 65.600 20.100 26.000 30.200 34.800 15.800 22.300 26.600 34.100 11.500 15.500 20.100 22.000 44.260 49.178 54.096",
  " 0.80  53.540 58.170 64.260 19.400 25.000 28.800 33.200 15.200 21.200 25.400 31.400 11.000 15.000 19.400 21.000 43.970 48.856 53.741",
  " 0.90  52.560 56.560 60.840 18.700 24.000 27.400 31.600 14.600 20.100 24.200 28.700 10.500 14.500 18.700 20.000 43.530 48.367 53.203",
  " 1.00  50.580 53.220 55.280 18.000 23.000 26.000 30.000 14.000 19.000 23.000 26.000 10.000 14.000 18.000 19.000 42.820 47.578 52.336",
  " 1.10  46.860 48.550 49.875 16.600 20.800 23.400 27.200 12.800 17.600 21.000 23.600  9.200 12.800 16.600 17.600 41.250 45.833 50.417",
  " 1.17  44.256 45.281 46.092 15.620 19.260 21.580 25.240 11.960 16.620 19.600 21.920  8.640 11.960 15.620 16.620 40.151 44.612 49.073",
  " 1.20  43.140 43.880 44.470 15.200 18.600 20.800 24.400 11.600 16.200 19.000 21.200  8.400 11.600 15.200 16.200 39.680 44.089 48.498",
  " 1.30  39.335 39.870 40.315 13.800 16.400 18.200 21.600 10.400 14.800 17.000 18.800  7.600 10.400 13.800 14.800 36.925 41.028 45.131",
  " 1.40  35.530 35.860 36.160 12.400 14.200 15.600 18.800  9.200 13.400 15.000 16.400  6.800  9.200 12.400 13.400 34.170 37.967 41.763",
  " 1.50  32.470 32.740 32.990 11.000 12.000 13.000 16.000  8.000 12.000 13.000 14.000  6.000  8.000 11.000 12.000 31.420 34.911 38.402",
  " 1.60  29.410 29.620 29.820 11.000 12.000 13.000 16.000  8.000 12.000 13.000 14.000  6.000  8.000 11.000 12.000 28.670 31.856 35.041",
  " 1.80  24.550 24.700 24.850 11.000 12.000 13.000 16.000  8.000 12.000 13.000 14.000  6.000  8.000 11.000 12.000 24.050 26.722 29.394",
  " 2.00  20.610 20.730 20.860 11.000 12.000 13.000 16.000  8.000 12.000 13.000 14.000  6.000  8.000 11.000 12.000 20.230 22.478 24.726",
  " 2.25  16.650 16.750 16.850 11.000 12.000 13.000 16.000  8.000 12.000 13.000 14.000  6.000  8.000 11.000 12.000 16.350 18.167 19.983",
 "99.99  16.650 16.750 16.850 11.000 12.000 13.000 16.000  8.000 12.000 13.000 14.000  6.000  8.000 11.000 12.000 16.350 18.167 19.983"
;
;
;
;
;>>
;-------------------------------------------------------------------------
; Added Lookup - Freeway Volume Flow (v/hr/ln) as a function of V/C   -
;       (D.Sivasailam 12/9/03).                                       -
;       Note: Volume flows on freeways for congested conditions       -
;        -  these values will be used to develop override hrly        -
;           freeway link volumes where V/C > 1.0                      -
;-------------------------------------------------------------------------
LOOKUP NAME = CFWYFLWA,         ; congested freeway flow array
       LOOKUP[1]=1, RESULT=2,   ;         Area Type 1
       LOOKUP[2]=1, RESULT=3,   ;         Area Type 2
       LOOKUP[3]=1, RESULT=4,   ;         Area Type 3
       LOOKUP[4]=1, RESULT=5,   ;         Area Type 4
       LOOKUP[5]=1, RESULT=6,   ;         Area Type 5
       LOOKUP[6]=1, RESULT=7,   ;         Area Type 6
       LOOKUP[7]=1, RESULT=8,   ;         Area Type 7
       INTERPOLATE=Y, LIST=T,   ; Interpolate Between Points, List out
;
; LOOKUP VALUES FOLLOW - UNITS: Vehicles Per Hour Per Lane (VPHPL)
;
;          Fwy   Fwy   Fwy   Fwy   Fwy   FWY   FWY
;    V/C   AT1   AT2   AT3   AT4   AT5   AT6   AT7
;   ----- ----- ----- ----- ----- ----- ----- -----
R= " 1.00  1500  1600  1800  1800  2000  2000  2100 ",
   " 1.20  1433  1528  1719  1719  1911  1911  2006 ",
   " 1.40  1366  1457  1639  1639  1821  1821  1912 ",
   " 1.60  1299  1385  1559  1559  1732  1732  1818 ",
   " 1.80  1214  1295  1457  1457  1619  1619  1699 ",
   " 2.00  1128  1204  1355  1355  1505  1505  1580 ",
   " 2.25  1017  1085  1221  1221  1356  1356  1424 ",
   "99.99  1017  1085  1221  1221  1356  1356  1424 "
;>>
;-------------------------------------------------------------------------
; End of Speed Lookup tables                                            -
;-------------------------------------------------------------------------

zones = 1

phase = linkread
  t0          = li.amhtime

;
  lw.amcap    = capacityfor(li.amlane,li.capclass)
  lw.pmcap    = capacityfor(li.pmlane,li.capclass)
```

```
  lw.opcap    = capacityfor(li.oplane,li.capclass)

  lw.amlane   = li.amlane
  lw.pmlane   = li.pmlane
  lw.oplane   = li.oplane

  lw.ffspd    = speedfor(1,li.spdclass)
  lw.distance = li.distance

  lw.amspd    = li.i6amspd
  lw.pmspd    = li.i6pmspd
  lw.opspd    = li.i6opspd

  lw.amvol    = li.i6amvol
  lw.pmvol    = li.i6pmvol
  lw.opvol    = li.i6opvol
  lw.totvol   = li.i624vol

  lw.amvmt    = lw.amvol*lw.distance
  lw.pmvmt    = lw.pmvol*lw.distance
  lw.opvmt    = lw.opvol*lw.distance

  lw.ftramvol = li.ftramvol

  lw.ftype    = li.ftype
  lw.atype    = li.atype
  lw.jur      = li.jur
  lw.amhovflg = li.amhovflg
  lw.pmhovflg = li.pmhovflg


endphase

phase = iloop
;
      linkloop               ;             select -
  if (lw.ftype>0  )  ;  <<---      ALL HIGHWAY links

;-------------------------------------------------------------------------
;  Define peaking class 1, 2, or 3 (AM, PM, or Even)                    -
;  Note AM,PM tolerences originally set to +/-7.5% changed to +/- 10%   -
;-------------------------------------------------------------------------
;

if (lw.totvol = 0)
    pkdfpct  = 0
else
    pkdfpct  = ((lw.ftramvol - lw.pmvol) / lw.totvol) * 100.00
endif

if     (pkdfpct >   7.50)
        pkclass = 1 ; AM  type peaking
elseif (pkdfpct <  -7.50)
        pkclass = 2; PM   type peaking
else
        pkclass = 3; Even type peaking
endif


;
;-------------------------------------------------------------------------
;  Define Facility Class 1, 2, or 3 (freeway, arterial, collector/local) -
;-------------------------------------------------------------------------
;

if     (lw.ftype = 1 || lw.ftype = 5 || lw.ftype = 6)
        ftclass = 1                   ; Freeway/Expresway/Ramp
elseif (lw.ftype = 2 || lw.ftype = 3)
        ftclass = 2                   ; Arterial
else
        ftclass = 3                   ; local
endif


;
;-------------------------------------------------------------------------
;  Define Joint Peak / Facility (JPF) Class 1-9
;-------------------------------------------------------------------------
;

if (pkclass = 1 && ftclass = 1)  jpf = 1 ; AM/freeway
if (pkclass = 1 && ftclass = 2)  jpf = 2 ; AM/arterial
if (pkclass = 1 && ftclass = 3)  jpf = 3 ; AM/collector/local

if (pkclass = 2 && ftclass = 1)  jpf = 4 ; PM/freeway
if (pkclass = 2 && ftclass = 2)  jpf = 5 ; PM/arterial
if (pkclass = 2 && ftclass = 3)  jpf = 6 ; PM/collector/local

if (pkclass = 3 && ftclass = 1)  jpf = 7 ; Even/freeway
if (pkclass = 3 && ftclass = 2)  jpf = 8 ; Even/arterial
if (pkclass = 3 && ftclass = 3)  jpf = 9 ; Even/collector/local

;
;-------------------------------------------------------------------------
; Now apply the appropriate initial distributions to daily volume       -
; and compute initial capacities to develop IVOL & CAPA hourly arrays   -
; Note:  Hourly AM,PM,Off-Pk volumes may not sum to period volumes in    -
; the loaded links file. reconciliation will occur following the initial -
; allocation of hourly volumes  IVOL[1] ... IVOL[24]                    -
;-------------------------------------------------------------------------
;
;
LOOP IDX=1,24
 IF (lw.amhovflg = 0 && lw.pmhovflg =0 ) ; if link is operating all day...
   IVOL[idx] = (HR_DIST(jpf,idx) * lw.totvol)/100.0 ; initial hrly volume
 ENDIF

 IF (lw.amhovflg = 1 )              ; if link operates during AM hrs...
   IVOL[idx] = (HR_HOVD(1,idx) * lw.totvol)/100.0 ; initial hrly volume
 ENDIF

 IF (lw.pmhovflg = 1)               ; if link operates during PM hrs...
   IVOL[idx] = (HR_HOVD(2,idx) * lw.totvol)/100.0 ; initial hrly volume
 ENDIF
                                                  ;
 ; Define Hourly lanes and capacities
     if     (idx >   6 && idx < 10)              ; compute capacity
        capa[idx] = lw.amcap                      ; based on time slot
        lane[idx] = lw.amlane                     ;;> am lanes

      elseif (idx > 16 && idx < 20)
        capa[idx] = lw.pmcap
        lane[idx] = lw.pmlane                     ;;> pm lanes
      else
        capa[idx] = lw.opcap
        lane[idx] = lw.oplane                     ;;> op lanes
     endif
ENDLOOP


;-------------------------------------------------------------------------
; Next factor the link hourly volumes to ensure they are equal to the    -
; the period volumes of the input loaded links file.                     -
;                                                                        -
;-------------------------------------------------------------------------
```

```
 _iniamvol  = IVOL[7]   + IVOL[8]  + IVOL[9]

 _inipmvol  = IVOL[17]  + IVOL[18] + IVOL[19]

 _iniopvol  = lw.totvol - (_iniamvol + _inipmvol)


LOOP IDX=1,24
   IF      (IDX =  7, 8, 9 && _iniamvol > 0)
     IVOL[idx] = IVOL[idx] *  (lw.amvol/_iniamvol)

    ELSEIF (IDX = 17,18,19 && _inipmvol > 0)
     IVOL[idx] = IVOL[idx] *  (lw.pmvol/_inipmvol)

    ELSEIF (_iniopvol > 0)
     IVOL[idx] = IVOL[idx] * ( (lw.totvol-(lw.amvol+lw.pmvol)) / _iniopvol )
   ENDIF


   ; Compute the Initial VMT
   ;
     IVMT[IDX]  = IVOL[IDX]* lw.DISTANCE    ;Final VMT = Final/Spread Vol
                                            ;      * Distance


   ;
   ; Compute the Initial V/C
   ;
       IVCR[IDX] =(IVOL[IDX] / CAPA[IDX])
   ;
   ; Compute the INITIAL Speed based on INITIAL V/C
   ; Lookup Curves based on Facility type and Area Type
         IF       (lw.ftype = 1,6 && lw.atype = 1,2)
             ISPD[IDX] = SPDA(1,IVCR[IDX])
         ENDIF
         IF       (lw.ftype = 1,6 && lw.atype = 3,4)
             ISPD[IDX] = SPDA(2,IVCR[IDX])
         ENDIF
         IF       (lw.ftype = 1,6 && lw.atype = 5-7)
             ISPD[IDX] = SPDA(3,IVCR[IDX])
         ENDIF

         IF       (lw.ftype = 2 && lw.atype = 1-2)
             ISPD[IDX] = SPDA(4,IVCR[IDX])
         ENDIF
         IF       (lw.ftype = 2 && lw.atype = 3-4)
             ISPD[IDX] = SPDA(5,IVCR[IDX])
         ENDIF
         IF       (lw.ftype = 2 && lw.atype = 5)
             ISPD[IDX] = SPDA(6,IVCR[IDX])
         ENDIF
         IF       (lw.ftype = 2 && lw.atype = 6-7)
             ISPD[IDX] = SPDA(7,IVCR[IDX])
         ENDIF

         IF       (lw.ftype = 3 && lw.atype = 1-2)
             ISPD[IDX] = SPDA(8,IVCR[IDX])
         ENDIF
         IF       (lw.ftype = 3 && lw.atype = 3-4)
             ISPD[IDX] = SPDA(9,IVCR[IDX])
         ENDIF
         IF       (lw.ftype = 3 && lw.atype = 5)
             ISPD[IDX] = SPDA(10,IVCR[IDX])
         ENDIF
         IF       (lw.ftype = 3 && lw.atype = 6-7)
             ISPD[IDX] = SPDA(11,IVCR[IDX])
         ENDIF

         IF       (lw.ftype = 4 && lw.atype = 1-2)
             ISPD[IDX] = SPDA(12,IVCR[IDX])
         ENDIF
         IF       (lw.ftype = 4 && lw.atype = 3-4)
             ISPD[IDX] = SPDA(13,IVCR[IDX])
         ENDIF
         IF       (lw.ftype = 4 && lw.atype = 5)
             ISPD[IDX] = SPDA(14,IVCR[IDX])
         ENDIF
         IF       (lw.ftype = 4 && lw.atype = 6-7)
             ISPD[IDX] = SPDA(15,IVCR[IDX])
         ENDIF


         IF       (lw.ftype = 5 && lw.atype = 1-2)
             ISPD[IDX] = SPDA(16,IVCR[IDX])
         ENDIF
         IF       (lw.ftype = 5 && lw.atype = 3-5)
             ISPD[IDX] = SPDA(17,IVCR[IDX])
         ENDIF
         IF       (lw.ftype = 5 && lw.atype = 6-7)
             ISPD[IDX] = SPDA(18,IVCR[IDX])
         ENDIF

ENDLOOP

;--------------------------------------------------------------------
; Now, invoke peak spreading routine to develop 'final' volumes     -
;--------------------------------------------------------------------

NVOL[18]=IVOL[18] ; Hour 18 (PM Peak hour 5-6PM)
NVOL[17]=IVOL[17] ;
NVOL[19]=IVOL[19] ;
FVOL[18]=IVOL[18] ;               spread to
   IF (NVOL[18] > CAPA[18])     ; shoulder hrs (17/19)
      FVOL[18] = CAPA[18]
;;>                    --  freeway override --
       if (lw.ftype=1)
          FVOL[18] = CFWYFLWA(lw.atype,IVCR[18]) * lane[18]
       endif
;;>
       XVOL[18] = NVOL[18] - FVOL[18] ; -- a 50/50 split
       NVOL[17] = IVOL[17] + (XVOL[18]*0.50)
       NVOL[19] = IVOL[19] + (XVOL[18]*0.50) ;
   ENDIF


NVOL[8]=IVOL[8]                     ; Hour 8 (AM Peak hour 7-8AM)
NVOL[7]=IVOL[7]                                 ;
NVOL[9]=IVOL[9]                                 ;
FVOL[8]=IVOL[8]                                 ; spread to
    IF (NVOL[8] > CAPA[8])                      ; shoulder hrs (7/9)
       FVOL[8] = CAPA[8]
;;>                    --  freeway override --
       if (lw.ftype=1)
          FVOL[8] = CFWYFLWA(lw.atype,IVCR[8]) * lane[8]
       endif
;;>
       XVOL[8] = NVOL[8] - FVOL[8]              ; -- a 50/50 split
           NVOL[7] = IVOL[7] + (XVOL[8]*0.50)   ;
           NVOL[9] = IVOL[9] + (XVOL[8]*0.50)   ;
    ENDIF

LOOP IDX = 7,2,-1                               ; Pre- AM Peak
     FVOL[IDX] = NVOL[IDX]                      ; Spread (hrs 7-2)
         IF (NVOL[IDX] > CAPA[IDX])             ;
             FVOL[IDX] = CAPA[IDX]
```

```
;;>                    --  freeway override --
        if (lw.ftype=1)
           FVOL[IDX] = CFWYFLWA(lw.atype,IVCR[IDX]) * lane[idx]
        endif
;;>
            XVOL[IDX] = NVOL[IDX] - FVOL[IDX]      ;
         ENDIF
            TMP=(IDX-1)                            ;
            NVOL[TMP] = IVOL[TMP]+XVOL[IDX]        ;
ENDLOOP
     NVOL[1] = IVOL[1] + XVOL[2]                   ; Hour 1 / No spread
     FVOL[1] = IVOL[1] + XVOL[2]                   ; Hour 1 / No spread
                                                   ; even if > capacity


LOOP IDX = 9,12                                    ; Post-AM Peak
     FVOL[IDX] = NVOL[IDX]                         ; Spread (hrs 9-12)
         IF (NVOL[IDX] > CAPA[IDX])                ;
            FVOL[IDX] = CAPA[IDX]
;;>                    --  freeway override --
        if (lw.ftype=1)
           FVOL[IDX] = CFWYFLWA(lw.atype,IVCR[IDX]) * lane[idx]
        endif
;;>
            XVOL[IDX] = NVOL[IDX] - FVOL[IDX]      ;
         ENDIF
            TMP=(IDX+1)                            ;
            NVOL[TMP] = IVOL[TMP]+XVOL[IDX]        ;
ENDLOOP



LOOP IDX = 17,14,-1                                ; Pre- PM Peak
     FVOL[IDX] = NVOL[IDX]                         ; Spread (hrs 17-14)
         IF (NVOL[IDX] > CAPA[IDX])                ;
            FVOL[IDX] = CAPA[IDX]
;;>                    --  freeway override --
        if (lw.ftype=1)
           FVOL[IDX] = CFWYFLWA(lw.atype,IVCR[IDX]) * lane[idx]
        endif
;;>
            XVOL[IDX] = NVOL[IDX] - FVOL[IDX]      ;
         ENDIF
            TMP=(IDX-1)                            ;
            NVOL[TMP] = IVOL[TMP]+XVOL[IDX]        ;
ENDLOOP
     NVOL[13] = IVOL[13] + XVOL[12] + XVOL[14]     ; Hour 13 (Noon Hr)
     FVOL[13] = IVOL[13] + XVOL[12] + XVOL[14]     ; Hour 13 (Noon Hr)
                                                   ; No Spread


LOOP IDX = 19,23                                   ; Post-PM Peak
     FVOL[IDX] = NVOL[IDX]                         ; Spread (hrs 19-23)
         IF (NVOL[IDX] > CAPA[IDX])                ;
            FVOL[IDX] = CAPA[IDX]
;;>                    --  freeway override --
        if (lw.ftype=1)
           FVOL[IDX] = CFWYFLWA(lw.atype,IVCR[IDX]) * lane[idx]
        endif
;;>
            XVOL[IDX] = NVOL[IDX] - FVOL[IDX]      ;
         ENDIF
            TMP=(IDX+1)                            ;
            NVOL[TMP] = IVOL[TMP]+XVOL[IDX]        ;
ENDLOOP
     NVOL[24] = IVOL[24] + XVOL[23]                ; Hour 24 (Last Hr)
     FVOL[24] = IVOL[24] + XVOL[23]                ; Hour 24 (Last Hr)
                                                   ; no Spread
```

```
;
;-----------------------------------------------------------------
; Now we're finished with the 'final' hourly volume development.
; Next, compute hourly V/C ratio, volume delay function, and Speed
; using the 'final' hourly volume. also compute Final VMT
; Note that the VC is based on NVOL, or the NONSPREADED volume (IVOL+XVOL)
;
;
;-----------------------------------------------------------------
;

LOOP IDX=1,24
;;   NVCR[IDX] =(NVOL[IDX] / CAPA[IDX])
;;>
      NVCR[IDX] =(FVOL[IDX] / CAPA[IDX])
        if (lw.ftype = 1 && IVCR[IDX] > NVCR[IDX])
                  NVCR[IDX] = IVCR[IDX]
        endif
;;>
   ;
   ; Compute the NEW  Speed based on NEW V/C ratio
   ;
          IF       (lw.ftype = 1,6 && lw.atype = 1,2)
             NSPD[IDX] = SPDA(1,NVCR[IDX])
          ENDIF
          IF       (lw.ftype = 1,6 && lw.atype = 3,4)
             NSPD[IDX] = SPDA(2,NVCR[IDX])
          ENDIF
          IF       (lw.ftype = 1,6 && lw.atype = 5-7)
             NSPD[IDX] = SPDA(3,NVCR[IDX])
          ENDIF

          IF       (lw.ftype = 2 && lw.atype = 1-2)
             NSPD[IDX] = SPDA(4,NVCR[IDX])
          ENDIF
          IF       (lw.ftype = 2 && lw.atype = 3-4)
             NSPD[IDX] = SPDA(5,NVCR[IDX])
          ENDIF
          IF       (lw.ftype = 2 && lw.atype = 5)
             NSPD[IDX] = SPDA(6,NVCR[IDX])
          ENDIF
          IF       (lw.ftype = 2 && lw.atype = 6-7)
             NSPD[IDX] = SPDA(7,NVCR[IDX])
          ENDIF

          IF       (lw.ftype = 3 && lw.atype = 1-2)
             NSPD[IDX] = SPDA(8,NVCR[IDX])
          ENDIF
          IF       (lw.ftype = 3 && lw.atype = 3-4)
             NSPD[IDX] = SPDA(9,NVCR[IDX])
          ENDIF
          IF       (lw.ftype = 3 && lw.atype = 5)
             NSPD[IDX] = SPDA(10,NVCR[IDX])
          ENDIF
          IF       (lw.ftype = 3 && lw.atype = 6-7)
             NSPD[IDX] = SPDA(11,NVCR[IDX])
          ENDIF

          IF       (lw.ftype = 4 && lw.atype = 1-2)
             NSPD[IDX] = SPDA(12,NVCR[IDX])
          ENDIF
          IF       (lw.ftype = 4 && lw.atype = 3-4)
             NSPD[IDX] = SPDA(13,NVCR[IDX])
          ENDIF
          IF       (lw.ftype = 4 && lw.atype = 5)
             NSPD[IDX] = SPDA(14,NVCR[IDX])
          ENDIF
```

```
         IF       (lw.ftype = 4 && lw.atype = 6-7)
             NSPD[IDX] = SPDA(15,NVCR[IDX])
         ENDIF


         IF       (lw.ftype = 5 && lw.atype = 1-2)
             NSPD[IDX] = SPDA(16,NVCR[IDX])
         ENDIF
         IF       (lw.ftype = 5 && lw.atype = 3-5)
             NSPD[IDX] = SPDA(17,NVCR[IDX])
         ENDIF
         IF       (lw.ftype = 5 && lw.atype = 6-7)
             NSPD[IDX] = SPDA(18,NVCR[IDX])
         ENDIF


   ; Compute the Final VMT / Speed / and Speed Difference
   ;
     FVMT[IDX]  = FVOL[IDX]* lw.DISTANCE    ; Final VMT = Final/spread vol * distance


     FSPD[IDX]  =               NSPD[IDX]   ; Final Speed is based non-spreaded volume


     DSPD[IDX]  =     ISPD[IDX]-NSPD[IDX]   ; ini spd - final (new) speed
ENDLOOP

;
;  Now smooth final speeds for the boundary hours 1,13,24
;

   FSPD[1]  =  FSPD[2]
   FSPD[24] =  FSPD[23]
   FSPD[13] = (FSPD[12] + FSPD[14]) / 2.0

;-----------------------------------------------------------------------------------------
---------------------


;  summarize initial / final(spread) hourly volumes

        IVOLTOT =  IVOLTOT  +
                   IVOL[1]  + IVOL[2]  + IVOL[3]  + IVOL[4]  + IVOL[5]  +  IVOL[6] +
                   IVOL[7]  + IVOL[8]  + IVOL[9]  + IVOL[10] + IVOL[11] + IVOL[12] +
                   IVOL[13] + IVOL[14] + IVOL[15] + IVOL[16] + IVOL[17] + IVOL[18] +
                   IVOL[19] + IVOL[20] + IVOL[21] + IVOL[22] + IVOL[23] + IVOL[24]

        FVOLTOT =  FVOLTOT  +
                   FVOL[1]  + FVOL[2]  + FVOL[3]  + FVOL[4]  + FVOL[5]  + FVOL[6]  +
                   FVOL[7]  + FVOL[8]  + FVOL[9]  + FVOL[10] + FVOL[11] + FVOL[12] +
                   FVOL[13] + FVOL[14] + FVOL[15] + FVOL[16] + FVOL[17] + FVOL[18] +
                   FVOL[19] + FVOL[20] + FVOL[21] + FVOL[22] + FVOL[23] + FVOL[24]

        FVMTTOT =  FVMTTOT  +
                   FVMT[1]  + FVMT[2]  + FVMT[3]  + FVMT[4]  + FVMT[5]  + FVMT[6]  +
                   FVMT[7]  + FVMT[8]  + FVMT[9]  + FVMT[10] + FVMT[11] + FVMT[12] +
                   FVMT[13] + FVMT[14] + FVMT[15] + FVMT[16] + FVMT[17] + FVMT[18] +
                   FVMT[19] + FVMT[20] + FVMT[21] + FVMT[22] + FVMT[23] + FVMT[24]
;
PRINT FORM=8.0,
 list= a(6),b(6),lw.ftype,lw.atype,
     lw.totvol,lw.amvol,lw.pmvol,pkdfpct,
      pkclass(3), ftclass(3), jpf(3),'   ',
     lw.amcap,lw.pmcap,lw.opcap,lw.jur, file=check.ini

PRINT FORM=10.2,
 list= a(6),b(6),
  ivol[1], ivol[2], ivol[3], ivol[4], ivol[5], ivol[6], ivol[7], ivol[8],
  ivol[9],ivol[10],ivol[11],ivol[12],ivol[13],ivol[14],ivol[15],ivol[16],
 ivol[17],ivol[18],ivol[19],ivol[20],ivol[21],ivol[22],ivol[23],ivol[24],
 file=check.i_v

 list= a(6),b(6),
  capa[1], capa[2], capa[3], capa[4], capa[5], capa[6], capa[7], capa[8],
  capa[9],capa[10],capa[11],capa[12],capa[13],capa[14],capa[15],capa[16],
 capa[17],capa[18],capa[19],capa[20],capa[21],capa[22],capa[23],capa[24],
 file=check.cap

 list= a(6),b(6),
  fvol[1], fvol[2], fvol[3], fvol[4], fvol[5], fvol[6], fvol[7], fvol[8],
  fvol[9],fvol[10],fvol[11],fvol[12],fvol[13],fvol[14],fvol[15],fvol[16],
 fvol[17],fvol[18],fvol[19],fvol[20],fvol[21],fvol[22],fvol[23],fvol[24],
 file=check.f_v

 list= a(6),b(6),
  xvol[1], xvol[2], xvol[3], xvol[4], xvol[5], xvol[6], xvol[7], xvol[8],
  xvol[9],xvol[10],xvol[11],xvol[12],xvol[13],xvol[14],xvol[15],xvol[16],
 xvol[17],xvol[18],xvol[19],xvol[20],xvol[21],xvol[22],xvol[23],xvol[24],
 file=check.x_v

 list= a(6),b(6),
  nvcr[1], nvcr[2], nvcr[3], nvcr[4], nvcr[5], nvcr[6], nvcr[7], nvcr[8],
  nvcr[9],nvcr[10],nvcr[11],nvcr[12],nvcr[13],nvcr[14],nvcr[15],nvcr[16],
 nvcr[17],nvcr[18],nvcr[19],nvcr[20],nvcr[21],nvcr[22],nvcr[23],nvcr[24],
 file=check.fvc

 list= a(6),b(6),
  ivcr[1], ivcr[2], ivcr[3], ivcr[4], ivcr[5], ivcr[6], ivcr[7], ivcr[8],
  ivcr[9],ivcr[10],ivcr[11],ivcr[12],ivcr[13],ivcr[14],ivcr[15],ivcr[16],
 ivcr[17],ivcr[18],ivcr[19],ivcr[20],ivcr[21],ivcr[22],ivcr[23],ivcr[24],
 file=check.ivc


PRINT FORM=3.0,
 list= a(6),b(6),
  fspd[1], fspd[2], fspd[3], fspd[4], fspd[5], fspd[6], fspd[7], fspd[8],
  fspd[9],fspd[10],fspd[11],fspd[12],fspd[13],fspd[14],fspd[15],fspd[16],
 fspd[17],fspd[18],fspd[19],fspd[20],fspd[21],fspd[22],fspd[23],fspd[24],
 file=HOURLYLK.SPD


PRINT FORM=3.0,
 list= a(6),b(6),
  dspd[1], dspd[2], dspd[3], dspd[4], dspd[5], dspd[6], dspd[7], dspd[8],
  dspd[9],dspd[10],dspd[11],dspd[12],dspd[13],dspd[14],dspd[15],dspd[16],
 dspd[17],dspd[18],dspd[19],dspd[20],dspd[21],dspd[22],dspd[23],dspd[24],
 file=check.sdf


PRINT FORM=10.2,
 list= a(6),b(6),
  fvmt[1], fvmt[2], fvmt[3], fvmt[4], fvmt[5], fvmt[6], fvmt[7], fvmt[8],
  fvmt[9],fvmt[10],fvmt[11],fvmt[12],fvmt[13],fvmt[14],fvmt[15],fvmt[16],
 fvmt[17],fvmt[18],fvmt[19],fvmt[20],fvmt[21],fvmt[22],fvmt[23],fvmt[24],
 file=HOURLYLK.VMT


 endif                        ;<<<<<<<<<-------- select links

; clear out all hourly arrays
Loop idx=1,24
 CAPA[idx] = 0
 IVOL[idx] = 0
 IVCR[idx] = 0
 IVMT[idx] = 0
 ISPD[idx] = 0
```

```
 XVOL[idx] = 0
 NVOL[idx] = 0
 FVOL[idx] = 0
 NVCR[idx] = 0
 NSPD[idx] = 0
 FSPD[idx] = 0
 DSPD[idx] = 0
 LANE[idx] = 0
 FVMT[idx] = 0
endloop

endlinkloop



endphase
endrun




; -------------------------------------------------------------
*copy tppl*.prn peak_spread.rpt
```

RUNNING.S

```
*del tppl*.prn
;-------------------------------------------------------------------------
; Program: RUNNING.S
;
; Calculate running emissions using hrly volumes developed from
; the Peak_Spread.S program
;
;
;
;
;-------------------------------------------------------------------------

 INNET   ='temp.net'       ; input basic network
 HRNETVMT='hourlylk.vmt'   ; input hourly vmt
 HRNETSPD='hourlylk.spd'   ; input hourly speed
 JRTRPDST='trip.dst'       ; veh trip juris level distr

;-------------------------------------------------------------
; Emission -
;-------------------------------------------------------------
;  1) DC                      15) PW
;  2) Mtg                     16) Sta
;  3) PG                      17) Alx
;  4) How                     18) StM
;  5) Aar                     19) WashCoExtls   2164-2168
;  6) Car                     20) Clk
;  7) Balt Extls 2172-2191    21) Fau
;  8) Cal                     22) Jef
;  9) Chs                     23) VA W.Extls    2154-2163
; 10) Frd                     24) Spts
; 11) Frd. Extls  2169-2171   25) KG
; 12) Arl                     26) Fbg
; 13) FFx                     27) VA S.Extls    2145-2153
; 14) Ldn
;-------------------------------------------------------------
 j01A_RAT='%_m6Sub_%%_m6Pre_%dc.r_a'; DC      Art HC,CO,Nx Rates
 j02A_RAT='%_m6Sub_%%_m6Pre_%mc.r_a'; Mtg     Art HC,CO,Nx Rates
 j03A_RAT='%_m6Sub_%%_m6Pre_%pg.r_a'; PG      Art HC,CO,Nx Rates
 j04A_RAT='%_m6Sub_%%_m6Pre_%pg.r_a'; "       Art HC,CO,Nx Rates
 j05A_RAT='%_m6Sub_%%_m6Pre_%pg.r_a'; "       Art HC,CO,Nx Rates
 j06A_RAT='%_m6Sub_%%_m6Pre_%pg.r_a'; "       Art HC,CO,Nx Rates
 j07A_RAT='%_m6Sub_%%_m6Pre_%pg.r_a'; "       Art HC,CO,Nx Rates
 j08A_RAT='%_m6Sub_%%_m6Pre_%ca.r_a'; Cal     Art HC,CO,Nx Rates
 j09A_RAT='%_m6Sub_%%_m6Pre_%ch.r_a'; Chs     Art HC,CO,Nx Rates
 j10A_RAT='%_m6Sub_%%_m6Pre_%fr.r_a'; Frd     Art HC,CO,Nx Rates
 j11A_RAT='%_m6Sub_%%_m6Pre_%fr.r_a'; "       Art HC,CO,Nx Rates
 j12A_RAT='%_m6Sub_%%_m6Pre_%ar.r_a'; Arl     Art HC,CO,Nx Rates
 j13A_RAT='%_m6Sub_%%_m6Pre_%fx.r_a'; Ffx     Art HC,CO,Nx Rates
 j14A_RAT='%_m6Sub_%%_m6Pre_%ld.r_a'; Ldn     Art HC,CO,Nx Rates
 j15A_RAT='%_m6Sub_%%_m6Pre_%pw.r_a'; PW      Art HC,CO,Nx Rates
 j16A_RAT='%_m6Sub_%%_m6Pre_%st.r_a'; Sta     Art HC,CO,Nx Rates
 j17A_RAT='%_m6Sub_%%_m6Pre_%al.r_a'; Alx     Art HC,CO,Nx Rates
 j18A_RAT='%_m6Sub_%%_m6Pre_%sm.r_a'; StM     Art HC,CO,Nx Rates
 j19A_RAT='%_m6Sub_%%_m6Pre_%we.r_a'; Wash    Art HC,CO,Nx Rates
 j20A_RAT='%_m6Sub_%%_m6Pre_%cl.r_a'; Clk     Art HC,CO,Nx Rates
 j21A_RAT='%_m6Sub_%%_m6Pre_%cl.r_a'; "       Art HC,CO,Nx Rates
 j22A_RAT='%_m6Sub_%%_m6Pre_%cl.r_a'; "       Art HC,CO,Nx Rates
 j23A_RAT='%_m6Sub_%%_m6Pre_%cl.r_a'; "       Art HC,CO,Nx Rates
 j24A_RAT='%_m6Sub_%%_m6Pre_%sp.r_a'; Spts    Art HC,CO,Nx Rates
 j25A_RAT='%_m6Sub_%%_m6Pre_%sp.r_a'; "       Art HC,CO,Nx Rates
 j26A_RAT='%_m6Sub_%%_m6Pre_%sp.r_a'; "       Art HC,CO,Nx Rates
 j27A_RAT='%_m6Sub_%%_m6Pre_%sp.r_a'; "       Art HC,CO,Nx Rates

;-------------------------------------------------
 j01F_RAT='%_m6Sub_%%_m6Pre_%dc.r_f'; DC      Fwy HC,CO,Nx Rates
 j02F_RAT='%_m6Sub_%%_m6Pre_%mc.r_f'; Mtg     Fwy HC,CO,Nx Rates
 j03F_RAT='%_m6Sub_%%_m6Pre_%pg.r_f'; PG      Fwy HC,CO,Nx Rates
 j04F_RAT='%_m6Sub_%%_m6Pre_%pg.r_f'; "       Fwy HC,CO,Nx Rates
 j05F_RAT='%_m6Sub_%%_m6Pre_%pg.r_f'; "       Fwy HC,CO,Nx Rates
 j06F_RAT='%_m6Sub_%%_m6Pre_%pg.r_f'; "       Fwy HC,CO,Nx Rates
 j07F_RAT='%_m6Sub_%%_m6Pre_%pg.r_f'; "       Fwy HC,CO,Nx Rates
 j08F_RAT='%_m6Sub_%%_m6Pre_%ca.r_f'; Cal     Fwy HC,CO,Nx Rates
 j09F_RAT='%_m6Sub_%%_m6Pre_%ch.r_f'; Chs     Fwy HC,CO,Nx Rates
 j10F_RAT='%_m6Sub_%%_m6Pre_%fr.r_f'; Frd     Fwy HC,CO,Nx Rates
 j11F_RAT='%_m6Sub_%%_m6Pre_%fr.r_f'; "       Fwy HC,CO,Nx Rates
 j12F_RAT='%_m6Sub_%%_m6Pre_%ar.r_f'; Arl     Fwy HC,CO,Nx Rates
 j13F_RAT='%_m6Sub_%%_m6Pre_%fx.r_f'; Ffx     Fwy HC,CO,Nx Rates
 j14F_RAT='%_m6Sub_%%_m6Pre_%ld.r_f'; Ldn     Fwy HC,CO,Nx Rates
 j15F_RAT='%_m6Sub_%%_m6Pre_%pw.r_f'; PW      Fwy HC,CO,Nx Rates
 j16F_RAT='%_m6Sub_%%_m6Pre_%st.r_f'; Sta     Fwy HC,CO,Nx Rates
 j17F_RAT='%_m6Sub_%%_m6Pre_%al.r_f'; Alx     Fwy HC,CO,Nx Rates
 j18F_RAT='%_m6Sub_%%_m6Pre_%sm.r_f'; StM     Fwy HC,CO,Nx Rates
 j19F_RAT='%_m6Sub_%%_m6Pre_%we.r_f'; Wash    Fwy HC,CO,Nx Rates
 j20F_RAT='%_m6Sub_%%_m6Pre_%cl.r_f'; Clk     Fwy HC,CO,Nx Rates
 j21F_RAT='%_m6Sub_%%_m6Pre_%cl.r_f'; "       Fwy HC,CO,Nx Rates
 j22F_RAT='%_m6Sub_%%_m6Pre_%cl.r_f'; "       Fwy HC,CO,Nx Rates
 j23F_RAT='%_m6Sub_%%_m6Pre_%cl.r_f'; "       Fwy HC,CO,Nx Rates
 j24F_RAT='%_m6Sub_%%_m6Pre_%sp.r_f'; Spts    Fwy HC,CO,Nx Rates
 j25F_RAT='%_m6Sub_%%_m6Pre_%sp.r_f'; "       Fwy HC,CO,Nx Rates
 j26F_RAT='%_m6Sub_%%_m6Pre_%sp.r_f'; "       Fwy HC,CO,Nx Rates
 j27F_RAT='%_m6Sub_%%_m6Pre_%sp.r_f'; "       Fwy HC,CO,Nx Rates

;-------------------------------------------------
 j01R_RAT='%_m6Sub_%%_m6Pre_%dc.ram'; DC      Rmp HC,CO,Nx Rates
 j02R_RAT='%_m6Sub_%%_m6Pre_%mc.ram'; Mtg     Rmp HC,CO,Nx Rates
 j03R_RAT='%_m6Sub_%%_m6Pre_%pg.ram'; PG      Rmp HC,CO,Nx Rates
 j04R_RAT='%_m6Sub_%%_m6Pre_%pg.ram'; "       Rmp HC,CO,Nx Rates
 j05R_RAT='%_m6Sub_%%_m6Pre_%pg.ram'; "       Rmp HC,CO,Nx Rates
 j06R_RAT='%_m6Sub_%%_m6Pre_%pg.ram'; "       Rmp HC,CO,Nx Rates
 j07R_RAT='%_m6Sub_%%_m6Pre_%pg.ram'; "       Rmp HC,CO,Nx Rates
 j08R_RAT='%_m6Sub_%%_m6Pre_%ca.ram'; Cal     Rmp HC,CO,Nx Rates
 j09R_RAT='%_m6Sub_%%_m6Pre_%ch.ram'; Chs     Rmp HC,CO,Nx Rates
 j10R_RAT='%_m6Sub_%%_m6Pre_%fr.ram'; Frd     Rmp HC,CO,Nx Rates
 j11R_RAT='%_m6Sub_%%_m6Pre_%fr.ram'; "       Rmp HC,CO,Nx Rates
 j12R_RAT='%_m6Sub_%%_m6Pre_%ar.ram'; Arl     Rmp HC,CO,Nx Rates
 j13R_RAT='%_m6Sub_%%_m6Pre_%fx.ram'; Ffx     Rmp HC,CO,Nx Rates
 j14R_RAT='%_m6Sub_%%_m6Pre_%ld.ram'; Ldn     Rmp HC,CO,Nx Rates
 j15R_RAT='%_m6Sub_%%_m6Pre_%pw.ram'; PW      Rmp HC,CO,Nx Rates
 j16R_RAT='%_m6Sub_%%_m6Pre_%st.ram'; Sta     Rmp HC,CO,Nx Rates
 j17R_RAT='%_m6Sub_%%_m6Pre_%al.ram'; Alx     Rmp HC,CO,Nx Rates
 j18R_RAT='%_m6Sub_%%_m6Pre_%sm.ram'; StM     Rmp HC,CO,Nx Rates
 j19R_RAT='%_m6Sub_%%_m6Pre_%we.ram'; Wash    Rmp HC,CO,Nx Rates
 j20R_RAT='%_m6Sub_%%_m6Pre_%cl.ram'; Clk     Rmp HC,CO,Nx Rates
 j21R_RAT='%_m6Sub_%%_m6Pre_%cl.ram'; "       Rmp HC,CO,Nx Rates
 j22R_RAT='%_m6Sub_%%_m6Pre_%cl.ram'; "       Rmp HC,CO,Nx Rates
 j23R_RAT='%_m6Sub_%%_m6Pre_%cl.ram'; "       Rmp HC,CO,Nx Rates
 j24R_RAT='%_m6Sub_%%_m6Pre_%sp.ram'; Spts    Rmp HC,CO,Nx Rates
 j25R_RAT='%_m6Sub_%%_m6Pre_%sp.ram'; "       Rmp HC,CO,Nx Rates
 j26R_RAT='%_m6Sub_%%_m6Pre_%sp.ram'; "       Rmp HC,CO,Nx Rates
 j27R_RAT='%_m6Sub_%%_m6Pre_%sp.ram'; "       Rmp HC,CO,Nx Rates

run pgm=hwynet
    neti     = @innet@
    neto     = temp2.net

    linki[2] = @hrnetvmt@ var = a, b,
    vmt01 vmt02,vmt03,vmt04,vmt05,vmt06,vmt07,vmt08,vmt09,vmt10,
    vmt11 vmt12,vmt13,vmt14,vmt15,vmt16,vmt17,vmt18,vmt19,vmt20,
    vmt21 vmt22,vmt23,vmt24

    linki[3] = @hrnetspd@ var = a, b,
```

```
    spd01 spd02,spd03,spd04,spd05,spd06,spd07,spd08,spd09,spd10,
    spd11 spd12,spd13,spd14,spd15,spd16,spd17,spd18,spd19,spd20,
    spd21 spd22,spd23,spd24

;  CREATE SOME FREQUENCY-CROSSTABS FOR CHECKING
;  if (i624vol > 0 && FType >0)
;           _cnt=1
;else
;           _cnt =0
;endif
;CROSSTAB VAR=_CNT,ROW=FTYPE, RANGE=0-5-1,0-5
;CROSSTAB VAR=_CNT,ROW=FTYPE, RANGE=1-7-1, COL=OPLANE,    RANGE=1-7-1

  IF (A=3192-3200           )  ;**debug selection
    list = a(6), b(6), ftype(3)
  endif
endrun


run pgm=hwyload
    neti     = temp2.net
; Set up arrays
  array,        ;
    vmt   =24,  ; Hourly VMT
    spd   =24,  ; Hourly Speed
    chr   =24,  ; Hourly composite HC Rate
    ccr   =24,  ; Hourly composite CO Rate
    cnr   =24,  ; Hourly composite Nx Rate
    hem   =24,  ; Hourly HC emissions
    cem   =24,  ; Hourly CO emissions
    nem   =24   ; Hourly Nx emissions

; Set up Lookup tables
;---------------------------------------------------------------------
; Jurisdictional Trip Distribution Array                             -
;---------------------------------------------------------------------
  LOOKUP NAME=JDST,               ;
         LOOKUP[01]=1, RESULT=2,  ;
         LOOKUP[02]=1, RESULT=3,  ;
         LOOKUP[03]=1, RESULT=4,  ;
         LOOKUP[04]=1, RESULT=5,  ;
         LOOKUP[05]=1, RESULT=6,  ;
         LOOKUP[06]=1, RESULT=7,  ;
         LOOKUP[07]=1, RESULT=8,  ;
         LOOKUP[08]=1, RESULT=9,  ;
         LOOKUP[09]=1, RESULT=10, ;
         LOOKUP[10]=1, RESULT=11, ;
         LOOKUP[11]=1, RESULT=12, ;
         LOOKUP[12]=1, RESULT=13, ;
         LOOKUP[13]=1, RESULT=14, ;
         LOOKUP[14]=1, RESULT=15, ;
         LOOKUP[15]=1, RESULT=16, ;
         LOOKUP[16]=1, RESULT=17, ;
         LOOKUP[17]=1, RESULT=18, ;
         LOOKUP[18]=1, RESULT=19, ;
         LOOKUP[19]=1, RESULT=20, ;
         LOOKUP[20]=1, RESULT=21, ;
         LOOKUP[21]=1, RESULT=22, ;
         LOOKUP[22]=1, RESULT=23, ;
         LOOKUP[23]=1, RESULT=24, ;
         LOOKUP[24]=1, RESULT=25, ;
         LOOKUP[25]=1, RESULT=26, ;
         LOOKUP[26]=1, RESULT=27, ;
         LOOKUP[27]=1, RESULT=28, ;
         INTERPOLATE = N,LIST=T,File=@JRTRPDST@

;---------------------------------------------------------------------
; Emission Rate Speed Lookups by jurisdiction                        -
```

```
;---------------------------------------------------------------------
; ARTERIAL LOOK-Ups Here
;---------------------------------------------------------------------
  LOOKUP NAME=J01AR,              ;
         LOOKUP[1]=1, RESULT=2,   ;
         LOOKUP[2]=1, RESULT=3,   ;
         LOOKUP[3]=1, RESULT=4,   ;
         INTERPOLATE = N,LIST=T,file=@J01A_RAT@

  LOOKUP NAME=J02AR,              ;
         LOOKUP[1]=1, RESULT=2,   ;
         LOOKUP[2]=1, RESULT=3,   ;
         LOOKUP[3]=1, RESULT=4,   ;
         INTERPOLATE = N,LIST=T,file=@J02A_RAT@

  LOOKUP NAME=J03AR,              ;
         LOOKUP[1]=1, RESULT=2,   ;
         LOOKUP[2]=1, RESULT=3,   ;
         LOOKUP[3]=1, RESULT=4,   ;
         INTERPOLATE = N,LIST=T,file=@J03A_RAT@

  LOOKUP NAME=J04AR,              ;
         LOOKUP[1]=1, RESULT=2,   ;
         LOOKUP[2]=1, RESULT=3,   ;
         LOOKUP[3]=1, RESULT=4,   ;
         INTERPOLATE = N,LIST=T,file=@J04A_RAT@

  LOOKUP NAME=J05AR,              ;
         LOOKUP[1]=1, RESULT=2,   ;
         LOOKUP[2]=1, RESULT=3,   ;
         LOOKUP[3]=1, RESULT=4,   ;
         INTERPOLATE = N,LIST=T,file=@J05A_RAT@

  LOOKUP NAME=J06AR,              ;
         LOOKUP[1]=1, RESULT=2,   ;
         LOOKUP[2]=1, RESULT=3,   ;
         LOOKUP[3]=1, RESULT=4,   ;
         INTERPOLATE = N,LIST=T,file=@J06A_RAT@

  LOOKUP NAME=J07AR,              ;
         LOOKUP[1]=1, RESULT=2,   ;
         LOOKUP[2]=1, RESULT=3,   ;
         LOOKUP[3]=1, RESULT=4,   ;
         INTERPOLATE = N,LIST=T,file=@J07A_RAT@

  LOOKUP NAME=J08AR,              ;
         LOOKUP[1]=1, RESULT=2,   ;
         LOOKUP[2]=1, RESULT=3,   ;
         LOOKUP[3]=1, RESULT=4,   ;
         INTERPOLATE = N,LIST=T,file=@J08A_RAT@

  LOOKUP NAME=J09AR,              ;
         LOOKUP[1]=1, RESULT=2,   ;
         LOOKUP[2]=1, RESULT=3,   ;
         LOOKUP[3]=1, RESULT=4,   ;
         INTERPOLATE = N,LIST=T,file=@J09A_RAT@

  LOOKUP NAME=J10AR,              ;
         LOOKUP[1]=1, RESULT=2,   ;
         LOOKUP[2]=1, RESULT=3,   ;
         LOOKUP[3]=1, RESULT=4,   ;
         INTERPOLATE = N,LIST=T,file=@J10A_RAT@

  LOOKUP NAME=J11AR,              ;
         LOOKUP[1]=1, RESULT=2,   ;
         LOOKUP[2]=1, RESULT=3,   ;
         LOOKUP[3]=1, RESULT=4,   ;
         INTERPOLATE = N,LIST=T,file=@J11A_RAT@
```

```
LOOKUP NAME=J12AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J12A_RAT@

LOOKUP NAME=J13AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J13A_RAT@

LOOKUP NAME=J14AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J14A_RAT@

LOOKUP NAME=J15AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J15A_RAT@

LOOKUP NAME=J16AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J16A_RAT@

LOOKUP NAME=J17AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J17A_RAT@

LOOKUP NAME=J18AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J18A_RAT@

LOOKUP NAME=J19AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J19A_RAT@

LOOKUP NAME=J20AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J20A_RAT@

LOOKUP NAME=J21AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J21A_RAT@

LOOKUP NAME=J22AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J22A_RAT@

LOOKUP NAME=J23AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J23A_RAT@

LOOKUP NAME=J24AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J24A_RAT@

LOOKUP NAME=J25AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J25A_RAT@

LOOKUP NAME=J26AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J26A_RAT@

LOOKUP NAME=J27AR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J27A_RAT@

;---------------------------------------------------------------------
; FREEWAY  LOOK-Ups Here
;---------------------------------------------------------------------
LOOKUP NAME=J01FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J01F_RAT@

LOOKUP NAME=J02FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J02F_RAT@

LOOKUP NAME=J03FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J03F_RAT@

LOOKUP NAME=J04FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J04F_RAT@

LOOKUP NAME=J05FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J05F_RAT@

LOOKUP NAME=J06FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J06F_RAT@
```

```
LOOKUP NAME=J07FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J07F_RAT@

LOOKUP NAME=J08FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J08F_RAT@

LOOKUP NAME=J09FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J09F_RAT@

LOOKUP NAME=J10FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J10F_RAT@

LOOKUP NAME=J11FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J11F_RAT@

LOOKUP NAME=J12FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J12F_RAT@

LOOKUP NAME=J13FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J13F_RAT@

LOOKUP NAME=J14FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J14F_RAT@

LOOKUP NAME=J15FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J15F_RAT@

LOOKUP NAME=J16FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J16F_RAT@

LOOKUP NAME=J17FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J17F_RAT@

LOOKUP NAME=J18FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J18F_RAT@

LOOKUP NAME=J19FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J19F_RAT@

LOOKUP NAME=J20FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J20F_RAT@

LOOKUP NAME=J21FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J21F_RAT@

LOOKUP NAME=J22FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J22F_RAT@

LOOKUP NAME=J23FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J23F_RAT@

LOOKUP NAME=J24FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J24F_RAT@

LOOKUP NAME=J25FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J25F_RAT@

LOOKUP NAME=J26FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J26F_RAT@

LOOKUP NAME=J27FR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J27F_RAT@

;-----------------------------------------------------------------
; RAMP     LOOK-Ups Here
;-----------------------------------------------------------------
LOOKUP NAME=J01RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J01R_RAT@

LOOKUP NAME=J02RR,              ;
```

```
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J02R_RAT@

LOOKUP NAME=J03RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J03R_RAT@

LOOKUP NAME=J04RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J04R_RAT@

LOOKUP NAME=J05RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J05R_RAT@

LOOKUP NAME=J06RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J06R_RAT@

LOOKUP NAME=J07RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J07R_RAT@

LOOKUP NAME=J08RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J08R_RAT@

LOOKUP NAME=J09RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J09R_RAT@

LOOKUP NAME=J10RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J10R_RAT@

LOOKUP NAME=J11RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J11R_RAT@

LOOKUP NAME=J12RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J12R_RAT@

LOOKUP NAME=J13RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J13R_RAT@

LOOKUP NAME=J14RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J14R_RAT@

LOOKUP NAME=J15RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J15R_RAT@

LOOKUP NAME=J16RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J16R_RAT@

LOOKUP NAME=J17RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J17R_RAT@

LOOKUP NAME=J18RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J18R_RAT@

LOOKUP NAME=J19RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J19R_RAT@

LOOKUP NAME=J20RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J20R_RAT@

LOOKUP NAME=J21RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J21R_RAT@

LOOKUP NAME=J22RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J22R_RAT@

LOOKUP NAME=J23RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J23R_RAT@

LOOKUP NAME=J24RR,              ;
        LOOKUP[1]=1, RESULT=2,  ;
        LOOKUP[2]=1, RESULT=3,  ;
        LOOKUP[3]=1, RESULT=4,  ;
        INTERPOLATE = N,LIST=T,file=@J24R_RAT@
```

```
  LOOKUP NAME=J25RR,              ;
         LOOKUP[1]=1, RESULT=2,  ;
         LOOKUP[2]=1, RESULT=3,  ;
         LOOKUP[3]=1, RESULT=4,  ;
         INTERPOLATE = N,LIST=T,file=@J25R_RAT@

  LOOKUP NAME=J26RR,              ;
         LOOKUP[1]=1, RESULT=2,  ;
         LOOKUP[2]=1, RESULT=3,  ;
         LOOKUP[3]=1, RESULT=4,  ;
         INTERPOLATE = N,LIST=T,file=@J26R_RAT@

  LOOKUP NAME=J27RR,              ;
         LOOKUP[1]=1, RESULT=2,  ;
         LOOKUP[2]=1, RESULT=3,  ;
         LOOKUP[3]=1, RESULT=4,  ;
         INTERPOLATE = N,LIST=T,file=@J27R_RAT@
;---------------------------------------------------------------------
; End of Emission Rate Lookups
;---------------------------------------------------------------------
  zones=1

  phase=linkread
    t0=li.amhtime       ; dummy in a t0 -any link attribute will do
    lw.ftype  = li.ftype   ; facility type code
    lw.jur    = li.jur     ; jurisdiction code
    lw.totvol = li.i624vol ; total link volume

    lw.vmt01  = li.vmt01   ;
    lw.vmt02  = li.vmt02   ;
    lw.vmt03  = li.vmt03   ;
    lw.vmt04  = li.vmt04   ;
    lw.vmt05  = li.vmt05   ;
    lw.vmt06  = li.vmt06   ;
    lw.vmt07  = li.vmt07   ;
    lw.vmt08  = li.vmt08   ;
    lw.vmt09  = li.vmt09   ;
    lw.vmt10  = li.vmt10   ;
    lw.vmt11  = li.vmt11   ;
    lw.vmt12  = li.vmt12   ;
    lw.vmt13  = li.vmt13   ;
    lw.vmt14  = li.vmt14   ;
    lw.vmt15  = li.vmt15   ;
    lw.vmt16  = li.vmt16   ;
    lw.vmt17  = li.vmt17   ;
    lw.vmt18  = li.vmt18   ;
    lw.vmt19  = li.vmt19   ;
    lw.vmt20  = li.vmt20   ;
    lw.vmt21  = li.vmt21   ;
    lw.vmt22  = li.vmt22   ;
    lw.vmt23  = li.vmt23   ;
    lw.vmt24  = li.vmt24   ;


    lw.spd01  = li.spd01   ;
    lw.spd02  = li.spd02   ;
    lw.spd03  = li.spd03   ;
    lw.spd04  = li.spd04   ;
    lw.spd05  = li.spd05   ;
    lw.spd06  = li.spd06   ;
    lw.spd07  = li.spd07   ;
    lw.spd08  = li.spd08   ;
    lw.spd09  = li.spd09   ;
    lw.spd10  = li.spd10   ;
    lw.spd11  = li.spd11   ;
    lw.spd12  = li.spd12   ;
    lw.spd13  = li.spd13   ;
    lw.spd14  = li.spd14   ;
    lw.spd15  = li.spd15   ;
    lw.spd16  = li.spd16   ;
    lw.spd17  = li.spd17   ;
    lw.spd18  = li.spd18   ;
    lw.spd19  = li.spd19   ;
    lw.spd20  = li.spd20   ;
    lw.spd21  = li.spd21   ;
    lw.spd22  = li.spd22   ;
    lw.spd23  = li.spd23   ;
    lw.spd24  = li.spd24   ;
  endphase

  phase=iloop
       linkloop
;;        IF (A=3192-3200           )  ;**debug selection
       IF (lw.FTYPE > 0 && lw.TOTVOL > 0)  ; <<-----  Link Selection Criterea
;---------------------------------------------------------------------
;Next  define air quality jurisdiction codes (jcd).                  -
;   These are different from the standard                            -
;   'jur' codes on network links, since they include                 -
;   include some external station ranges.                            -       -
;---------------------------------------------------------------------
;


    IF (lw.Jur = 0) ; 1  DC
        JCD =    1
    ENDIF
    IF (lw.Jur = 1) ; 2  Mtg
        JCD =    2
    ENDIF
    IF (lw.Jur = 2) ; 3  PG
        JCD =    3
    ENDIF
    IF (lw.Jur = 9) ;10  Frd
        JCD =   10
    ENDIF
    IF (lw.Jur = 14) ; 6  Car
        JCD =    6
    ENDIF
    IF (lw.Jur = 10) ; 4  How
        JCD =    4
    ENDIF
    IF (lw.Jur = 11) ; 5  Aar
        JCD =    5
    ENDIF
    IF (lw.Jur = 15) ; 8  Cal
        JCD =    8
    ENDIF
    IF (lw.Jur = 16) ;18  StM
        JCD =   18
    ENDIF
    IF (lw.Jur = 12) ; 9  Chs
        JCD =    9
    ENDIF
    IF (lw.Jur = 3) ;12  Arl
        JCD =   12
    ENDIF
    IF (lw.Jur = 4) ;17  Alx
        JCD =   17
    ENDIF
    IF (lw.Jur = 5) ;13  FFx
        JCD =   13
    ENDIF
    IF (lw.Jur = 6) ;14  Ldn
        JCD =   14
    ENDIF
```

```
   IF (lw.Jur = 7) ;15  PW
       JCD =   15
   ENDIF
   IF (lw.Jur = 17) ;25  KG
       JCD =   25
   ENDIF
   IF (lw.Jur = 19) ;16  Sta
       JCD =   16
   ENDIF
   IF (lw.Jur = 18) ;26  Fbg
       JCD =   26
   ENDIF
   IF (lw.Jur = 20) ;24  Spts
       JCD =   24
   ENDIF
   IF (lw.Jur = 21) ;21  Fau
       JCD =   21
   ENDIF
   IF (lw.Jur = 22) ;20  Clk
       JCD =   20
   ENDIF
   IF (lw.Jur = 23) ;22  Jef
       JCD =   22
   ENDIF
   IF ((A>2144 & A<2154) || (B>2144 & B<2154)) ;27  S.Extls    2145-2153
       JCD =   27
   ENDIF
   IF ((A>2153 & A<2164) || (B>2153 & B<2164)) ;23  W.Extls    2154-2163
       JCD =   23
   ENDIF
   IF ((A>2163 & A<2169) || (B>2163 & B<2169)) ;19  NW. Extl   2164-2168
       JCD =   19
   ENDIF
   IF ((A>2168 & A<2172) || (B>2168 & B<2172)) ;11  Frd        2169-2171
       JCD =   11
   ENDIF
   IF ((A>2171 & A<2192) || (B>2171 & B<2192)) ; 7  Balt Extls 2172-2191
       JCD =    7
   ENDIF

;---------------------------------------------------------------------
; end of JCD definitions
;---------------------------------------------------------------------
;

;---------------------------------------------------------------------
; Put vmt and spd into hourly arrays
;---------------------------------------------------------------------
;
    vmt[01]=lw.vmt01 vmt[02]=lw.vmt02 vmt[03]=lw.vmt03 vmt[04]=lw.vmt04
    vmt[05]=lw.vmt05 vmt[06]=lw.vmt06 vmt[07]=lw.vmt07 vmt[08]=lw.vmt08
    vmt[09]=lw.vmt09 vmt[10]=lw.vmt10 vmt[11]=lw.vmt11 vmt[12]=lw.vmt12
    vmt[13]=lw.vmt13 vmt[14]=lw.vmt14 vmt[15]=lw.vmt15 vmt[16]=lw.vmt16
    vmt[17]=lw.vmt17 vmt[18]=lw.vmt18 vmt[19]=lw.vmt19 vmt[20]=lw.vmt20
    vmt[21]=lw.vmt21 vmt[22]=lw.vmt22 vmt[23]=lw.vmt23 vmt[24]=lw.vmt24


    spd[01]=lw.spd01 spd[02]=lw.spd02 spd[03]=lw.spd03 spd[04]=lw.spd04
    spd[05]=lw.spd05 spd[06]=lw.spd06 spd[07]=lw.spd07 spd[08]=lw.spd08
    spd[09]=lw.spd09 spd[10]=lw.spd10 spd[11]=lw.spd11 spd[12]=lw.spd12
    spd[13]=lw.spd13 spd[14]=lw.spd14 spd[15]=lw.spd15 spd[16]=lw.spd16
    spd[17]=lw.spd17 spd[18]=lw.spd18 spd[19]=lw.spd19 spd[20]=lw.spd20
    spd[21]=lw.spd21 spd[22]=lw.spd22 spd[23]=lw.spd23 spd[24]=lw.spd24


;---------------------------------------------------------------------
; Now compute hourly composite HC,CO,NOx emission rates (CHR,CCR,CNR)
; which, for running rates, are computed by facility type
; (freeways, arterials, and freeway ramps)
;---------------------------------------------------------------------
;
if (lw.ftype = 1 || lw.ftype = 5)
   LOOP IDX= 1,24
        CHR[IDX] = (J01FR(1,Spd[idx]) * JDST(JCD,1))  +   ; COMPOSITE
                   (J02FR(1,Spd[idx]) * JDST(JCD,2))  +   ; HC FREEWAY
                   (J03FR(1,Spd[idx]) * JDST(JCD,3))  +   ; RATE
                   (J04FR(1,Spd[idx]) * JDST(JCD,4))  +   ;
                   (J05FR(1,Spd[idx]) * JDST(JCD,5))  +   ;
                   (J06FR(1,Spd[idx]) * JDST(JCD,6))  +   ;
                   (J07FR(1,Spd[idx]) * JDST(JCD,7))  +   ;
                   (J08FR(1,Spd[idx]) * JDST(JCD,8))  +   ;
                   (J09FR(1,Spd[idx]) * JDST(JCD,9))  +   ;
                   (J10FR(1,Spd[idx]) * JDST(JCD,10)) +   ;
                   (J11FR(1,Spd[idx]) * JDST(JCD,11)) +   ;
                   (J12FR(1,Spd[idx]) * JDST(JCD,12)) +   ;
                   (J13FR(1,Spd[idx]) * JDST(JCD,13)) +   ;
                   (J14FR(1,Spd[idx]) * JDST(JCD,14)) +   ;
                   (J15FR(1,Spd[idx]) * JDST(JCD,15)) +   ;
                   (J16FR(1,Spd[idx]) * JDST(JCD,16)) +   ;
                   (J17FR(1,Spd[idx]) * JDST(JCD,17)) +   ;
                   (J18FR(1,Spd[idx]) * JDST(JCD,18)) +   ;
                   (J19FR(1,Spd[idx]) * JDST(JCD,19)) +   ;
                   (J20FR(1,Spd[idx]) * JDST(JCD,20)) +   ;
                   (J21FR(1,Spd[idx]) * JDST(JCD,21)) +   ;
                   (J22FR(1,Spd[idx]) * JDST(JCD,22)) +   ;
                   (J23FR(1,Spd[idx]) * JDST(JCD,23)) +   ;
                   (J24FR(1,Spd[idx]) * JDST(JCD,24)) +   ;
                   (J25FR(1,Spd[idx]) * JDST(JCD,25)) +   ;
                   (J26FR(1,Spd[idx]) * JDST(JCD,26)) +   ;
                   (J27FR(1,Spd[idx]) * JDST(JCD,27))     ;


        CCR[IDX] = (J01FR(2,Spd[idx]) * JDST(JCD,1))  +   ; COMPOSITE
                   (J02FR(2,Spd[idx]) * JDST(JCD,2))  +   ; CO FREEWAY
                   (J03FR(2,Spd[idx]) * JDST(JCD,3))  +   ; RATE
                   (J04FR(2,Spd[idx]) * JDST(JCD,4))  +   ;
                   (J05FR(2,Spd[idx]) * JDST(JCD,5))  +   ;
                   (J06FR(2,Spd[idx]) * JDST(JCD,6))  +   ;
                   (J07FR(2,Spd[idx]) * JDST(JCD,7))  +   ;
                   (J08FR(2,Spd[idx]) * JDST(JCD,8))  +   ;
                   (J09FR(2,Spd[idx]) * JDST(JCD,9))  +   ;
                   (J10FR(2,Spd[idx]) * JDST(JCD,10)) +   ;
                   (J11FR(2,Spd[idx]) * JDST(JCD,11)) +   ;
                   (J12FR(2,Spd[idx]) * JDST(JCD,12)) +   ;
                   (J13FR(2,Spd[idx]) * JDST(JCD,13)) +   ;
                   (J14FR(2,Spd[idx]) * JDST(JCD,14)) +   ;
                   (J15FR(2,Spd[idx]) * JDST(JCD,15)) +   ;
                   (J16FR(2,Spd[idx]) * JDST(JCD,16)) +   ;
                   (J17FR(2,Spd[idx]) * JDST(JCD,17)) +   ;
                   (J18FR(2,Spd[idx]) * JDST(JCD,18)) +   ;
                   (J19FR(2,Spd[idx]) * JDST(JCD,19)) +   ;
                   (J20FR(2,Spd[idx]) * JDST(JCD,20)) +   ;
                   (J21FR(2,Spd[idx]) * JDST(JCD,21)) +   ;
                   (J22FR(2,Spd[idx]) * JDST(JCD,22)) +   ;
                   (J23FR(2,Spd[idx]) * JDST(JCD,23)) +   ;
                   (J24FR(2,Spd[idx]) * JDST(JCD,24)) +   ;
                   (J25FR(2,Spd[idx]) * JDST(JCD,25)) +   ;
                   (J26FR(2,Spd[idx]) * JDST(JCD,26)) +   ;
                   (J27FR(2,Spd[idx]) * JDST(JCD,27))     ;


        CNR[IDX] = (J01FR(3,Spd[idx]) * JDST(JCD,1))  +   ; COMPOSITE
                   (J02FR(3,Spd[idx]) * JDST(JCD,2))  +   ; Nx FREEWAY
                   (J03FR(3,Spd[idx]) * JDST(JCD,3))  +   ; RATE
                   (J04FR(3,Spd[idx]) * JDST(JCD,4))  +   ;
```

```
                  (J05FR(3,Spd[idx]) * JDST(JCD,5))  +   ;
                  (J06FR(3,Spd[idx]) * JDST(JCD,6))  +   ;
                  (J07FR(3,Spd[idx]) * JDST(JCD,7))  +   ;
                  (J08FR(3,Spd[idx]) * JDST(JCD,8))  +   ;
                  (J09FR(3,Spd[idx]) * JDST(JCD,9))  +   ;
                  (J10FR(3,Spd[idx]) * JDST(JCD,10)) +   ;
                  (J11FR(3,Spd[idx]) * JDST(JCD,11)) +   ;
                  (J12FR(3,Spd[idx]) * JDST(JCD,12)) +   ;
                  (J13FR(3,Spd[idx]) * JDST(JCD,13)) +   ;
                  (J14FR(3,Spd[idx]) * JDST(JCD,14)) +   ;
                  (J15FR(3,Spd[idx]) * JDST(JCD,15)) +   ;
                  (J16FR(3,Spd[idx]) * JDST(JCD,16)) +   ;
                  (J17FR(3,Spd[idx]) * JDST(JCD,17)) +   ;
                  (J18FR(3,Spd[idx]) * JDST(JCD,18)) +   ;
                  (J19FR(3,Spd[idx]) * JDST(JCD,19)) +   ;
                  (J20FR(3,Spd[idx]) * JDST(JCD,20)) +   ;
                  (J21FR(3,Spd[idx]) * JDST(JCD,21)) +   ;
                  (J22FR(3,Spd[idx]) * JDST(JCD,22)) +   ;
                  (J23FR(3,Spd[idx]) * JDST(JCD,23)) +   ;
                  (J24FR(3,Spd[idx]) * JDST(JCD,24)) +   ;
                  (J25FR(3,Spd[idx]) * JDST(JCD,25)) +   ;
                  (J26FR(3,Spd[idx]) * JDST(JCD,26)) +   ;
                  (J27FR(3,Spd[idx]) * JDST(JCD,27))     ;


         HEM[IDX] =  CHR[idx]  * VMT[idx]          ; HC Emissions (gm)
         CEM[IDX] =  CCR[idx]  * VMT[idx]          ; HC Emissions (gm)
         NEM[IDX] =  CNR[idx]  * VMT[idx]          ; HC Emissions (gm)


      ENDLOOP
   ENDIF

; ---------------------------------------------------------------------

if (lw.ftype > 1 && lw.ftype < 5)
      LOOP IDX= 1,24
         CHR[IDX] = (J01AR(1,Spd[idx]) * JDST(JCD,1))  +   ; COMPOSITE
                    (J02AR(1,Spd[idx]) * JDST(JCD,2))  +   ; HC Arterial
                    (J03AR(1,Spd[idx]) * JDST(JCD,3))  +   ; RATE
                    (J04AR(1,Spd[idx]) * JDST(JCD,4))  +   ;
                    (J05AR(1,Spd[idx]) * JDST(JCD,5))  +   ;
                    (J06AR(1,Spd[idx]) * JDST(JCD,6))  +   ;
                    (J07AR(1,Spd[idx]) * JDST(JCD,7))  +   ;
                    (J08AR(1,Spd[idx]) * JDST(JCD,8))  +   ;
                    (J09AR(1,Spd[idx]) * JDST(JCD,9))  +   ;
                    (J10AR(1,Spd[idx]) * JDST(JCD,10)) +   ;
                    (J11AR(1,Spd[idx]) * JDST(JCD,11)) +   ;
                    (J12AR(1,Spd[idx]) * JDST(JCD,12)) +   ;
                    (J13AR(1,Spd[idx]) * JDST(JCD,13)) +   ;
                    (J14AR(1,Spd[idx]) * JDST(JCD,14)) +   ;
                    (J15AR(1,Spd[idx]) * JDST(JCD,15)) +   ;
                    (J16AR(1,Spd[idx]) * JDST(JCD,16)) +   ;
                    (J17AR(1,Spd[idx]) * JDST(JCD,17)) +   ;
                    (J18AR(1,Spd[idx]) * JDST(JCD,18)) +   ;
                    (J19AR(1,Spd[idx]) * JDST(JCD,19)) +   ;
                    (J20AR(1,Spd[idx]) * JDST(JCD,20)) +   ;
                    (J21AR(1,Spd[idx]) * JDST(JCD,21)) +   ;
                    (J22AR(1,Spd[idx]) * JDST(JCD,22)) +   ;
                    (J23AR(1,Spd[idx]) * JDST(JCD,23)) +   ;
                    (J24AR(1,Spd[idx]) * JDST(JCD,24)) +   ;
                    (J25AR(1,Spd[idx]) * JDST(JCD,25)) +   ;
                    (J26AR(1,Spd[idx]) * JDST(JCD,26)) +   ;
                    (J27AR(1,Spd[idx]) * JDST(JCD,27))     ;


         CCR[IDX] = (J01AR(2,Spd[idx]) * JDST(JCD,1))  +   ; COMPOSITE
                    (J02AR(2,Spd[idx]) * JDST(JCD,2))  +   ; CO Arterial
                    (J03AR(2,Spd[idx]) * JDST(JCD,3))  +   ; RATE
                    (J04AR(2,Spd[idx]) * JDST(JCD,4))  +   ;
                    (J05AR(2,Spd[idx]) * JDST(JCD,5))  +   ;
                    (J06AR(2,Spd[idx]) * JDST(JCD,6))  +   ;
                    (J07AR(2,Spd[idx]) * JDST(JCD,7))  +   ;
                    (J08AR(2,Spd[idx]) * JDST(JCD,8))  +   ;
                    (J09AR(2,Spd[idx]) * JDST(JCD,9))  +   ;
                    (J10AR(2,Spd[idx]) * JDST(JCD,10)) +   ;
                    (J11AR(2,Spd[idx]) * JDST(JCD,11)) +   ;
                    (J12AR(2,Spd[idx]) * JDST(JCD,12)) +   ;
                    (J13AR(2,Spd[idx]) * JDST(JCD,13)) +   ;
                    (J14AR(2,Spd[idx]) * JDST(JCD,14)) +   ;
                    (J15AR(2,Spd[idx]) * JDST(JCD,15)) +   ;
                    (J16AR(2,Spd[idx]) * JDST(JCD,16)) +   ;
                    (J17AR(2,Spd[idx]) * JDST(JCD,17)) +   ;
                    (J18AR(2,Spd[idx]) * JDST(JCD,18)) +   ;
                    (J19AR(2,Spd[idx]) * JDST(JCD,19)) +   ;
                    (J20AR(2,Spd[idx]) * JDST(JCD,20)) +   ;
                    (J21AR(2,Spd[idx]) * JDST(JCD,21)) +   ;
                    (J22AR(2,Spd[idx]) * JDST(JCD,22)) +   ;
                    (J23AR(2,Spd[idx]) * JDST(JCD,23)) +   ;
                    (J24AR(2,Spd[idx]) * JDST(JCD,24)) +   ;
                    (J25AR(2,Spd[idx]) * JDST(JCD,25)) +   ;
                    (J26AR(2,Spd[idx]) * JDST(JCD,26)) +   ;
                    (J27AR(2,Spd[idx]) * JDST(JCD,27))     ;


         CNR[IDX] = (J01AR(3,Spd[idx]) * JDST(JCD,1))  +   ; COMPOSITE
                    (J02AR(3,Spd[idx]) * JDST(JCD,2))  +   ; Nx ARTERIAL
                    (J03AR(3,Spd[idx]) * JDST(JCD,3))  +   ; RATE
                    (J04AR(3,Spd[idx]) * JDST(JCD,4))  +   ;
                    (J05AR(3,Spd[idx]) * JDST(JCD,5))  +   ;
                    (J06AR(3,Spd[idx]) * JDST(JCD,6))  +   ;
                    (J07AR(3,Spd[idx]) * JDST(JCD,7))  +   ;
                    (J08AR(3,Spd[idx]) * JDST(JCD,8))  +   ;
                    (J09AR(3,Spd[idx]) * JDST(JCD,9))  +   ;
                    (J10AR(3,Spd[idx]) * JDST(JCD,10)) +   ;
                    (J11AR(3,Spd[idx]) * JDST(JCD,11)) +   ;
                    (J12AR(3,Spd[idx]) * JDST(JCD,12)) +   ;
                    (J13AR(3,Spd[idx]) * JDST(JCD,13)) +   ;
                    (J14AR(3,Spd[idx]) * JDST(JCD,14)) +   ;
                    (J15AR(3,Spd[idx]) * JDST(JCD,15)) +   ;
                    (J16AR(3,Spd[idx]) * JDST(JCD,16)) +   ;
                    (J17AR(3,Spd[idx]) * JDST(JCD,17)) +   ;
                    (J18AR(3,Spd[idx]) * JDST(JCD,18)) +   ;
                    (J19AR(3,Spd[idx]) * JDST(JCD,19)) +   ;
                    (J20AR(3,Spd[idx]) * JDST(JCD,20)) +   ;
                    (J21AR(3,Spd[idx]) * JDST(JCD,21)) +   ;
                    (J22AR(3,Spd[idx]) * JDST(JCD,22)) +   ;
                    (J23AR(3,Spd[idx]) * JDST(JCD,23)) +   ;
                    (J24AR(3,Spd[idx]) * JDST(JCD,24)) +   ;
                    (J25AR(3,Spd[idx]) * JDST(JCD,25)) +   ;
                    (J26AR(3,Spd[idx]) * JDST(JCD,26)) +   ;
                    (J27AR(3,Spd[idx]) * JDST(JCD,27))     ;


         HEM[IDX] =  CHR[idx]  * VMT[idx]          ; HC Emissions (gm)
         CEM[IDX] =  CCR[idx]  * VMT[idx]          ; HC Emissions (gm)
         NEM[IDX] =  CNR[idx]  * VMT[idx]          ; HC Emissions (gm)


      ENDLOOP
   ENDIF

   ; ---------------------------------------------------------------------
```

```
if (lw.ftype = 6 )
   LOOP IDX= 1,24
        SPD[IDX] = 35         ; Make RAMP SPEEDS EQUAL to 35 MPH
                              ; Per the lookup table
        CHR[IDX] = (J01RR(1,Spd[idx]) * JDST(JCD,1))  +   ; COMPOSITE
                   (J02RR(1,Spd[idx]) * JDST(JCD,2))  +   ; HC Arterial
                   (J03RR(1,Spd[idx]) * JDST(JCD,3))  +   ; RATE
                   (J04RR(1,Spd[idx]) * JDST(JCD,4))  +   ;
                   (J05RR(1,Spd[idx]) * JDST(JCD,5))  +   ;
                   (J06RR(1,Spd[idx]) * JDST(JCD,6))  +   ;
                   (J07RR(1,Spd[idx]) * JDST(JCD,7))  +   ;
                   (J08RR(1,Spd[idx]) * JDST(JCD,8))  +   ;
                   (J09RR(1,Spd[idx]) * JDST(JCD,9))  +   ;
                   (J10RR(1,Spd[idx]) * JDST(JCD,10)) +   ;
                   (J11RR(1,Spd[idx]) * JDST(JCD,11)) +   ;
                   (J12RR(1,Spd[idx]) * JDST(JCD,12)) +   ;
                   (J13RR(1,Spd[idx]) * JDST(JCD,13)) +   ;
                   (J14RR(1,Spd[idx]) * JDST(JCD,14)) +   ;
                   (J15RR(1,Spd[idx]) * JDST(JCD,15)) +   ;
                   (J16RR(1,Spd[idx]) * JDST(JCD,16)) +   ;
                   (J17RR(1,Spd[idx]) * JDST(JCD,17)) +   ;
                   (J18RR(1,Spd[idx]) * JDST(JCD,18)) +   ;
                   (J19RR(1,Spd[idx]) * JDST(JCD,19)) +   ;
                   (J20RR(1,Spd[idx]) * JDST(JCD,20)) +   ;
                   (J21RR(1,Spd[idx]) * JDST(JCD,21)) +   ;
                   (J22RR(1,Spd[idx]) * JDST(JCD,22)) +   ;
                   (J23RR(1,Spd[idx]) * JDST(JCD,23)) +   ;
                   (J24RR(1,Spd[idx]) * JDST(JCD,24)) +   ;
                   (J25RR(1,Spd[idx]) * JDST(JCD,25)) +   ;
                   (J26RR(1,Spd[idx]) * JDST(JCD,26)) +   ;
                   (J27RR(1,Spd[idx]) * JDST(JCD,27))     ;


        CCR[IDX] = (J01RR(2,Spd[idx]) * JDST(JCD,1))  +   ; COMPOSITE
                   (J02RR(2,Spd[idx]) * JDST(JCD,2))  +   ; CO Arterial
                   (J03RR(2,Spd[idx]) * JDST(JCD,3))  +   ; RATE
                   (J04RR(2,Spd[idx]) * JDST(JCD,4))  +   ;
                   (J05RR(2,Spd[idx]) * JDST(JCD,5))  +   ;
                   (J06RR(2,Spd[idx]) * JDST(JCD,6))  +   ;
                   (J07RR(2,Spd[idx]) * JDST(JCD,7))  +   ;
                   (J08RR(2,Spd[idx]) * JDST(JCD,8))  +   ;
                   (J09RR(2,Spd[idx]) * JDST(JCD,9))  +   ;
                   (J10RR(2,Spd[idx]) * JDST(JCD,10)) +   ;
                   (J11RR(2,Spd[idx]) * JDST(JCD,11)) +   ;
                   (J12RR(2,Spd[idx]) * JDST(JCD,12)) +   ;
                   (J13RR(2,Spd[idx]) * JDST(JCD,13)) +   ;
                   (J14RR(2,Spd[idx]) * JDST(JCD,14)) +   ;
                   (J15RR(2,Spd[idx]) * JDST(JCD,15)) +   ;
                   (J16RR(2,Spd[idx]) * JDST(JCD,16)) +   ;
                   (J17RR(2,Spd[idx]) * JDST(JCD,17)) +   ;
                   (J18RR(2,Spd[idx]) * JDST(JCD,18)) +   ;
                   (J19RR(2,Spd[idx]) * JDST(JCD,19)) +   ;
                   (J20RR(2,Spd[idx]) * JDST(JCD,20)) +   ;
                   (J21RR(2,Spd[idx]) * JDST(JCD,21)) +   ;
                   (J22RR(2,Spd[idx]) * JDST(JCD,22)) +   ;
                   (J23RR(2,Spd[idx]) * JDST(JCD,23)) +   ;
                   (J24RR(2,Spd[idx]) * JDST(JCD,24)) +   ;
                   (J25RR(2,Spd[idx]) * JDST(JCD,25)) +   ;
                   (J26RR(2,Spd[idx]) * JDST(JCD,26)) +   ;
                   (J27RR(2,Spd[idx]) * JDST(JCD,27))     ;


        CNR[IDX] = (J01RR(3,Spd[idx]) * JDST(JCD,1))  +   ; COMPOSITE
                   (J02RR(3,Spd[idx]) * JDST(JCD,2))  +   ; Nx ARTERIAL
                   (J03RR(3,Spd[idx]) * JDST(JCD,3))  +   ; RATE
                   (J04RR(3,Spd[idx]) * JDST(JCD,4))  +   ;
                   (J05RR(3,Spd[idx]) * JDST(JCD,5))  +   ;
                   (J06RR(3,Spd[idx]) * JDST(JCD,6))  +   ;
                   (J07RR(3,Spd[idx]) * JDST(JCD,7))  +   ;
                   (J08RR(3,Spd[idx]) * JDST(JCD,8))  +   ;
                   (J09RR(3,Spd[idx]) * JDST(JCD,9))  +   ;
                   (J10RR(3,Spd[idx]) * JDST(JCD,10)) +   ;
                   (J11RR(3,Spd[idx]) * JDST(JCD,11)) +   ;
                   (J12RR(3,Spd[idx]) * JDST(JCD,12)) +   ;
                   (J13RR(3,Spd[idx]) * JDST(JCD,13)) +   ;
                   (J14RR(3,Spd[idx]) * JDST(JCD,14)) +   ;
                   (J15RR(3,Spd[idx]) * JDST(JCD,15)) +   ;
                   (J16RR(3,Spd[idx]) * JDST(JCD,16)) +   ;
                   (J17RR(3,Spd[idx]) * JDST(JCD,17)) +   ;
                   (J18RR(3,Spd[idx]) * JDST(JCD,18)) +   ;
                   (J19RR(3,Spd[idx]) * JDST(JCD,19)) +   ;
                   (J20RR(3,Spd[idx]) * JDST(JCD,20)) +   ;
                   (J21RR(3,Spd[idx]) * JDST(JCD,21)) +   ;
                   (J22RR(3,Spd[idx]) * JDST(JCD,22)) +   ;
                   (J23RR(3,Spd[idx]) * JDST(JCD,23)) +   ;
                   (J24RR(3,Spd[idx]) * JDST(JCD,24)) +   ;
                   (J25RR(3,Spd[idx]) * JDST(JCD,25)) +   ;
                   (J26RR(3,Spd[idx]) * JDST(JCD,26)) +   ;
                   (J27RR(3,Spd[idx]) * JDST(JCD,27))     ;


        HEM[IDX] =  CHR[idx]  * VMT[idx]          ; HC Emissions (gm)
        CEM[IDX] =  CCR[idx]  * VMT[idx]          ; HC Emissions (gm)
        NEM[IDX] =  CNR[idx]  * VMT[idx]          ; HC Emissions (gm)

     ENDLOOP
  ENDIF


 ;****** debug section - dumping link-level outputs for selected links
  if (a=3200-3300)
      list = '     A     B  hr spd jcd  ft     vmt  hc_r  co_r  nx_R      HC_E      CO_E
 N_E',
             file=running.dbg
      list = '------ ----- --- --- --- --- ------- ----- ----- -----   -------   -------   --
 -----',
             file=running.dbg
   LOOP IDX= 1,24
      list = a(6), b(6), idx(4), spd[idx](4),jcd(4),lw.ftype(4),
      VMT[idx](8.2), chr[idx](6.3),' ',ccr[idx](6.3),' ',cnr[idx](6.3),' ',
             HEM[idx](10.2),CEM[idx](10.2),NEM[idx](10.2),
             file=running.dbg
   ENDLOOP
  endif
 ;******  end of debug section


 ;-----------------------------------------------------------------
 ; Accumulate total,hourly HC, CO, NOX emissions and VMT          -            -
 ; Reset All Hourly Arrays before going to the next link          -
 ;-----------------------------------------------------------------
 ;

        HEM_01   = HEM_01  + HEM[01]   ; Accumulated HC Emiss. Hour 01
        CEM_01   = CEM_01  + CEM[01]   ; Accumulated CO Emiss. Hour 01
        NEM_01   = NEM_01  + NEM[01]   ; Accumulated Nx Emiss. Hour 01
        VMT_01   = VMT_01  + vmt[01]   ; Accumulated VMT       Hour 01

        HEM_02   = HEM_02  + HEM[02]   ; Accumulated HC Emiss. Hour 02
        CEM_02   = CEM_02  + CEM[02]   ; Accumulated CO Emiss. Hour 02
        NEM_02   = NEM_02  + NEM[02]   ; Accumulated Nx Emiss. Hour 02
        VMT_02   = VMT_02  + vmt[02]   ; Accumulated VMT       Hour 02

        HEM_03   = HEM_03  + HEM[03]   ; Accumulated HC Emiss. Hour 03
        CEM_03   = CEM_03  + CEM[03]   ; Accumulated CO Emiss. Hour 03
```

```
NEM_03   = NEM_03  + NEM[03]   ; Accumulated Nx Emiss. Hour 03
VMT_03   = VMT_03  + vmt[03]   ; Accumulated VMT       Hour 03

HEM_04   = HEM_04  + HEM[04]   ; Accumulated HC Emiss. Hour 04
CEM_04   = CEM_04  + CEM[04]   ; Accumulated CO Emiss. Hour 04
NEM_04   = NEM_04  + NEM[04]   ; Accumulated Nx Emiss. Hour 04
VMT_04   = VMT_04  + vmt[04]   ; Accumulated VMT       Hour 04

HEM_05   = HEM_05  + HEM[05]   ; Accumulated HC Emiss. Hour 05
CEM_05   = CEM_05  + CEM[05]   ; Accumulated CO Emiss. Hour 05
NEM_05   = NEM_05  + NEM[05]   ; Accumulated Nx Emiss. Hour 05
VMT_05   = VMT_05  + vmt[05]   ; Accumulated VMT       Hour 05

HEM_06   = HEM_06  + HEM[06]   ; Accumulated HC Emiss. Hour 06
CEM_06   = CEM_06  + CEM[06]   ; Accumulated CO Emiss. Hour 06
NEM_06   = NEM_06  + NEM[06]   ; Accumulated Nx Emiss. Hour 06
VMT_06   = VMT_06  + vmt[06]   ; Accumulated VMT       Hour 06

HEM_07   = HEM_07  + HEM[07]   ; Accumulated HC Emiss. Hour 07
CEM_07   = CEM_07  + CEM[07]   ; Accumulated CO Emiss. Hour 07
NEM_07   = NEM_07  + NEM[07]   ; Accumulated Nx Emiss. Hour 07
VMT_07   = VMT_07  + vmt[07]   ; Accumulated VMT       Hour 07

HEM_08   = HEM_08  + HEM[08]   ; Accumulated HC Emiss. Hour 08
CEM_08   = CEM_08  + CEM[08]   ; Accumulated CO Emiss. Hour 08
NEM_08   = NEM_08  + NEM[08]   ; Accumulated Nx Emiss. Hour 08
VMT_08   = VMT_08  + vmt[08]   ; Accumulated VMT       Hour 08

HEM_09   = HEM_09  + HEM[09]   ; Accumulated HC Emiss. Hour 09
CEM_09   = CEM_09  + CEM[09]   ; Accumulated CO Emiss. Hour 09
NEM_09   = NEM_09  + NEM[09]   ; Accumulated Nx Emiss. Hour 09
VMT_09   = VMT_09  + vmt[09]   ; Accumulated VMT       Hour 09

HEM_10   = HEM_10  + HEM[10]   ; Accumulated HC Emiss. Hour 10
CEM_10   = CEM_10  + CEM[10]   ; Accumulated CO Emiss. Hour 10
NEM_10   = NEM_10  + NEM[10]   ; Accumulated Nx Emiss. Hour 10
VMT_10   = VMT_10  + vmt[10]   ; Accumulated VMT       Hour 10

HEM_11   = HEM_11  + HEM[11]   ; Accumulated HC Emiss. Hour 11
CEM_11   = CEM_11  + CEM[11]   ; Accumulated CO Emiss. Hour 11
NEM_11   = NEM_11  + NEM[11]   ; Accumulated Nx Emiss. Hour 11
VMT_11   = VMT_11  + vmt[11]   ; Accumulated VMT       Hour 11

HEM_12   = HEM_12  + HEM[12]   ; Accumulated HC Emiss. Hour 12
CEM_12   = CEM_12  + CEM[12]   ; Accumulated CO Emiss. Hour 12
NEM_12   = NEM_12  + NEM[12]   ; Accumulated Nx Emiss. Hour 12
VMT_12   = VMT_12  + vmt[12]   ; Accumulated VMT       Hour 12

HEM_13   = HEM_13  + HEM[13]   ; Accumulated HC Emiss. Hour 13
CEM_13   = CEM_13  + CEM[13]   ; Accumulated CO Emiss. Hour 13
NEM_13   = NEM_13  + NEM[13]   ; Accumulated Nx Emiss. Hour 13
VMT_13   = VMT_13  + vmt[13]   ; Accumulated VMT       Hour 13

HEM_14   = HEM_14  + HEM[14]   ; Accumulated HC Emiss. Hour 14
CEM_14   = CEM_14  + CEM[14]   ; Accumulated CO Emiss. Hour 14
NEM_14   = NEM_14  + NEM[14]   ; Accumulated Nx Emiss. Hour 14
VMT_14   = VMT_14  + vmt[14]   ; Accumulated VMT       Hour 14

HEM_15   = HEM_15  + HEM[15]   ; Accumulated HC Emiss. Hour 15
CEM_15   = CEM_15  + CEM[15]   ; Accumulated CO Emiss. Hour 15
NEM_15   = NEM_15  + NEM[15]   ; Accumulated Nx Emiss. Hour 15
VMT_15   = VMT_15  + vmt[15]   ; Accumulated VMT       Hour 15

HEM_16   = HEM_16  + HEM[16]   ; Accumulated HC Emiss. Hour 16
CEM_16   = CEM_16  + CEM[16]   ; Accumulated CO Emiss. Hour 16
NEM_16   = NEM_16  + NEM[16]   ; Accumulated Nx Emiss. Hour 16
VMT_16   = VMT_16  + vmt[16]   ; Accumulated VMT       Hour 16

HEM_17   = HEM_17  + HEM[17]   ; Accumulated HC Emiss. Hour 17
CEM_17   = CEM_17  + CEM[17]   ; Accumulated CO Emiss. Hour 17
NEM_17   = NEM_17  + NEM[17]   ; Accumulated Nx Emiss. Hour 17
VMT_17   = VMT_17  + vmt[17]   ; Accumulated VMT       Hour 17

HEM_18   = HEM_18  + HEM[18]   ; Accumulated HC Emiss. Hour 18
CEM_18   = CEM_18  + CEM[18]   ; Accumulated CO Emiss. Hour 18
NEM_18   = NEM_18  + NEM[18]   ; Accumulated Nx Emiss. Hour 18
VMT_18   = VMT_18  + vmt[18]   ; Accumulated VMT       Hour 18

HEM_19   = HEM_19  + HEM[19]   ; Accumulated HC Emiss. Hour 19
CEM_19   = CEM_19  + CEM[19]   ; Accumulated CO Emiss. Hour 19
NEM_19   = NEM_19  + NEM[19]   ; Accumulated Nx Emiss. Hour 19
VMT_19   = VMT_19  + vmt[19]   ; Accumulated VMT       Hour 19

HEM_20   = HEM_20  + HEM[20]   ; Accumulated HC Emiss. Hour 20
CEM_20   = CEM_20  + CEM[20]   ; Accumulated CO Emiss. Hour 20
NEM_20   = NEM_20  + NEM[20]   ; Accumulated Nx Emiss. Hour 20
VMT_20   = VMT_20  + vmt[20]   ; Accumulated VMT       Hour 20

HEM_21   = HEM_21  + HEM[21]   ; Accumulated HC Emiss. Hour 21
CEM_21   = CEM_21  + CEM[21]   ; Accumulated CO Emiss. Hour 21
NEM_21   = NEM_21  + NEM[21]   ; Accumulated Nx Emiss. Hour 21
VMT_21   = VMT_21  + vmt[21]   ; Accumulated VMT       Hour 21

HEM_22   = HEM_22  + HEM[22]   ; Accumulated HC Emiss. Hour 22
CEM_22   = CEM_22  + CEM[22]   ; Accumulated CO Emiss. Hour 22
NEM_22   = NEM_22  + NEM[22]   ; Accumulated Nx Emiss. Hour 22
VMT_22   = VMT_22  + vmt[22]   ; Accumulated VMT       Hour 22

HEM_23   = HEM_23  + HEM[23]   ; Accumulated HC Emiss. Hour 23
CEM_23   = CEM_23  + CEM[23]   ; Accumulated CO Emiss. Hour 23
NEM_23   = NEM_23  + NEM[23]   ; Accumulated Nx Emiss. Hour 23
VMT_23   = VMT_23  + vmt[23]   ; Accumulated VMT       Hour 23

HEM_24   = HEM_24  + HEM[24]   ; Accumulated HC Emiss. Hour 24
CEM_24   = CEM_24  + CEM[24]   ; Accumulated CO Emiss. Hour 24
NEM_24   = NEM_24  + NEM[24]   ; Accumulated Nx Emiss. Hour 24
VMT_24   = VMT_24  + vmt[24]   ; Accumulated VMT       Hour 24


  LOOP IDX= 1,24
       HEMTOT   = HEMTOT  + HEM[IDX]   ; Accumulated HC Emiss. Total
       CEMTOT   = CEMTOT  + CEM[IDX]   ; Accumulated CO Emiss. Total
       NEMTOT   = NEMTOT  + NEM[IDX]   ; Accumulated Nx Emiss. Total
       VMTTOT   = VMTTOT  + vmt[IDX]   ; Accumulated VMT       Total
  ENDLOOP



;---------------------------------------------------------------------
; Finally, write out all the link information
;
;---------------------------------------------------------------------


;
PRINT FORM=8.0,
 list= a(6),b(6),
     lw.ftype, lw.jur, jcd,
     file=linkout.INI

PRINT FORM=3.0,
 list= a(6),b(6),
  spd[1], spd[2], spd[3], spd[4], spd[5], spd[6], spd[7], spd[8],
  spd[9],spd[10],spd[11],spd[12],spd[13],spd[14],spd[15],spd[16],
 spd[17],spd[18],spd[19],spd[20],spd[21],spd[22],spd[23],spd[24],
 file=linkout.SPD
```

```
PRINT FORM=10.2,
 list= a(6),b(6),
  vmt[1], vmt[2], vmt[3], vmt[4], vmt[5], vmt[6], vmt[7], vmt[8],
  vmt[9],vmt[10],vmt[11],vmt[12],vmt[13],vmt[14],vmt[15],vmt[16],
 vmt[17],vmt[18],vmt[19],vmt[20],vmt[21],vmt[22],vmt[23],vmt[24],
 file=linkout.VMT

PRINT FORM=10.2,
 list= a(6),b(6),
  hem[1], hem[2], hem[3], hem[4], hem[5], hem[6], hem[7], hem[8],
  hem[9],hem[10],hem[11],hem[12],hem[13],hem[14],hem[15],hem[16],
 hem[17],hem[18],hem[19],hem[20],hem[21],hem[22],hem[23],hem[24],
 file=linkout.hem

PRINT FORM=10.2,
 list= a(6),b(6),
  cem[1], cem[2], cem[3], cem[4], cem[5], cem[6], cem[7], cem[8],
  cem[9],cem[10],cem[11],cem[12],cem[13],cem[14],cem[15],cem[16],
 cem[17],cem[18],cem[19],cem[20],cem[21],cem[22],cem[23],cem[24],
 file=linkout.cem

PRINT FORM=10.2,
 list= a(6),b(6),
  nem[1], nem[2], nem[3], nem[4], nem[5], nem[6], nem[7], nem[8],
  nem[9],nem[10],nem[11],nem[12],nem[13],nem[14],nem[15],nem[16],
 nem[17],nem[18],nem[19],nem[20],nem[21],nem[22],nem[23],nem[24],
 file=linkout.nem
;-------------------------------------------------------------------------
; End of link write section
;
;-------------------------------------------------------------------------



ENDIF ;  <<-----  END OF Link Selection Criterea loop
ENDLINKLOOP  ;  End of linkloop

;
; Now, List out the total emissions by Hour
;
 LIST = 'Summary of HC, CO, Nx Emissions(gms), by Hour '
 List ='  '
 print form= 15.2cs  LIST = 'Hour 01: ', HEM_01, CEM_01, NEM_01,VMT_01
 print form= 15.2cs  LIST = 'Hour 02: ', HEM_02, CEM_02, NEM_02,VMT_02
 print form= 15.2cs  LIST = 'Hour 03: ', HEM_03, CEM_03, NEM_03,VMT_03
 print form= 15.2cs  LIST = 'Hour 04: ', HEM_04, CEM_04, NEM_04,VMT_04
 print form= 15.2cs  LIST = 'Hour 05: ', HEM_05, CEM_05, NEM_05,VMT_05
 print form= 15.2cs  LIST = 'Hour 06: ', HEM_06, CEM_06, NEM_06,VMT_06
 print form= 15.2cs  LIST = 'Hour 07: ', HEM_07, CEM_07, NEM_07,VMT_07
 print form= 15.2cs  LIST = 'Hour 08: ', HEM_08, CEM_08, NEM_08,VMT_08
 print form= 15.2cs  LIST = 'Hour 09: ', HEM_09, CEM_09, NEM_09,VMT_09
 print form= 15.2cs  LIST = 'Hour 10: ', HEM_10, CEM_10, NEM_10,VMT_10
 print form= 15.2cs  LIST = 'Hour 11: ', HEM_11, CEM_11, NEM_11,VMT_11
 print form= 15.2cs  LIST = 'Hour 12: ', HEM_12, CEM_12, NEM_12,VMT_12
 print form= 15.2cs  LIST = 'Hour 13: ', HEM_13, CEM_13, NEM_13,VMT_13
 print form= 15.2cs  LIST = 'Hour 14: ', HEM_14, CEM_14, NEM_14,VMT_14
 print form= 15.2cs  LIST = 'Hour 15: ', HEM_15, CEM_15, NEM_15,VMT_15
 print form= 15.2cs  LIST = 'Hour 16: ', HEM_16, CEM_16, NEM_16,VMT_16
 print form= 15.2cs  LIST = 'Hour 17: ', HEM_17, CEM_17, NEM_17,VMT_17
 print form= 15.2cs  LIST = 'Hour 18: ', HEM_18, CEM_18, NEM_18,VMT_18
 print form= 15.2cs  LIST = 'Hour 19: ', HEM_19, CEM_19, NEM_19,VMT_19
 print form= 15.2cs  LIST = 'Hour 20: ', HEM_20, CEM_20, NEM_20,VMT_20
 print form= 15.2cs  LIST = 'Hour 21: ', HEM_21, CEM_21, NEM_21,VMT_21
 print form= 15.2cs  LIST = 'Hour 22: ', HEM_22, CEM_22, NEM_22,VMT_22
 print form= 15.2cs  LIST = 'Hour 23: ', HEM_23, CEM_23, NEM_23,VMT_23
 print form= 15.2cs  LIST = 'Hour 24: ', HEM_24, CEM_24, NEM_24,VMT_24
 List ='  '
 print form= 15.2cs,
 LIST=                      'SUM ',HEMTOT(20.2),'          ',NEMTOT(20.2)
 print form= 15.2cs,
 LIST =   '                   ', CEMTOT(20.2),'        ', VMTTOT(20.2)

endphase

endrun

;-------------------------
run pgm=hwynet
    neti     = @innet@
    neto     = dummy.net

    linki[2] = linkout.vmt  var = a, b,
    vmt01 vmt02,vmt03,vmt04,vmt05,vmt06,vmt07,vmt08,vmt09,vmt10,
    vmt11 vmt12,vmt13,vmt14,vmt15,vmt16,vmt17,vmt18,vmt19,vmt20,
    vmt21 vmt22,vmt23,vmt24

    linki[3] = linkout.spd var = a, b,
    spd01 spd02,spd03,spd04,spd05,spd06,spd07,spd08,spd09,spd10,
    spd11 spd12,spd13,spd14,spd15,spd16,spd17,spd18,spd19,spd20,
    spd21 spd22,spd23,spd24

    linki[4] = linkout.hem var = a, b,
    hem01 hem02,hem03,hem04,hem05,hem06,hem07,hem08,hem09,hem10,
    hem11 hem12,hem13,hem14,hem15,hem16,hem17,hem18,hem19,hem20,
    hem21 hem22,hem23,hem24

    linki[5] = linkout.cem var = a, b,
    cem01 cem02,cem03,cem04,cem05,cem06,cem07,cem08,cem09,cem10,
    cem11 cem12,cem13,cem14,cem15,cem16,cem17,cem18,cem19,cem20,
    cem21 cem22,cem23,cem24

    linki[6] = linkout.nem var = a, b,
    nem01 nem02,nem03,nem04,nem05,nem06,nem07,nem08,nem09,nem10,
    nem11 nem12,nem13,nem14,nem15,nem16,nem17,nem18,nem19,nem20,
    nem21 nem22,nem23,nem24

; -------------------------------------------------------------
 if (i624vol > 0 && FType >0)
           _cnt=1
 else
           _cnt =0
 endif
; define MSA/NonMSA
; MSA= dc,mtg,pg,arl,alx,ffx,ldn,pw,frd,chs,cal,stf
 if (JUR=0-7 || jur = 9 || jur=12 || jur=15 || jur=19)
           msa= 1
 else
           msa= 0
 endif

  VMT =
    vmt01+vmt02+vmt03+vmt04+vmt05+vmt06+vmt07+vmt08+vmt09+vmt10+
    vmt11+vmt12+vmt13+vmt14+vmt15+vmt16+vmt17+vmt18+vmt19+vmt20+
    vmt21+vmt22+vmt23+vmt24

 HEM =
    hem01+hem02+hem03+hem04+hem05+hem06+hem07+hem08+hem09+hem10+
    hem11+hem12+hem13+hem14+hem15+hem16+hem17+hem18+hem19+hem20+
    hem21+hem22+hem23+hem24

 CEM =
    cem01+cem02+cem03+cem04+cem05+cem06+cem07+cem08+cem09+cem10+
    cem11+cem12+cem13+cem14+cem15+cem16+cem17+cem18+cem19+cem20+
    cem21+cem22+cem23+cem24
```

```
 NEM =
    nem01+nem02+nem03+nem04+nem05+nem06+nem07+nem08+nem09+nem10+
    nem11+nem12+nem13+nem14+nem15+nem16+nem17+nem18+nem19+nem20+
    nem21+nem22+nem23+nem24


  wSPD=
      vmt01*spd01 + vmt02*spd02 + vmt03*spd03 + vmt04*spd04 +
      vmt05*spd05 + vmt06*spd06 + vmt07*spd07 + vmt08*spd08 +
      vmt09*spd09 + vmt10*spd10 + vmt11*spd11 + vmt12*spd12 +
      vmt13*spd13 + vmt14*spd14 + vmt15*spd15 + vmt16*spd16 +
      vmt17*spd17 + vmt18*spd18 + vmt19*spd19 + vmt20*spd20 +
      vmt21*spd21 + vmt22*spd22 + vmt23*spd23 + vmt24*spd24

IF (VMT = 0)
    SPD = 0
ELSE
    SPD = Wspd / vmt ;  daily weighted link speed
ENDIF

IF     (SPD <  2.5)
    spdgroup=    0
ELSEIF (SPD <  7.5)
    spdgroup=    5
ELSEIF (SPD < 12.5)
    spdgroup=   10
ELSEIF (SPD < 17.5)
    spdgroup=   15
ELSEIF (SPD < 22.5)
    spdgroup=   20
ELSEIF (SPD < 27.5)
    spdgroup=   25
ELSEIF (SPD < 32.5)
    spdgroup=   30
ELSEIF (SPD < 37.5)
    spdgroup=   35
ELSEIF (SPD < 42.5)
    spdgroup=   40
ELSEIF (SPD < 47.5)
    spdgroup=   45
ELSEIF (SPD < 52.5)
    spdgroup=   50
ELSEIF (SPD < 57.5)
    spdgroup=   55
ELSEIF (SPD < 62.5)
    spdgroup=   60
ELSE
    spdgroup=   65
ENDIF



 WSPD01=vmt01*spd01  WSPD09=vmt09*spd09  WSPD17=vmt17*spd17
 WSPD02=vmt02*spd02  WSPD10=vmt10*spd10  WSPD18=vmt18*spd18
 WSPD03=vmt03*spd03  WSPD11=vmt11*spd11  WSPD19=vmt19*spd19
 WSPD04=vmt04*spd04  WSPD12=vmt12*spd12  WSPD20=vmt20*spd20
 WSPD05=vmt05*spd05  WSPD13=vmt13*spd13  WSPD21=vmt21*spd21
 WSPD06=vmt06*spd06  WSPD14=vmt14*spd14  WSPD22=vmt22*spd22
 WSPD07=vmt07*spd07  WSPD15=vmt15*spd15  WSPD23=vmt23*spd23
 WSPD08=vmt08*spd08  WSPD16=vmt16*spd16  WSPD24=vmt24*spd24



;--------------------------------------------------------------------
; Hourly Crosstab summaries start here.                             -
; Each Hourly Crosstab is a summary of:                             -
; VMT, Spd*VMT, HC(gms), CO(gms), Nx(gms), Wgtd Speed(mph),         -
; HC(tns), CO(tns), Nx(tns)                                         -
;--------------------------------------------------------------------
;
 CROSSTAB VAR=vmt01,wspd01,HEM01,CEM01,NEM01,form=12cs,
          ROW=jur,RANGE=0-30-1,,0-30,
          COL=FTYPE, RANGE=0-6-1,0-6,
          COMP=wSPD01/vmt01, form=9.3cs,
          COMP=HEM01/907184.74, form=9.3cs,;HC Emissions (in TONs)
          COMP=CEM01/907184.74, form=9.3cs,;CO Emissions (in TONs)
          COMP=NEM01/907184.74, form=9.3cs ;Nx Emissions (in TONs)

 CROSSTAB VAR=vmt02,wspd02,HEM02,CEM02,NEM02,form=12cs,
          ROW=jur,RANGE=0-30-1,,0-30,
          COL=FTYPE, RANGE=0-6-1,0-6,
          COMP=wSPD02/vmt02, form=9.3cs,
          COMP=HEM02/907184.74, form=9.3cs,;HC Emissions (in TONs)
          COMP=CEM02/907184.74, form=9.3cs,;CO Emissions (in TONs)
          COMP=NEM02/907184.74, form=9.3cs ;Nx Emissions (in TONs)


 CROSSTAB VAR=vmt03,wspd03,HEM03,CEM03,NEM03,form=12cs,
          ROW=jur,RANGE=0-30-1,,0-30,
          COL=FTYPE, RANGE=0-6-1,0-6,
          COMP=wSPD03/vmt03, form=9.3cs,
          COMP=HEM03/907184.74, form=9.3cs,;HC Emissions (in TONs)
          COMP=CEM03/907184.74, form=9.3cs,;CO Emissions (in TONs)
          COMP=NEM03/907184.74, form=9.3cs ;Nx Emissions (in TONs)


 CROSSTAB VAR=vmt04,wspd04,HEM04,CEM04,NEM04,form=12cs,
          ROW=jur,RANGE=0-30-1,,0-30,
          COL=FTYPE, RANGE=0-6-1,0-6,
          COMP=wSPD04/vmt04, form=9.3cs,
          COMP=HEM04/907184.74, form=9.3cs,;HC Emissions (in TONs)
          COMP=CEM04/907184.74, form=9.3cs,;CO Emissions (in TONs)
          COMP=NEM04/907184.74, form=9.3cs ;Nx Emissions (in TONs)


 CROSSTAB VAR=vmt05,wspd05,HEM05,CEM05,NEM05,form=12cs,
          ROW=jur,RANGE=0-30-1,,0-30,
          COL=FTYPE, RANGE=0-6-1,0-6,
          COMP=wSPD05/vmt05, form=9.3cs,
          COMP=HEM05/907184.74, form=9.3cs,;HC Emissions (in TONs)
          COMP=CEM05/907184.74, form=9.3cs,;CO Emissions (in TONs)
          COMP=NEM05/907184.74, form=9.3cs ;Nx Emissions (in TONs)


 CROSSTAB VAR=vmt06,wspd06,HEM06,CEM06,NEM06,form=12cs,
          ROW=jur,RANGE=0-30-1,,0-30,
          COL=FTYPE, RANGE=0-6-1,0-6,
          COMP=wSPD06/vmt06, form=9.3cs,
          COMP=HEM06/907184.74, form=9.3cs,;HC Emissions (in TONs)
          COMP=CEM06/907184.74, form=9.3cs,;CO Emissions (in TONs)
          COMP=NEM06/907184.74, form=9.3cs ;Nx Emissions (in TONs)


 CROSSTAB VAR=vmt07,wspd07,HEM07,CEM07,NEM07,form=12cs,
          ROW=jur,RANGE=0-30-1,,0-30,
          COL=FTYPE, RANGE=0-6-1,0-6,
          COMP=wSPD07/vmt07, form=9.3cs,
          COMP=HEM07/907184.74, form=9.3cs,;HC Emissions (in TONs)
          COMP=CEM07/907184.74, form=9.3cs,;CO Emissions (in TONs)
          COMP=NEM07/907184.74, form=9.3cs ;Nx Emissions (in TONs)


 CROSSTAB VAR=vmt08,wspd08,HEM08,CEM08,NEM08,form=12cs,
          ROW=jur,RANGE=0-30-1,,0-30,
```

```
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD08/vmt08, form=9.3cs,
         COMP=HEM08/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM08/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM08/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt09,wspd09,HEM09,CEM09,NEM09,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD09/vmt09, form=9.3cs,
         COMP=HEM09/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM09/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM09/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt10,wspd10,HEM10,CEM10,NEM10,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD10/vmt10, form=9.3cs,
         COMP=HEM10/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM10/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM10/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt11,wspd11,HEM11,CEM11,NEM11,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD11/vmt11, form=9.3cs,
         COMP=HEM11/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM11/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM11/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt12,wspd12,HEM12,CEM12,NEM12,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD12/vmt12, form=9.3cs,
         COMP=HEM12/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM12/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM12/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt13,wspd13,HEM13,CEM13,NEM13,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD13/vmt13, form=9.3cs,
         COMP=HEM13/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM13/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM13/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt14,wspd14,HEM14,CEM14,NEM14,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD14/vmt14, form=9.3cs,
         COMP=HEM14/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM14/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM14/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt15,wspd15,HEM15,CEM15,NEM15,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD15/vmt15, form=9.3cs,
         COMP=HEM15/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM15/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM15/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt16,wspd16,HEM16,CEM16,NEM16,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD16/vmt16, form=9.3cs,
         COMP=HEM16/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM16/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM16/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt17,wspd17,HEM17,CEM17,NEM17,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD17/vmt17, form=9.3cs,
         COMP=HEM17/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM17/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM17/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt18,wspd18,HEM18,CEM18,NEM18,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD18/vmt18, form=9.3cs,
         COMP=HEM18/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM18/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM18/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt19,wspd19,HEM19,CEM19,NEM19,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD19/vmt19, form=9.3cs,
         COMP=HEM19/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM19/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM19/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt20,wspd20,HEM20,CEM20,NEM20,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD20/vmt20, form=9.3cs,
         COMP=HEM20/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM20/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM20/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt21,wspd21,HEM21,CEM21,NEM21,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD21/vmt21, form=9.3cs,
         COMP=HEM21/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM21/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM21/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt22,wspd22,HEM22,CEM22,NEM22,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
         COMP=wSPD22/vmt22, form=9.3cs,
         COMP=HEM22/907184.74, form=9.3cs,;HC Emissions (in TONs)
         COMP=CEM22/907184.74, form=9.3cs,;CO Emissions (in TONs)
         COMP=NEM22/907184.74, form=9.3cs ;Nx Emissions (in TONs)


CROSSTAB VAR=vmt23,wspd23,HEM23,CEM23,NEM23,form=12cs,
         ROW=jur,RANGE=0-30-1,,0-30,
         COL=FTYPE, RANGE=0-6-1,0-6,
```

```
          COMP=wSPD23/vmt23, form=9.3cs,
          COMP=HEM23/907184.74, form=9.3cs,;HC Emissions (in TONs)
          COMP=CEM23/907184.74, form=9.3cs,;CO Emissions (in TONs)
          COMP=NEM23/907184.74, form=9.3cs ;Nx Emissions (in TONs)


 CROSSTAB VAR=vmt24,wspd24,HEM24,CEM24,NEM24,form=12cs,
          ROW=jur,RANGE=0-30-1,,0-30,
          COL=FTYPE, RANGE=0-6-1,0-6,
          COMP=wSPD24/vmt24, form=9.3cs,
          COMP=HEM24/907184.74, form=9.3cs,;HC Emissions (in TONs)
          COMP=CEM24/907184.74, form=9.3cs,;CO Emissions (in TONs)
          COMP=NEM24/907184.74, form=9.3cs ;Nx Emissions (in TONs)


;---------------------------------------------------------------------
; Total Regional Crosstab summary is below:  Juris x FTpe           -
; Reflects all 24 hours.                                            -
; VMT, Spd*VMT, HC(gms), CO(gms), Nx(gms), Wgtd Speed(mph),         -
; HC(tns), CO(tns), Nx(tns)                                         -
;---------------------------------------------------------------------
; TOTAL VMT, HC, CO, Nox Emissions
 CROSSTAB VAR=_cnt, vmt, hem, cem, nem,wspd,form=12cs,
          ROW=jur,RANGE=0-30-1,,0-30,
          COL=FTYPE, RANGE=0-6-1,0-6,
          COMP=wspd/vmt, form=12.1cs,  ;WGTD SPEED
          COMP=HEM /907184.74, form=9.3cs,;HC Emissions (in TONs)
          COMP=CEM /907184.74, form=9.3cs,;CO Emissions (in TONs)
          COMP=NEM /907184.74, form=9.3cs ;Nx Emissions (in TONs)
;---------------------------------------------------------------------
; Total Regional Crosstab summary is below:    MSA x Ftype          -
; Reflects all 24 hours.                                            -
; VMT, Spd*VMT, HC(gms), CO(gms), Nx(gms), Wgtd Speed(mph),         -
; HC(tns), CO(tns), Nx(tns)                                         -
;---------------------------------------------------------------------
; TOTAL VMT, HC, CO, Nox Emissions
 CROSSTAB VAR=_cnt, vmt, hem, cem, nem,wspd,form=12cs,
          ROW=msa,RANGE=0-30-1,,0-30,
          COL=FTYPE, RANGE=0-6-1,0-6,
          COMP=wspd/vmt, form=12.1cs,  ;WGTD SPEED
          COMP=HEM/907184.74, form=9.3cs,;HC Emissions (in TONs)
          COMP=CEM/907184.74, form=9.3cs,;CO Emissions (in TONs)
          COMP=NEM/907184.74, form=9.3cs ;Nx Emissions (in TONs)

 CROSSTAB VAR=_cnt,vmt,hem,cem,nem,form=12cs,
          ROW=spdgroup,RANGE=0-65-1,,0-65,
          COL=msa, RANGE=0-1-1,0-1,
          COMP=HEM/907184.74, form=9.3cs, ;HC Emissions in tons
          COMP=CEM/907184.74, form=9.3cs, ;CO Emissions in tons
          COMP=NEM/907184.74, form=9.3cs, ;Nx Emissions in tons
          COMP=HEM/vmt, form=9.3cs,    ;HC Emissions composite rate
          COMP=CEM/vmt, form=9.3cs,    ;CO Emissions composite rate
          COMP=NEM/vmt, form=9.3cs     ;Nx Emissions composite rate
endrun

*copy tppl*.prn running.rpt

*del tppl*.prn
```

STRT_SKR.S

```
*del tppl*.PRN
;-------------------------------------------------------------------------
;STRT_SKR.S - Calculation of Zonal Startup / SOAK emissions
;Note 4/15/03 update -"smoothed" hot/cold start distribution at hour 4
;     4/29/03 corrected starting emission equation error
;
;-------------------------------------------------------------------------
;INPUT Files:
 JRTRPDST='trip.dst'       ; veh trip juris level distr
 HRLY_Os ='HRLYORIG.ASC'   ; Hourly Zonal trip origins
 HRLY_Ds ='HRLYDEST.ASC'   ; Hourly Zonal trip destinations

;-----------------------------------------------------------
; Emission juris codes
;-----------------------------------------------------------
;  1) DC                      15) PW
;  2) Mtg                     16) Sta
;  3) PG                      17) Alx
;  4) How                     18) StM
;  5) Aar                     19) WashCoExtls   2164-2168
;  6) Car                     20) Clk
;  7) Balt Extls 2172-2191    21) Fau
;  8) Cal                     22) Jef
;  9) Chs                     23) VA W.Extls    2154-2163
; 10) Frd                     24) Spts
; 11) Frd. Extls  2169-2171   25) KG
; 12) Arl                     26) Fbg
; 13) FFx                     27) VA S.Extls    2145-2153
; 14) Ldn
;-----------------------------------------------------------
 j01S_RAT='%_m6Sub_%%_m6Pre_%dc.stt'; DC      Strt HC,CO,Nx Rates
 j02S_RAT='%_m6Sub_%%_m6Pre_%mc.stt'; Mtg     Strt HC,CO,Nx Rates
 j03S_RAT='%_m6Sub_%%_m6Pre_%pg.stt'; PG      Strt HC,CO,Nx Rates
 j04S_RAT='%_m6Sub_%%_m6Pre_%pg.stt'; "       Strt HC,CO,Nx Rates
 j05S_RAT='%_m6Sub_%%_m6Pre_%pg.stt'; "       Strt HC,CO,Nx Rates
 j06S_RAT='%_m6Sub_%%_m6Pre_%pg.stt'; "       Strt HC,CO,Nx Rates
 j07S_RAT='%_m6Sub_%%_m6Pre_%pg.stt'; "       Strt HC,CO,Nx Rates
 j08S_RAT='%_m6Sub_%%_m6Pre_%ca.stt'; Cal     Strt HC,CO,Nx Rates
 j09S_RAT='%_m6Sub_%%_m6Pre_%ch.stt'; Chs     Strt HC,CO,Nx Rates
 j10S_RAT='%_m6Sub_%%_m6Pre_%fr.stt'; Frd     Strt HC,CO,Nx Rates
 j11S_RAT='%_m6Sub_%%_m6Pre_%fr.stt'; "       Strt HC,CO,Nx Rates
 j12S_RAT='%_m6Sub_%%_m6Pre_%ar.stt'; Arl     Strt HC,CO,Nx Rates
 j13S_RAT='%_m6Sub_%%_m6Pre_%fx.stt'; Ffx     Strt HC,CO,Nx Rates
 j14S_RAT='%_m6Sub_%%_m6Pre_%ld.stt'; Ldn     Strt HC,CO,Nx Rates
 j15S_RAT='%_m6Sub_%%_m6Pre_%pw.stt'; PW      Strt HC,CO,Nx Rates
 j16S_RAT='%_m6Sub_%%_m6Pre_%st.stt'; Sta     Strt HC,CO,Nx Rates
 j17S_RAT='%_m6Sub_%%_m6Pre_%al.stt'; Alx     Strt HC,CO,Nx Rates
 j18S_RAT='%_m6Sub_%%_m6Pre_%sm.stt'; StM     Strt HC,CO,Nx Rates
 j19S_RAT='%_m6Sub_%%_m6Pre_%we.stt'; Wash    Strt HC,CO,Nx Rates
 j20S_RAT='%_m6Sub_%%_m6Pre_%cl.stt'; Clk     Strt HC,CO,Nx Rates
 j21S_RAT='%_m6Sub_%%_m6Pre_%cl.stt'; "       Strt HC,CO,Nx Rates
 j22S_RAT='%_m6Sub_%%_m6Pre_%cl.stt'; "       Strt HC,CO,Nx Rates
 j23S_RAT='%_m6Sub_%%_m6Pre_%cl.stt'; "       Strt HC,CO,Nx Rates
 j24S_RAT='%_m6Sub_%%_m6Pre_%sp.stt'; Spts    Strt HC,CO,Nx Rates
 j25S_RAT='%_m6Sub_%%_m6Pre_%sp.stt'; "       Strt HC,CO,Nx Rates
 j26S_RAT='%_m6Sub_%%_m6Pre_%sp.stt'; "       Strt HC,CO,Nx Rates
 j27S_RAT='%_m6Sub_%%_m6Pre_%sp.stt'; "       Strt HC,CO,Nx Rates
;-------------------------------------------------
;-----------------------------------------------------------
 J01SKRAT='%_m6Sub_%%_m6Pre_%dc.ram'; DC      Strt HC,CO,Nx Rates
 j02SKRAT='%_m6Sub_%%_m6Pre_%mc.ram'; Mtg     Strt HC,CO,Nx Rates
 j03SKRAT='%_m6Sub_%%_m6Pre_%pg.ram'; PG      Strt HC,CO,Nx Rates
 j04SKRAT='%_m6Sub_%%_m6Pre_%pg.ram'; "       Strt HC,CO,Nx Rates
 j05SKRAT='%_m6Sub_%%_m6Pre_%pg.ram'; "       Strt HC,CO,Nx Rates
 j06SKRAT='%_m6Sub_%%_m6Pre_%pg.ram'; "       Strt HC,CO,Nx Rates
 j07SKRAT='%_m6Sub_%%_m6Pre_%pg.ram'; "       Strt HC,CO,Nx Rates
 j08SKRAT='%_m6Sub_%%_m6Pre_%ca.ram'; Cal     Strt HC,CO,Nx Rates
 j09SKRAT='%_m6Sub_%%_m6Pre_%ch.ram'; Chs     Strt HC,CO,Nx Rates
 j10SKRAT='%_m6Sub_%%_m6Pre_%fr.ram'; Frd     Strt HC,CO,Nx Rates
 j11SKRAT='%_m6Sub_%%_m6Pre_%fr.ram'; "       Strt HC,CO,Nx Rates
 j12SKRAT='%_m6Sub_%%_m6Pre_%ar.ram'; Arl     Strt HC,CO,Nx Rates
 j13SKRAT='%_m6Sub_%%_m6Pre_%fx.ram'; Ffx     Strt HC,CO,Nx Rates
 j14SKRAT='%_m6Sub_%%_m6Pre_%ld.ram'; Ldn     Strt HC,CO,Nx Rates
 j15SKRAT='%_m6Sub_%%_m6Pre_%pw.ram'; PW      Strt HC,CO,Nx Rates
 j16SKRAT='%_m6Sub_%%_m6Pre_%st.ram'; Sta     Strt HC,CO,Nx Rates
 j17SKRAT='%_m6Sub_%%_m6Pre_%al.ram'; Alx     Strt HC,CO,Nx Rates
 j18SKRAT='%_m6Sub_%%_m6Pre_%sm.ram'; StM     Strt HC,CO,Nx Rates
 j19SKRAT='%_m6Sub_%%_m6Pre_%we.ram'; Wash    Strt HC,CO,Nx Rates
 j20SKRAT='%_m6Sub_%%_m6Pre_%cl.ram'; Clk     Strt HC,CO,Nx Rates
 j21SKRAT='%_m6Sub_%%_m6Pre_%cl.ram'; "       Strt HC,CO,Nx Rates
 j22SKRAT='%_m6Sub_%%_m6Pre_%cl.ram'; "       Strt HC,CO,Nx Rates
 j23SKRAT='%_m6Sub_%%_m6Pre_%cl.ram'; "       Strt HC,CO,Nx Rates
 j24SKRAT='%_m6Sub_%%_m6Pre_%sp.ram'; Spts    Strt HC,CO,Nx Rates
 j25SKRAT='%_m6Sub_%%_m6Pre_%sp.ram'; "       Strt HC,CO,Nx Rates
 j26SKRAT='%_m6Sub_%%_m6Pre_%sp.ram'; "       Strt HC,CO,Nx Rates
 j27SKRAT='%_m6Sub_%%_m6Pre_%sp.ram'; "       Strt HC,CO,Nx Rates
;-------------------------------------------------
run pgm=matrix
; -------------------------------------------------------------
zones=2191

ZDATI[1]=@HRLY_Os@,Z=1-5,
  HORIG01=  6- 15, HORIG02= 16-  25, HORIG03= 26- 35, HORIG04= 36- 45,
  HORIG05= 46- 55, HORIG06= 56-  65, HORIG07= 66- 75, HORIG08= 76- 85,
  HORIG09= 86- 95, HORIG10= 96- 105, HORIG11=106-115, HORIG12=116-125,
  HORIG13=126-135, HORIG14=136- 145, HORIG15=146-155, HORIG16=156-165,
  HORIG17=166-175, HORIG18=176- 185, HORIG19=186-195, HORIG20=196-205,
  HORIG21=206-215, HORIG22=216- 225, HORIG23=226-235, HORIG24=236-245

ZDATI[2]=@HRLY_Ds@,Z=1-5,
  HDEST01=  6- 15, HDEST02= 16-  25, HDEST03= 26- 35, HDEST04= 36- 45,
  HDEST05= 46- 55, HDEST06= 56-  65, HDEST07= 66- 75, HDEST08= 76- 85,
  HDEST09= 86- 95, HDEST10= 96- 105, HDEST11=106-115, HDEST12=116-125,
  HDEST13=126-135, HDEST14=136- 145, HDEST15=146-155, HDEST16=156-165,
  HDEST17=166-175, HDEST18=176- 185, HDEST19=186-195, HDEST20=196-205,
  HDEST21=206-215, HDEST22=216- 225, HDEST23=226-235, HDEST24=236-245

; Set up arrays

ARRAY,
STRT =24, ; hrly Origins
STOP =24, ; hrly Destinations

CHCR =24, ; Cold Start Composite HC Rate
CCCR =24, ; Cold Start Composite CO Rate
CNCR =24, ; Cold Start Composite Nx Rate

HHCR =24, ; Hot  Start Composite HC Rate
HCCR =24, ; Hot  Start Composite CO Rate
HNCR =24, ; Hot  Start Composite Nx Rate

HSKR =24, ; Hot  Soak  Composite    Rate

STHE =24, ; Start Composite HC EMISS
STCE =24, ; Start Composite CO EMISS
STNE =24, ; Start Composite Nx EMISS

HSKE =24, ; Start Composite Nx EMISS
```

```
J_HC =27, ; Juris. HC Total EMISSIONS  GMS
J_CO =27, ; Juris. CO Total EMISSIONS  GMS
J_NX =27, ; Juris. NX Total EMISSIONS  GMS
J_SK =27, ; Juris. Sk Total EMISSIONS  GMS

J_HCR=27, ; Juris. HC Total RATE Gm/Tr
J_COR=27, ; Juris. CO Total RATE Gm/Tr
J_NXR=27, ; Juris. NX Total RATE Gm/Tr
J_SKR=27, ; Juris. Sk Total RATE Gm/Tr

J_HCT=27, ; Juris. HC Total EMISSIONS  TONS
J_COT=27, ; Juris. CO Total EMISSIONS  TONS
J_NXT=27, ; Juris. NX Total EMISSIONS  TONS
J_SKT=27, ; Juris. NX Total EMISSIONS  TONS

J_ST =27, ; Juris. Trip Starts
J_SP =27  ; Juris. Trip Stops

;Establish hourly cold and hot start pcts
;- The proportion of Cold, Hot Vehicle starts for each hour (1-24)
;  Distributions taken from 1994 HTS - Hamid Humeida

;-------------------------------------------------------------------------
; NOTE: The hourly arrays are indexed from 1 to 24, as follows:
; 1-  12mid - 12:59AM       13- 12noon - 12:59PM
; 2- 1:00AM -  1:59AM       14- 1:00PM -  1:59PM
; 3- 2:00AM -  2:59AM       15- 2:00PM -  2:59PM
; 4- 3:00AM -  3:59AM       16- 3:00PM -  3:59PM
; 5- 4:00AM -  4:59AM       17- 4:00PM -  4:59PM
; 6- 5:00AM -  5:59AM       18- 5:00PM -  5:59PM
; 7- 6:00AM -  6:59AM       19- 6:00PM -  6:59PM
; 8- 7:00AM -  1:59AM       20- 7:00PM -  7:59PM
; 9- 8:00AM -  2:59AM       21- 8:00PM -  8:59PM
;10- 9:00AM -  3:59AM       22- 9:00PM -  9:59PM
;11-10:00AM -  4:59AM       23-10:00PM - 10:59PM
;12-11:00AM - 11:59AM       24-11:00PM - 11:59PM
;-------------------------------------------------------------------------
;
;

LOOKUP NAME=CHDST,             ;Cold / Hot Distribution for each hour
       LOOKUP[1]=1, RESULT=2,  ; Pct. Cold Starts
       LOOKUP[2]=1, RESULT=3,  ; Pct. Hot  Starts
       INTERPOLATE = N,LIST=T,

R= " 1,0.847,0.153 ",
   " 2,0.838,0.162 ",
   " 3,0.929,0.071 ",
   " 4,0.912,0.088 ",
   " 5,0.894,0.106 ",
   " 6,0.930,0.070 ",
   " 7,0.884,0.116 ",
   " 8,0.829,0.171 ",
   " 9,0.730,0.270 ",
   "10,0.615,0.385 ",
   "11,0.554,0.446 ",
   "12,0.551,0.449 ",
   "13,0.505,0.495 ",
   "14,0.512,0.488 ",
   "15,0.562,0.438 ",
   "16,0.583,0.417 ",
   "17,0.605,0.395 ",
   "18,0.599,0.401 ",
   "19,0.552,0.448 ",
   "20,0.570,0.430 ",
   "21,0.616,0.384 ",
   "22,0.664,0.336 ",
   "23,0.711,0.289 ",
   "24,0.730,0.270 "

;-------------------------------------------------------------------------
LOOKUP NAME=JCDTAZRG,          ; JCD AQ Juris code/taz range equiv
       LOOKUP[1]=1, RESULT=2,  ; Low  TAZ Range
       LOOKUP[2]=1, RESULT=3,  ; High TAZ Range
       LOOKUP[3]=1, RESULT=4,  ; JCD Code (1-27)
       INTERPOLATE = N,LIST=T,FAIL=0,0,0,
 R=    "  1,    1,  319,  1 ", ; DC
       "  2,  320,  627,  2 ", ; Mtg
       "  3,  640, 1020,  3 ", ; PG
       "  4, 1080, 1099,  4 ", ; How
       "  5, 1110, 1142,  5 ", ; Aar
       "  6, 1060, 1073,  6 ", ; Car
       "  7, 2172, 2191,  7 ", ; Balt Extls 2172-2191
       "  8, 1150, 1163,  8 ", ; Cal
       "  9, 1200, 1223,  9 ", ; Chs
       " 10, 1030, 1053, 10 ", ; Frd
       " 11, 2169, 2171, 11 ", ; Frd Extls 2169-2171
       " 12, 1230, 1311, 12 ", ; Arl
       " 13, 1400, 1755, 13 ", ; FFx
       " 14, 1780, 1905, 14 ", ; Ldn
       " 15, 1920, 2061, 15 ", ; PW
       " 16, 2080, 2093, 16 ", ; Sta
       " 17, 1330, 1389, 17 ", ; Alx
       " 18, 1170, 1190, 18 ", ; StM
       " 19, 2164, 2168, 19 ", ; NW. Extls  2164-2168
       " 20, 2130, 2132, 20 ", ; Clk
       " 21, 2115, 2125, 21 ", ; Fau
       " 22, 2135, 2141, 22 ", ; Jef
       " 23, 2154, 2163, 23 ", ; W.Extls    2154-2163
       " 24, 2105, 2110, 24 ", ; Spts
       " 25, 2070, 2074, 25 ", ; KG
       " 26, 2100, 2101, 26 ", ; Fbg
       " 27, 2145, 2153, 27 "  ; S.Extls    2145-2153
;
;-------------------------------------------------------------------------
; Jurisdictional Trip Distribution Array                                 -
;-------------------------------------------------------------------------
  LOOKUP NAME=JDST,                ;
         LOOKUP[01]=1, RESULT=2,  ;
         LOOKUP[02]=1, RESULT=3,  ;
         LOOKUP[03]=1, RESULT=4,  ;
         LOOKUP[04]=1, RESULT=5,  ;
         LOOKUP[05]=1, RESULT=6,  ;
         LOOKUP[06]=1, RESULT=7,  ;
         LOOKUP[07]=1, RESULT=8,  ;
         LOOKUP[08]=1, RESULT=9,  ;
         LOOKUP[09]=1, RESULT=10, ;
         LOOKUP[10]=1, RESULT=11, ;
         LOOKUP[11]=1, RESULT=12, ;
         LOOKUP[12]=1, RESULT=13, ;
         LOOKUP[13]=1, RESULT=14, ;
         LOOKUP[14]=1, RESULT=15, ;
         LOOKUP[15]=1, RESULT=16, ;
         LOOKUP[16]=1, RESULT=17, ;
         LOOKUP[17]=1, RESULT=18, ;
         LOOKUP[18]=1, RESULT=19, ;
         LOOKUP[19]=1, RESULT=20, ;
         LOOKUP[20]=1, RESULT=21, ;
         LOOKUP[21]=1, RESULT=22, ;
         LOOKUP[22]=1, RESULT=23, ;
         LOOKUP[23]=1, RESULT=24, ;
         LOOKUP[24]=1, RESULT=25, ;
         LOOKUP[25]=1, RESULT=26, ;
         LOOKUP[26]=1, RESULT=27, ;
         LOOKUP[27]=1, RESULT=28, ;
         INTERPOLATE = N,LIST=T,FAIL=0,0,0,File=@jrtrpdst@
```

```
;--------------------------------------------------------------------
; Hot & Cold start emission rate lookups by jurisdiction           -
;--------------------------------------------------------------------
;
;--------------------------------------------------------------------
  LOOKUP NAME=J01SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J01S_RAT@

  LOOKUP NAME=J02SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J02S_RAT@

  LOOKUP NAME=J03SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J03S_RAT@

  LOOKUP NAME=J04SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J04S_RAT@

  LOOKUP NAME=J05SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J05S_RAT@

  LOOKUP NAME=J06SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J06S_RAT@

  LOOKUP NAME=J07SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J07S_RAT@

  LOOKUP NAME=J08SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J08S_RAT@

  LOOKUP NAME=J09SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J09S_RAT@

  LOOKUP NAME=J10SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J10S_RAT@

  LOOKUP NAME=J11SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J11S_RAT@

  LOOKUP NAME=J12SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J12S_RAT@

  LOOKUP NAME=J13SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J13S_RAT@

  LOOKUP NAME=J14SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
         LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
         LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
         LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
         INTERPOLATE = N,LIST=T,file=@J14S_RAT@

  LOOKUP NAME=J15SR,              ;
         LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
         LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
         LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
```

```
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J15S_RAT@

LOOKUP NAME=J16SR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J16S_RAT@

LOOKUP NAME=J17SR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J17S_RAT@

LOOKUP NAME=J18SR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J18S_RAT@

LOOKUP NAME=J19SR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J19S_RAT@

LOOKUP NAME=J20SR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J20S_RAT@

LOOKUP NAME=J21SR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J21S_RAT@

LOOKUP NAME=J22SR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J22S_RAT@

LOOKUP NAME=J23SR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J23S_RAT@

LOOKUP NAME=J24SR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J24S_RAT@

LOOKUP NAME=J25SR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J25S_RAT@

LOOKUP NAME=J26SR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J26S_RAT@

LOOKUP NAME=J27SR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J27S_RAT@

;---------------------------------------------------------------------
; End of Hot Cold Emission Rate Lookups
;---------------------------------------------------------------------
;---------------------------------------------------------------------
; Hot Soak         emission rate lookups by jurisdiction          -
;---------------------------------------------------------------------
;
;---------------------------------------------------------------------
LOOKUP NAME=J01KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J01SKRAT@

LOOKUP NAME=J02KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
```

```
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J02SKRAT@

LOOKUP NAME=J03KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J03SKRAT@

LOOKUP NAME=J04KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J04SKRAT@

LOOKUP NAME=J05KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J05SKRAT@

LOOKUP NAME=J06KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J06SKRAT@

LOOKUP NAME=J07KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J07SKRAT@

LOOKUP NAME=J08KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J08SKRAT@

LOOKUP NAME=J09KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J09SKRAT@

LOOKUP NAME=J10KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J10SKRAT@

LOOKUP NAME=J11KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J11SKRAT@

LOOKUP NAME=J12KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J12SKRAT@

LOOKUP NAME=J13KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J13SKRAT@

LOOKUP NAME=J14KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J14SKRAT@

LOOKUP NAME=J15KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J15SKRAT@

LOOKUP NAME=J16KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J16SKRAT@

LOOKUP NAME=J17KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
```

```
        INTERPOLATE = N,LIST=T,file=@J17SKRAT@

LOOKUP NAME=J18KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J18SKRAT@

LOOKUP NAME=J19KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J19SKRAT@

LOOKUP NAME=J20KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J20SKRAT@

LOOKUP NAME=J21KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J21SKRAT@

LOOKUP NAME=J22KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J22SKRAT@

LOOKUP NAME=J23KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J23SKRAT@

LOOKUP NAME=J24KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J24SKRAT@

LOOKUP NAME=J25KR,              ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J25SKRAT@

  LOOKUP NAME=J26KR,            ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J26SKRAT@

  LOOKUP NAME=J27KR,            ;
        LOOKUP[1]=1, RESULT=2,  ; Hot  HC Start Rate (gm/trip)
        LOOKUP[2]=1, RESULT=3,  ; Hot  CO Start Rate (gm/trip)
        LOOKUP[3]=1, RESULT=4,  ; Hot  Nx Start Rate (gm/trip)
        LOOKUP[4]=1, RESULT=5,  ; Cold HC Start Rate (gm/trip)
        LOOKUP[5]=1, RESULT=6,  ; Cold CO Start Rate (gm/trip)
        LOOKUP[6]=1, RESULT=7,  ; Cold Nx Start Rate (gm/trip)
        INTERPOLATE = N,LIST=T,file=@J27SKRAT@

;---------------------------------------------------------------------
; End of Hot sOAK Emission Rate Lookups
;---------------------------------------------------------------------
;---------------------------------------------------------------------
; Now put hourly zonal trip starts, STOPS into HOURLY arrayS                      -
;---------------------------------------------------------------------

 STRT[01]=HORIG01 STRT[07]=HORIG07 STRT[13]=HORIG13 STRT[19]=HORIG19
 STRT[02]=HORIG02 STRT[08]=HORIG08 STRT[14]=HORIG14 STRT[20]=HORIG20
 STRT[03]=HORIG03 STRT[09]=HORIG09 STRT[15]=HORIG15 STRT[21]=HORIG21
 STRT[04]=HORIG04 STRT[10]=HORIG10 STRT[16]=HORIG16 STRT[22]=HORIG22
 STRT[05]=HORIG05 STRT[11]=HORIG11 STRT[17]=HORIG17 STRT[23]=HORIG23
 STRT[06]=HORIG06 STRT[12]=HORIG12 STRT[18]=HORIG18 STRT[24]=HORIG24

 STOP[01]=HDEST01 STOP[07]=HDEST07 STOP[13]=HDEST13 STOP[19]=HDEST19
 STOP[02]=HDEST02 STOP[08]=HDEST08 STOP[14]=HDEST14 STOP[20]=HDEST20
 STOP[03]=HDEST03 STOP[09]=HDEST09 STOP[15]=HDEST15 STOP[21]=HDEST21
 STOP[04]=HDEST04 STOP[10]=HDEST10 STOP[16]=HDEST16 STOP[22]=HDEST22
 STOP[05]=HDEST05 STOP[11]=HDEST11 STOP[17]=HDEST17 STOP[23]=HDEST23
 STOP[06]=HDEST06 STOP[12]=HDEST12 STOP[18]=HDEST18 STOP[24]=HDEST24

;---------------------------------------------------------------------
;Next, define air quality jurisdiction codes (jcd).                    -
; associated with each origin TAZ.                                     -
; These include some external station ranges as well as jurisdictions  -       -
;---------------------------------------------------------------------
     JCD = 0  ; initialize JCD Code
LOOP IDX = 1,27
     IF ( (I >= JCDTAZRG(1,IDX)) && (I <= JCDTAZRG(2,IDX)) )
         JCD = JCDTAZRG(3,IDX)
     ENDIF
ENDLOOP
;---------------------------------------------------------------------
; end of JCD definitions
;---------------------------------------------------------------------
;---------------------------------------------------------------------
; Now compute hourly
; Composite Hot  HC,CO,NOx emission rates (HHCR,HCCR,HNCR) &
; Composite Cold HC,CO,NOx emission rates (CHCR,CCCR,CNCR)
;---------------------------------------------------------------------
;

   LOOP IDX= 1,24
             ; 35 is dummy lookup value to read 1-line lookup tabs
```

```
; ---- Begin Hot Start composite HC, CO, Nx Rate computations ----------

       HHCR[IDX] = (J01SR(1,35) * JDST(JCD,1))  +   ; Hot Start
                   (J02SR(1,35) * JDST(JCD,2))  +   ; HC Composite
                   (J03SR(1,35) * JDST(JCD,3))  +   ; RATE
                   (J04SR(1,35) * JDST(JCD,4))  +   ;
                   (J05SR(1,35) * JDST(JCD,5))  +   ;
                   (J06SR(1,35) * JDST(JCD,6))  +   ;
                   (J07SR(1,35) * JDST(JCD,7))  +   ;
                   (J08SR(1,35) * JDST(JCD,8))  +   ;
                   (J09SR(1,35) * JDST(JCD,9))  +   ;
                   (J10SR(1,35) * JDST(JCD,10)) +   ;
                   (J11SR(1,35) * JDST(JCD,11)) +   ;
                   (J12SR(1,35) * JDST(JCD,12)) +   ;
                   (J13SR(1,35) * JDST(JCD,13)) +   ;
                   (J14SR(1,35) * JDST(JCD,14)) +   ;
                   (J15SR(1,35) * JDST(JCD,15)) +   ;
                   (J16SR(1,35) * JDST(JCD,16)) +   ;
                   (J17SR(1,35) * JDST(JCD,17)) +   ;
                   (J18SR(1,35) * JDST(JCD,18)) +   ;
                   (J19SR(1,35) * JDST(JCD,19)) +   ;
                   (J20SR(1,35) * JDST(JCD,20)) +   ;
                   (J21SR(1,35) * JDST(JCD,21)) +   ;
                   (J22SR(1,35) * JDST(JCD,22)) +   ;
                   (J23SR(1,35) * JDST(JCD,23)) +   ;
                   (J24SR(1,35) * JDST(JCD,24)) +   ;
                   (J25SR(1,35) * JDST(JCD,25)) +   ;
                   (J26SR(1,35) * JDST(JCD,26)) +   ;
                   (J27SR(1,35) * JDST(JCD,27))     ;

       HCCR[IDX] = (J01SR(2,35) * JDST(JCD,1))  +   ; Hot Start
                   (J02SR(2,35) * JDST(JCD,2))  +   ; CO Composite
                   (J03SR(2,35) * JDST(JCD,3))  +   ; RATE
                   (J04SR(2,35) * JDST(JCD,4))  +   ;
                   (J05SR(2,35) * JDST(JCD,5))  +   ;
                   (J06SR(2,35) * JDST(JCD,6))  +   ;
                   (J07SR(2,35) * JDST(JCD,7))  +   ;
                   (J08SR(2,35) * JDST(JCD,8))  +   ;
                   (J09SR(2,35) * JDST(JCD,9))  +   ;
                   (J10SR(2,35) * JDST(JCD,10)) +   ;
                   (J11SR(2,35) * JDST(JCD,11)) +   ;
                   (J12SR(2,35) * JDST(JCD,12)) +   ;
                   (J13SR(2,35) * JDST(JCD,13)) +   ;
                   (J14SR(2,35) * JDST(JCD,14)) +   ;
                   (J15SR(2,35) * JDST(JCD,15)) +   ;
                   (J16SR(2,35) * JDST(JCD,16)) +   ;
                   (J17SR(2,35) * JDST(JCD,17)) +   ;
                   (J18SR(2,35) * JDST(JCD,18)) +   ;
                   (J19SR(2,35) * JDST(JCD,19)) +   ;
                   (J20SR(2,35) * JDST(JCD,20)) +   ;
                   (J21SR(2,35) * JDST(JCD,21)) +   ;
                   (J22SR(2,35) * JDST(JCD,22)) +   ;
                   (J23SR(2,35) * JDST(JCD,23)) +   ;
                   (J24SR(2,35) * JDST(JCD,24)) +   ;
                   (J25SR(2,35) * JDST(JCD,25)) +   ;
                   (J26SR(2,35) * JDST(JCD,26)) +   ;
                   (J27SR(2,35) * JDST(JCD,27))     ;


       HNCR[IDX] = (J01SR(3,35) * JDST(JCD,1))  +   ; Hot Start
                   (J02SR(3,35) * JDST(JCD,2))  +   ; Nx Composite
                   (J03SR(3,35) * JDST(JCD,3))  +   ; RATE
                   (J04SR(3,35) * JDST(JCD,4))  +   ;
                   (J05SR(3,35) * JDST(JCD,5))  +   ;
                   (J06SR(3,35) * JDST(JCD,6))  +   ;
                   (J07SR(3,35) * JDST(JCD,7))  +   ;
                   (J08SR(3,35) * JDST(JCD,8))  +   ;
                   (J09SR(3,35) * JDST(JCD,9))  +   ;
                   (J10SR(3,35) * JDST(JCD,10)) +   ;
                   (J11SR(3,35) * JDST(JCD,11)) +   ;
                   (J12SR(3,35) * JDST(JCD,12)) +   ;
                   (J13SR(3,35) * JDST(JCD,13)) +   ;
                   (J14SR(3,35) * JDST(JCD,14)) +   ;
                   (J15SR(3,35) * JDST(JCD,15)) +   ;
                   (J16SR(3,35) * JDST(JCD,16)) +   ;
                   (J17SR(3,35) * JDST(JCD,17)) +   ;
                   (J18SR(3,35) * JDST(JCD,18)) +   ;
                   (J19SR(3,35) * JDST(JCD,19)) +   ;
                   (J20SR(3,35) * JDST(JCD,20)) +   ;
                   (J21SR(3,35) * JDST(JCD,21)) +   ;
                   (J22SR(3,35) * JDST(JCD,22)) +   ;
                   (J23SR(3,35) * JDST(JCD,23)) +   ;
                   (J24SR(3,35) * JDST(JCD,24)) +   ;
                   (J25SR(3,35) * JDST(JCD,25)) +   ;
                   (J26SR(3,35) * JDST(JCD,26)) +   ;
                   (J27SR(3,35) * JDST(JCD,27))     ;

; -------- End Hot Start Rate Computations --------------------------
; ---Begin Cold Start Composite HC, CO, Nx Rate computations ----------

       CHCR[IDX] = (J01SR(4,35) * JDST(JCD,1))  +   ; Hot Start
                   (J02SR(4,35) * JDST(JCD,2))  +   ; HC Composite
                   (J03SR(4,35) * JDST(JCD,3))  +   ; RATE
                   (J04SR(4,35) * JDST(JCD,4))  +   ;
                   (J05SR(4,35) * JDST(JCD,5))  +   ;
                   (J06SR(4,35) * JDST(JCD,6))  +   ;
                   (J07SR(4,35) * JDST(JCD,7))  +   ;
                   (J08SR(4,35) * JDST(JCD,8))  +   ;
                   (J09SR(4,35) * JDST(JCD,9))  +   ;
                   (J10SR(4,35) * JDST(JCD,10)) +   ;
                   (J11SR(4,35) * JDST(JCD,11)) +   ;
                   (J12SR(4,35) * JDST(JCD,12)) +   ;
                   (J13SR(4,35) * JDST(JCD,13)) +   ;
                   (J14SR(4,35) * JDST(JCD,14)) +   ;
                   (J15SR(4,35) * JDST(JCD,15)) +   ;
                   (J16SR(4,35) * JDST(JCD,16)) +   ;
                   (J17SR(4,35) * JDST(JCD,17)) +   ;
                   (J18SR(4,35) * JDST(JCD,18)) +   ;
                   (J19SR(4,35) * JDST(JCD,19)) +   ;
                   (J20SR(4,35) * JDST(JCD,20)) +   ;
                   (J21SR(4,35) * JDST(JCD,21)) +   ;
                   (J22SR(4,35) * JDST(JCD,22)) +   ;
                   (J23SR(4,35) * JDST(JCD,23)) +   ;
                   (J24SR(4,35) * JDST(JCD,24)) +   ;
                   (J25SR(4,35) * JDST(JCD,25)) +   ;
                   (J26SR(4,35) * JDST(JCD,26)) +   ;
                   (J27SR(4,35) * JDST(JCD,27))     ;

       CCCR[IDX] = (J01SR(5,35) * JDST(JCD,1))  +   ; Hot Start
                   (J02SR(5,35) * JDST(JCD,2))  +   ; CO Composite
                   (J03SR(5,35) * JDST(JCD,3))  +   ; RATE
                   (J04SR(5,35) * JDST(JCD,4))  +   ;
                   (J05SR(5,35) * JDST(JCD,5))  +   ;
                   (J06SR(5,35) * JDST(JCD,6))  +   ;
                   (J07SR(5,35) * JDST(JCD,7))  +   ;
                   (J08SR(5,35) * JDST(JCD,8))  +   ;
                   (J09SR(5,35) * JDST(JCD,9))  +   ;
                   (J10SR(5,35) * JDST(JCD,10)) +   ;
                   (J11SR(5,35) * JDST(JCD,11)) +   ;
                   (J12SR(5,35) * JDST(JCD,12)) +   ;
                   (J13SR(5,35) * JDST(JCD,13)) +   ;
                   (J14SR(5,35) * JDST(JCD,14)) +   ;
                   (J15SR(5,35) * JDST(JCD,15)) +   ;
                   (J16SR(5,35) * JDST(JCD,16)) +   ;
                   (J17SR(5,35) * JDST(JCD,17)) +   ;
```

```
                (J18SR(5,35) * JDST(JCD,18)) +   ;
                (J19SR(5,35) * JDST(JCD,19)) +   ;
                (J20SR(5,35) * JDST(JCD,20)) +   ;
                (J21SR(5,35) * JDST(JCD,21)) +   ;
                (J22SR(5,35) * JDST(JCD,22)) +   ;
                (J23SR(5,35) * JDST(JCD,23)) +   ;
                (J24SR(5,35) * JDST(JCD,24)) +   ;
                (J25SR(5,35) * JDST(JCD,25)) +   ;
                (J26SR(5,35) * JDST(JCD,26)) +   ;
                (J27SR(5,35) * JDST(JCD,27))     ;


      CNCR[IDX] = (J01SR(6,35) * JDST(JCD,1))  +   ; Hot Start
                  (J02SR(6,35) * JDST(JCD,2))  +   ; Nx Composite
                  (J03SR(6,35) * JDST(JCD,3))  +   ; RATE
                  (J04SR(6,35) * JDST(JCD,4))  +   ;
                  (J05SR(6,35) * JDST(JCD,5))  +   ;
                  (J06SR(6,35) * JDST(JCD,6))  +   ;
                  (J07SR(6,35) * JDST(JCD,7))  +   ;
                  (J08SR(6,35) * JDST(JCD,8))  +   ;
                  (J09SR(6,35) * JDST(JCD,9))  +   ;
                  (J10SR(6,35) * JDST(JCD,10)) +   ;
                  (J11SR(6,35) * JDST(JCD,11)) +   ;
                  (J12SR(6,35) * JDST(JCD,12)) +   ;
                  (J13SR(6,35) * JDST(JCD,13)) +   ;
                  (J14SR(6,35) * JDST(JCD,14)) +   ;
                  (J15SR(6,35) * JDST(JCD,15)) +   ;
                  (J16SR(6,35) * JDST(JCD,16)) +   ;
                  (J17SR(6,35) * JDST(JCD,17)) +   ;
                  (J18SR(6,35) * JDST(JCD,18)) +   ;
                  (J19SR(6,35) * JDST(JCD,19)) +   ;
                  (J20SR(6,35) * JDST(JCD,20)) +   ;
                  (J21SR(6,35) * JDST(JCD,21)) +   ;
                  (J22SR(6,35) * JDST(JCD,22)) +   ;
                  (J23SR(6,35) * JDST(JCD,23)) +   ;
                  (J24SR(6,35) * JDST(JCD,24)) +   ;
                  (J25SR(6,35) * JDST(JCD,25)) +   ;
                  (J26SR(6,35) * JDST(JCD,26)) +   ;
                  (J27SR(6,35) * JDST(JCD,27))     ;
; -------- End Cold Start Rate Computations ---------------------------

;--------------------------------------------------------------------
; Now compute hourly
; Composite Hot sOAK emission rate
;--------------------------------------------------------------------
      HSKR[IDX] = (J01KR(4,35) * JDST(JCD,1))  +   ; Hot Start
                  (J02KR(4,35) * JDST(JCD,2))  +   ; HC Composite
                  (J03KR(4,35) * JDST(JCD,3))  +   ; RATE
                  (J04KR(4,35) * JDST(JCD,4))  +   ;
                  (J05KR(4,35) * JDST(JCD,5))  +   ;
                  (J06KR(4,35) * JDST(JCD,6))  +   ;
                  (J07KR(4,35) * JDST(JCD,7))  +   ;
                  (J08KR(4,35) * JDST(JCD,8))  +   ;
                  (J09KR(4,35) * JDST(JCD,9))  +   ;
                  (J10KR(4,35) * JDST(JCD,10)) +   ;
                  (J11KR(4,35) * JDST(JCD,11)) +   ;
                  (J12KR(4,35) * JDST(JCD,12)) +   ;
                  (J13KR(4,35) * JDST(JCD,13)) +   ;
                  (J14KR(4,35) * JDST(JCD,14)) +   ;
                  (J15KR(4,35) * JDST(JCD,15)) +   ;
                  (J16KR(4,35) * JDST(JCD,16)) +   ;
                  (J17KR(4,35) * JDST(JCD,17)) +   ;
                  (J18KR(4,35) * JDST(JCD,18)) +   ;
                  (J19KR(4,35) * JDST(JCD,19)) +   ;
                  (J20KR(4,35) * JDST(JCD,20)) +   ;
                  (J21KR(4,35) * JDST(JCD,21)) +   ;
                  (J22KR(4,35) * JDST(JCD,22)) +   ;
                  (J23KR(4,35) * JDST(JCD,23)) +   ;
                  (J24KR(4,35) * JDST(JCD,24)) +   ;
                  (J25KR(4,35) * JDST(JCD,25)) +   ;
                  (J26KR(4,35) * JDST(JCD,26)) +   ;
                  (J27KR(4,35) * JDST(JCD,27))     ;

   ; -------- End HOT SOAK Rate Computations --------------------------
   ; -------- Now apply Composite Cold/Hot Rates to trip origins---------
   ; -- TO COMPUTE HOURLY HC, CO, AND Nx EMISSIONS ----------------------
   ;  w/ 4/29/03 correction

      STHE[idx] =  strt[idx] *
                ((CHCR[idx] * CHDST(1,idx)) + (HHCR[idx] * CHDST(2,idx)))

      STCE[idx] =  strt[idx] *
                ((CCCR[idx] * CHDST(1,idx)) + (HCCR[idx] * CHDST(2,idx)))

      STNE[idx] =  strt[idx] *
                ((CNCR[idx] * CHDST(1,idx)) + (HNCR[idx] * CHDST(2,idx)))

   ; -------- Now apply Composite sOAK Rate to trip DESTINATIONS -------
   ; -- TO COMPUTE HOURLY SOAK EMISSIONS ----------------------

      HSKE[idx] =  STOP[idx] * HSKR[IDX]

   ;--------------------------------------------------------------------
   ;------- accumulate JUrisdictional HC, CO, Nx emissions, starts------
   ;--------------------------------------------------------------------
   LOOP JDX=1,27
   IF (JCD=JDX) ;
       J_HC[jdx] = J_HC[jdx] + STHE[idx]
       J_CO[jdx] = J_CO[jdx] + STCE[idx]
       J_NX[jdx] = J_NX[jdx] + STNE[idx]
       J_sk[jdx] = J_sk[jdx] + HSKE[idx]
       J_ST[jdx] = J_ST[jdx] + STRT[idx]
       J_SP[jdx] = J_SP[jdx] + STOP[idx]
   ENDIF
   ENDLOOP
   ;--------------------------------------------------------------------
   ;---END of Jurisdictional HC, CO, Nx emissions, starts --------------
   ;--------------------------------------------------------------------

   ;--------------------------------------
   ;TOTAL Regional HC,CO,Nx, Trip Origins -
   ;--------------------------------------
   ;
      TOTHC_ST = TOTHC_ST + STHE[idx]
      TOTCO_ST = TOTCO_ST + STCE[idx]
      TOTNX_ST = TOTNX_ST + STNE[idx]
      TOTsk    = TOTsk    + hske[idx]
      TOTstart = TOTstart + strt[idx]
      TOTstOP  = TOTstOP  + stOP[idx]
   ENDLOOP

   ;<<<<<<--------------
   if (i=627,628)
    t1=jdst(jcd,1),
    t2=jdst(jcd,2)   t3=jdst(jcd,3)  t4=jdst(jcd,4)   t5=jdst(jcd,5),
    t6=jdst(jcd,6)   t7=jdst(jcd,7)  t8=jdst(jcd,8)   t9=jdst(jcd,9),
   t10=jdst(jcd,10) t11=jdst(jcd,11) t12=jdst(jcd,12) t13=jdst(jcd,13),
   t14=jdst(jcd,14) t15=jdst(jcd,15) t16=jdst(jcd,16) t17=jdst(jcd,17),
   t18=jdst(jcd,18) t19=jdst(jcd,19) t20=jdst(jcd,20) t21=jdst(jcd,21),
   t22=jdst(jcd,22) t23=jdst(jcd,23) t24=jdst(jcd,24) t25=jdst(jcd,25),
   t26=jdst(jcd,26) t27=jdst(jcd,27)

   list=i(5), t1(5.3),  t2(5.3),  t3(5.3),  t4(5.3),
              t5(5.3),  t6(5.3),  t7(5.3),  t8(5.3),
              t9(5.3), t10(5.3), t11(5.3), t12(5.3),
             t13(5.3), t14(5.3), t15(5.3), t16(5.3),
```

```
         t17(5.3), t18(5.3), t19(5.3), t20(5.3),
         t21(5.3), t22(5.3), t23(5.3), t24(5.3),
         t25(5.3), t26(5.3), t27(5.3),
                     file=dud.dat  ;
endif
;<<<<<<--------------


IF (I=ZONES)       ;  ALL DONE WITH ZONE LOOP
;
; GET AVG rateS BY JURIS.& REGIONAL
;
;JURIS rATES:
LOOP TDX=1,27
  IF (J_ST[TDX] = 0)
    J_HCR[TDX] =  0
    J_COR[TDX] =  0
    J_NXR[TDX] =  0
  ELSE
    J_HCR[TDX] = J_HC[TDX] / J_ST[TDX]
    J_COR[TDX] = J_CO[TDX] / J_ST[TDX]
    J_NXR[TDX] = J_NX[TDX] / J_ST[TDX]
  ENDIF

  IF (J_Sp[TDX] = 0)
    J_SKR[TDX] =  0
  ELSE
    J_skR[TDX] = J_sk[TDX] / J_Sp[TDX]
  ENDIF
;
; CALCULATE EMISSIONS IN TONS HERE
;
    J_HCT[TDX] = J_HC[TDX]/907184.74
    J_COT[TDX] = J_CO[TDX]/907184.74
    J_NXT[TDX] = J_NX[TDX]/907184.74
    J_skT[TDX] = J_sk[TDX]/907184.74
;
;
;
ENDLOOP

;TOTAL RATES:
  IF (TOTSTART  = 0)
    T_HCR = 0
    T_COR = 0
    T_NXR = 0
  ELSE
    T_HCR = TOTHC_ST / TOTSTART
    T_COR = TOTCO_ST / TOTSTART
    T_NXR = TOTNX_ST / TOTSTART
  ENDIF

  IF (TOTSTop   = 0)
    T_SKR= 0
  ELSE
    T_SKR= TOTsk    / TOTSTop
  ENDIF

; CALCULATE TOTAL TONS
    T_HCT = TOTHC_ST / 907184.74
    T_COT = TOTCO_ST / 907184.74
    T_NXT = TOTNX_ST / 907184.74
    T_SKT = TOTSK    / 907184.74
```

```
; Now List Out The Results
;
LIST=' '
LIST=' '
LIST=' Total Starts and HC, CO, Nox Emissions (gms) By Jurisdiction'
LIST='                                                          HC           CO          Nx '
LIST='Juris     HC(gms)      CO(gms)     Nx(gms)     Trip Starts    gm/trip      gm/trip     gm/trip'
LIST='------ ------------ ------------ ------------ ------------ ------------ ------------ ------------'
print, form=13.2,list ='  DC  ',J_HC[01] J_CO[01] J_NX[01] J_ST[01](13.0) J_HCR[01] j_COR[01] J_NXR[01]
print, form=13.2,list =' MTG  ',J_HC[02] J_CO[02] J_NX[02] J_ST[02](13.0) J_HCR[02] j_COR[02] J_NXR[02]
print, form=13.2,list ='  PG  ',J_HC[03] J_CO[03] J_NX[03] J_ST[03](13.0) J_HCR[03] j_COR[03] J_NXR[03]
print, form=13.2,list =' ARL  ',J_HC[12] J_CO[12] J_NX[12] J_ST[12](13.0) J_HCR[12] j_COR[12] J_NXR[12]
print, form=13.2,list =' ALX  ',J_HC[17] J_CO[17] J_NX[17] J_ST[17](13.0) J_HCR[17] j_COR[17] J_NXR[17]
print, form=13.2,list =' FFX  ',J_HC[13] J_CO[13] J_NX[13] J_ST[13](13.0) J_HCR[13] j_COR[13] J_NXR[13]
print, form=13.2,list =' LDN  ',J_HC[14] J_CO[14] J_NX[14] J_ST[14](13.0) J_HCR[14] j_COR[14] J_NXR[14]
print, form=13.2,list ='  PW  ',J_HC[15] J_CO[15] J_NX[15] J_ST[15](13.0) J_HCR[15] j_COR[15] J_NXR[15]
print, form=13.2,list =' FRD  ',J_HC[10] J_CO[10] J_NX[10] J_ST[10](13.0) J_HCR[10] j_COR[10] J_NXR[10]
print, form=13.2,list =' HOW  ',J_HC[04] J_CO[04] J_NX[04] J_ST[04](13.0) J_HCR[04] j_COR[04] J_NXR[04]
print, form=13.2,list =' AAR  ',J_HC[05] J_CO[05] J_NX[05] J_ST[05](13.0) J_HCR[05] j_COR[05] J_NXR[05]
print, form=13.2,list =' CHS  ',J_HC[09] J_CO[09] J_NX[09] J_ST[09](13.0) J_HCR[09] j_COR[09] J_NXR[09]
print, form=13.2,list =' CAR  ',J_HC[06] J_CO[06] J_NX[06] J_ST[06](13.0) J_HCR[06] j_COR[06] J_NXR[06]
print, form=13.2,list =' CAL  ',J_HC[08] J_CO[08] J_NX[08] J_ST[08](13.0) J_HCR[08] j_COR[08] J_NXR[08]
print, form=13.2,list =' STM  ',J_HC[18] J_CO[18] J_NX[18] J_ST[18](13.0) J_HCR[18] j_COR[18] J_NXR[18]
print, form=13.2,list ='  KG  ',J_HC[25] J_CO[25] J_NX[25] J_ST[25](13.0) J_HCR[25] j_COR[25] J_NXR[25]
print, form=13.2,list =' FBG  ',J_HC[26] J_CO[26] J_NX[26] J_ST[26](13.0) J_HCR[26] j_COR[26] J_NXR[26]
print, form=13.2,list =' STA  ',J_HC[16] J_CO[16] J_NX[16] J_ST[16](13.0) J_HCR[16] j_COR[16] J_NXR[16]
print, form=13.2,list =' SPT  ',J_HC[24] J_CO[24] J_NX[24] J_ST[24](13.0) J_HCR[24] j_COR[24] J_NXR[24]
print, form=13.2,list =' FAU  ',J_HC[21] J_CO[21] J_NX[21] J_ST[21](13.0) J_HCR[21] j_COR[21] J_NXR[21]
print, form=13.2,list =' CLK  ',J_HC[20] J_CO[20] J_NX[20] J_ST[20](13.0) J_HCR[20] j_COR[20] J_NXR[20]
print, form=13.2,list =' JEF  ',J_HC[22] J_CO[22] J_NX[22] J_ST[22](13.0) J_HCR[22] j_COR[22] J_NXR[22]
print, form=13.2,list ='BaltEx',J_HC[07] J_CO[07] J_NX[07] J_ST[07](13.0) J_HCR[07] j_COR[07] J_NXR[07]
print, form=13.2,list ='Frd Ex',J_HC[11] J_CO[11] J_NX[11] J_ST[11](13.0) J_HCR[11] j_COR[11] J_NXR[11]
print, form=13.2,list ='WashEx',J_HC[19] J_CO[19] J_NX[19] J_ST[19](13.0) J_HCR[19] j_COR[19] J_NXR[19]
print, form=13.2,list ='VaW Ex',J_HC[23] J_CO[23] J_NX[23] J_ST[23](13.0) J_HCR[23] j_COR[23] J_NXR[23]
print, form=13.2,list ='VaS Ex',J_HC[27] J_CO[27] J_NX[27] J_ST[27](13.0) J_HCR[27] j_COR[27] J_NXR[27]
list=' '
print, form=13.2,list =' TOTAL',TOTHC_ST,TOTCO_ST,TOTNX_ST,TOTSTART(13.0)  T_HCR   T_COR   T_NXR


LIST=' '
LIST=' '
LIST=' '
LIST=' Total Starts and HC, CO, Nox Emissions (tons) By Jurisdiction'
LIST='                                                          '
LIST='Juris     HC(tns)      CO(tns)     Nx(tns)     Trip Starts '
LIST='------ ------------ ------------ ------------ ------------ '
print, form=13.3,list ='  DC  ',J_HCT[01] J_COT[01] J_NXT[01] J_ST[01](13.0)
print, form=13.3,list =' MTG  ',J_HCT[02] J_COT[02] J_NXT[02] J_ST[02](13.0)
print, form=13.3,list ='  PG  ',J_HCT[03] J_COT[03] J_NXT[03] J_ST[03](13.0)
print, form=13.3,list =' ARL  ',J_HCT[12] J_COT[12] J_NXT[12] J_ST[12](13.0)
print, form=13.3,list =' ALX  ',J_HCT[17] J_COT[17] J_NXT[17] J_ST[17](13.0)
print, form=13.3,list =' FFX  ',J_HCT[13] J_COT[13] J_NXT[13] J_ST[13](13.0)
print, form=13.3,list =' LDN  ',J_HCT[14] J_COT[14] J_NXT[14] J_ST[14](13.0)
print, form=13.3,list ='  PW  ',J_HCT[15] J_COT[15] J_NXT[15] J_ST[15](13.0)
print, form=13.3,list =' FRD  ',J_HCT[10] J_COT[10] J_NXT[10] J_ST[10](13.0)
print, form=13.3,list =' HOW  ',J_HCT[04] J_COT[04] J_NXT[04] J_ST[04](13.0)
print, form=13.3,list =' AAR  ',J_HCT[05] J_COT[05] J_NXT[05] J_ST[05](13.0)
print, form=13.3,list =' CHS  ',J_HCT[09] J_COT[09] J_NXT[09] J_ST[09](13.0)
print, form=13.3,list =' CAR  ',J_HCT[06] J_COT[06] J_NXT[06] J_ST[06](13.0)
print, form=13.3,list =' CAL  ',J_HCT[08] J_COT[08] J_NXT[08] J_ST[08](13.0)
print, form=13.3,list =' STM  ',J_HCT[18] J_COT[18] J_NXT[18] J_ST[18](13.0)
print, form=13.3,list ='  KG  ',J_HCT[25] J_COT[25] J_NXT[25] J_ST[25](13.0)
print, form=13.3,list =' FBG  ',J_HCT[26] J_COT[26] J_NXT[26] J_ST[26](13.0)
print, form=13.3,list =' STA  ',J_HCT[16] J_COT[16] J_NXT[16] J_ST[16](13.0)
print, form=13.3,list =' SPT  ',J_HCT[24] J_COT[24] J_NXT[24] J_ST[24](13.0)
print, form=13.3,list =' FAU  ',J_HCT[21] J_COT[21] J_NXT[21] J_ST[21](13.0)
```

```
print, form=13.3,list =' CLK  ',J_HCT[20] J_COT[20] J_NXT[20] J_ST[20](13.0)
print, form=13.3,list =' JEF  ',J_HCT[22] J_COT[22] J_NXT[22] J_ST[22](13.0)
print, form=13.3,list ='BaltEx',J_HCT[07] J_COT[07] J_NXT[07] J_ST[07](13.0)
print, form=13.3,list ='Frd Ex',J_HCT[11] J_COT[11] J_NXT[11] J_ST[11](13.0)
print, form=13.3,list ='WashEx',J_HCT[19] J_COT[19] J_NXT[19] J_ST[19](13.0)
```

```
print, form=13.3,list ='VaW Ex',J_HCT[23] J_COT[23] J_NXT[23] J_ST[23](13.0)
print, form=13.3,list ='VaS Ex',J_HCT[27] J_COT[27] J_NXT[27] J_ST[27](13.0)
list=' '
print, form=13.3,list =' TOTAL',T_HCT   ,T_COT   ,T_NXT   ,TOTSTART(13.0)


; Soak  Results
;
LIST=' '
LIST=' '
LIST=' Total Hot Soak Emissions (gms) By Jurisdiction'
LIST='                                  soak rate '
LIST='Juris  Soaks(gms)    Trip Stops    gm/trip   '
LIST='------ ------------ ------------ ------------ '
print, form=13.2,list ='  DC  ',J_sk[01]  J_Sp[01](13.0) J_skR[01]
print, form=13.2,list =' MTG  ',J_sk[02]  J_Sp[02](13.0) J_skR[02]
print, form=13.2,list ='  PG  ',J_sk[03]  J_Sp[03](13.0) J_skR[03]
print, form=13.2,list =' ARL  ',J_sk[12]  J_Sp[12](13.0) J_skR[12]
print, form=13.2,list =' ALX  ',J_sk[17]  J_Sp[17](13.0) J_skR[17]
print, form=13.2,list =' FFX  ',J_sk[13]  J_Sp[13](13.0) J_skR[13]
print, form=13.2,list =' LDN  ',J_sk[14]  J_Sp[14](13.0) J_skR[14]
print, form=13.2,list ='  PW  ',J_sk[15]  J_Sp[15](13.0) J_skR[15]
print, form=13.2,list =' FRD  ',J_sk[10]  J_Sp[10](13.0) J_skR[10]
print, form=13.2,list =' HOW  ',J_sk[04]  J_Sp[04](13.0) J_skR[04]
print, form=13.2,list =' AAR  ',J_sk[05]  J_Sp[05](13.0) J_skR[05]
print, form=13.2,list =' CHS  ',J_sk[09]  J_Sp[09](13.0) J_skR[09]
print, form=13.2,list =' CAR  ',J_sk[06]  J_Sp[06](13.0) J_skR[06]
print, form=13.2,list =' CAL  ',J_sk[08]  J_Sp[08](13.0) J_skR[08]
print, form=13.2,list =' STM  ',J_sk[18]  J_Sp[18](13.0) J_skR[18]
print, form=13.2,list ='  KG  ',J_sk[25]  J_Sp[25](13.0) J_skR[25]
print, form=13.2,list =' FBG  ',J_sk[26]  J_Sp[26](13.0) J_skR[26]
print, form=13.2,list =' STA  ',J_sk[16]  J_Sp[16](13.0) J_skR[16]
print, form=13.2,list =' SPT  ',J_sk[24]  J_Sp[24](13.0) J_skR[24]
print, form=13.2,list =' FAU  ',J_sk[21]  J_Sp[21](13.0) J_skR[21]
print, form=13.2,list =' CLK  ',J_sk[20]  J_Sp[20](13.0) J_skR[20]
print, form=13.2,list =' JEF  ',J_sk[22]  J_Sp[22](13.0) J_skR[22]
print, form=13.2,list ='BaltEx',J_sk[07]  J_Sp[07](13.0) J_skR[07]
print, form=13.2,list ='Frd Ex',J_sk[11]  J_Sp[11](13.0) J_skR[11]
print, form=13.2,list ='WashEx',J_sk[19]  J_Sp[19](13.0) J_skR[19]
print, form=13.2,list ='VaW Ex',J_sk[23]  J_Sp[23](13.0) J_skR[23]
print, form=13.2,list ='VaS Ex',J_sk[27]  J_Sp[27](13.0) J_skR[27]
list=' '
print, form=13.2,list =' TOTAL',TOTsk   TOTSTop(13.0)  T_SKR


LIST=' '
LIST=' '
LIST=' '
LIST=' Total Stop Soak Emissions (tons) By Jurisdiction'
LIST='                                                  '
LIST='Juris  Soaks(tns)   Trip Starts '
LIST='------ ------------ ------------  '
print, form=13.3,list ='  DC  ',J_skT[01]  J_Sp[01](13.0)
print, form=13.3,list =' MTG  ',J_skT[02]  J_Sp[02](13.0)
print, form=13.3,list ='  PG  ',J_skT[03]  J_Sp[03](13.0)
print, form=13.3,list =' ARL  ',J_skT[12]  J_Sp[12](13.0)
print, form=13.3,list =' ALX  ',J_skT[17]  J_Sp[17](13.0)
print, form=13.3,list =' FFX  ',J_skT[13]  J_Sp[13](13.0)
print, form=13.3,list =' LDN  ',J_skT[14]  J_Sp[14](13.0)
print, form=13.3,list ='  PW  ',J_skT[15]  J_Sp[15](13.0)
print, form=13.3,list =' FRD  ',J_skT[10]  J_Sp[10](13.0)
print, form=13.3,list =' HOW  ',J_skT[04]  J_Sp[04](13.0)
print, form=13.3,list =' AAR  ',J_skT[05]  J_Sp[05](13.0)
print, form=13.3,list =' CHS  ',J_skT[09]  J_Sp[09](13.0)
print, form=13.3,list =' CAR  ',J_skT[06]  J_Sp[06](13.0)
print, form=13.3,list =' CAL  ',J_skT[08]  J_Sp[08](13.0)
print, form=13.3,list =' STM  ',J_skT[18]  J_Sp[18](13.0)
print, form=13.3,list ='  KG  ',J_skT[25]  J_Sp[25](13.0)
print, form=13.3,list =' FBG  ',J_skT[26]  J_Sp[26](13.0)
print, form=13.3,list =' STA  ',J_skT[16]  J_Sp[16](13.0)
print, form=13.3,list =' SPT  ',J_skT[24]  J_Sp[24](13.0)
print, form=13.3,list =' FAU  ',J_skT[21]  J_Sp[21](13.0)
print, form=13.3,list =' CLK  ',J_skT[20]  J_Sp[20](13.0)
print, form=13.3,list =' JEF  ',J_skT[22]  J_Sp[22](13.0)
print, form=13.3,list ='BaltEx',J_skT[07]  J_Sp[07](13.0)
print, form=13.3,list ='Frd Ex',J_skT[11]  J_Sp[11](13.0)
print, form=13.3,list ='WashEx',J_skT[19]  J_Sp[19](13.0)
print, form=13.3,list ='VaW Ex',J_skT[23]  J_Sp[23](13.0)
print, form=13.3,list ='VaS Ex',J_skT[27]  J_Sp[27](13.0)
list=' '
print, form=13.3,list =' TOTAL',T_skT   , TOTSTop(13.0)

;
; Finally compute, print MSA pollutant, trip totals
; MSA defined as  dc,mtg,pg,arl,alx,ffx,ldn,pw,frd,chs,cal,&sta

MSA_ST = J_ST[01] + J_ST[02] + J_ST[03] + J_ST[08] +
         J_ST[09] + J_ST[10] + J_ST[12] + J_ST[13] +
         J_ST[14] + J_ST[15] + J_ST[16] + J_ST[17]

MSA_SP = J_SP[01] + J_SP[02] + J_SP[03] + J_SP[08] +
         J_SP[09] + J_SP[10] + J_SP[12] + J_SP[13] +
         J_SP[14] + J_SP[15] + J_SP[16] + J_SP[17]

MSA_SKT= J_SKT[01] + J_SKT[02] + J_SKT[03] + J_SKT[08] +
         J_SKT[09] + J_SKT[10] + J_SKT[12] + J_SKT[13] +
         J_SKT[14] + J_SKT[15] + J_SKT[16] + J_SKT[17]

MSA_HCT= J_HCT[01] + J_HCT[02] + J_HCT[03] + J_HCT[08] +
         J_HCT[09] + J_HCT[10] + J_HCT[12] + J_HCT[13] +
         J_HCT[14] + J_HCT[15] + J_HCT[16] + J_HCT[17]

MSA_COT= J_COT[01] + J_COT[02] + J_COT[03] + J_COT[08] +
         J_COT[09] + J_COT[10] + J_COT[12] + J_COT[13] +
         J_COT[14] + J_COT[15] + J_COT[16] + J_COT[17]

MSA_NXT= J_NXT[01] + J_NXT[02] + J_NXT[03] + J_NXT[08] +
         J_NXT[09] + J_NXT[10] + J_NXT[12] + J_NXT[13] +
         J_NXT[14] + J_NXT[15] + J_NXT[16] + J_NXT[17]


list=' '
list=' MSA TOTALS:  '
list=' MSA definition: DC,MTG,PG,ARL,ALX,FFX,LDN,PW,FRD,CHS,CAL,& STA '
list=' '

print, form=13.3,list ='MSA TRIP STARTS:              ',MSA_ST
print, form=13.3,list ='MSA TRIP STOPS :              ',MSA_SP
print, form=13.3,list ='MSA HC Start Emissions(Tons): ',MSA_HCT
print, form=13.3,list ='MSA CO Start Emissions(Tons): ',MSA_COT
print, form=13.3,list ='MSA NX Start Emissions(Tons): ',MSA_NXT
print, form=13.3,list ='MSA Hot Soak Emissions(Tons): ',MSA_SKT

ENDIF
ENDRUN
*copy tppl*.PRN stRT_SKR.rpt
```

ZONESPRD.S

```
*DEL TPPL*.PRN
;--------------------------------------------------------------------
;ZONESPRD.S - This program reads zonal modeled AM/PM/Off-Pk Prd trips
;             and allocates them to each hour.  Output is a zone file
;             of hourly origins and destinations.
;--------------------------------------------------------------------
;
; The input file is named: trip.end (from the aqtrip.s program)
; The output files are:    hrlyorig.asc (hourly trip origins file)
;                          hrlydest.asc (hourly trip destins.file)
;
;

run pgm=matrix
zones=2191
ZDATI[1] = trip.end,z=1-5,day_o=6-15,day_d=16-25,
                          am_o=26-35, am_d=36-45,
                          pm_o=46-55, pm_d=56-65,
                          op_o=66-75, op_d=76-85

; Set up arrays

ARRAY,
HORIG=24, ; hrly Origins
HDEST=24  ; hrly Destinations

;Establish  Hrly distributions for AM, PM and Off-Pk Period Trips
;Curves taken from 1994 HTS Auto Dr. Trips in motion distributions
;Note: Hours range from 1-24, instead of 0-23
; Pcts of last hour (24) may be adjusted slightly to force normalization

LOOKUP NAME=HR_DST,           ; <<-- HRLY Trip Distributions
       LOOKUP[1]=1, RESULT=2,  ; AM
       LOOKUP[2]=1, RESULT=3,  ; PM
       LOOKUP[3]=1, RESULT=4,  ; OP
       INTERPOLATE = N,LIST=T,

R= " 1, 0.000, 0.000, 0.003, ",
   " 2, 0.000, 0.000, 0.004, ",
   " 3, 0.000, 0.000, 0.003, ",
   " 4, 0.000, 0.000, 0.003, ",
   " 5, 0.000, 0.000, 0.005, ",
   " 6, 0.000, 0.000, 0.022, ",
   " 7, 0.201, 0.000, 0.000, ",
   " 8, 0.398, 0.000, 0.000, ",
   " 9, 0.401, 0.000, 0.000, ",
   "10, 0.000, 0.000, 0.097, ",
   "11, 0.000, 0.000, 0.082, ",
   "12, 0.000, 0.000, 0.092, ",
   "13, 0.000, 0.000, 0.101, ",
   "14, 0.000, 0.000, 0.089, ",
   "15, 0.000, 0.000, 0.090, ",
   "16, 0.000, 0.000, 0.116, ",
   "17, 0.000, 0.314, 0.000, ",
   "18, 0.000, 0.373, 0.000, ",
   "19, 0.000, 0.313, 0.000, ",
   "20, 0.000, 0.000, 0.108, ",
   "21, 0.000, 0.000, 0.077, ",
   "22, 0.000, 0.000, 0.058, ",
   "23, 0.000, 0.000, 0.034, ",
   "24, 0.000, 0.000, 0.016, "

;--------------------------------------------------------------------
; NOTE: The hourly arrays are indexed from 1 to 24, as follows:
; 1-  12mid - 12:59AM       13- 12noon - 12:59PM
; 2- 1:00AM -  1:59AM       14- 1:00PM -  1:59PM
; 3- 2:00AM -  2:59AM       15- 2:00PM -  2:59PM
; 4- 3:00AM -  3:59AM       16- 3:00PM -  3:59PM
; 5- 4:00AM -  4:59AM       17- 4:00PM -  4:59PM
; 6- 5:00AM -  5:59AM       18- 5:00PM -  5:59PM
; 7- 6:00AM -  6:59AM       19- 6:00PM -  6:59PM
; 8- 7:00AM -  1:59AM       20- 7:00PM -  7:59PM
; 9- 8:00AM -  2:59AM       21- 8:00PM -  8:59PM
;10- 9:00AM -  3:59AM       22- 9:00PM -  9:59PM
;11-10:00AM -  4:59AM       23-10:00PM - 10:59PM
;12-11:00AM - 11:59AM       24-11:00PM - 11:59PM
;--------------------------------------------------------------------
;
 SUM_IO = SUM_IO + am_o[i] + pm_o[i] + op_o[i] ; sum of input origins
 SUM_ID = SUM_ID + am_d[i] + pm_d[i] + op_d[i] ; sum of input destinations
;--------------------------------------------------------------------
; Hourly Origin Trip Allocation                                      -
;--------------------------------------------------------------------

LOOP IDX=1,24
 HORIG[IDX] = (am_o[i] * HR_DST(1,IDX)) +
              (pm_o[i] * HR_DST(2,IDX)) +
              (op_o[i] * HR_DST(3,IDX))

 HDEST[IDX] = (am_d[i] * HR_DST(1,IDX)) +
              (pm_d[i] * HR_DST(2,IDX)) +
              (op_d[i] * HR_DST(3,IDX))

;
; Let's accumulate hourly Origins/Destinations to make sure 'Outs=Ins'
;
 SUM_HO = SUM_HO + HORIG[IDX]  ; sum of hourly origins
 SUM_HD = SUM_HD + HDEST[IDX]  ; sum of hourly destinations

ENDLOOP
;
; List out Hourly hourly Origins/Destinations
;
LIST = I(5),
       HORIG[01](10.2),HORIG[02](10.2),HORIG[03](10.2),HORIG[04](10.2),
       HORIG[05](10.2),HORIG[06](10.2),HORIG[07](10.2),HORIG[08](10.2),
       HORIG[09](10.2),HORIG[10](10.2),HORIG[11](10.2),HORIG[12](10.2),
       HORIG[13](10.2),HORIG[14](10.2),HORIG[15](10.2),HORIG[16](10.2),
       HORIG[17](10.2),HORIG[18](10.2),HORIG[19](10.2),HORIG[20](10.2),
       HORIG[21](10.2),HORIG[22](10.2),HORIG[23](10.2),HORIG[24](10.2),
       file=HRLYORIG.ASC

LIST = I(5),
       HDEST[01](10.2),HDEST[02](10.2),HDEST[03](10.2),HDEST[04](10.2),
       HDEST[05](10.2),HDEST[06](10.2),HDEST[07](10.2),HDEST[08](10.2),
       HDEST[09](10.2),HDEST[10](10.2),HDEST[11](10.2),HDEST[12](10.2),
       HDEST[13](10.2),HDEST[14](10.2),HDEST[15](10.2),HDEST[16](10.2),
       HDEST[17](10.2),HDEST[18](10.2),HDEST[19](10.2),HDEST[20](10.2),
       HDEST[21](10.2),HDEST[22](10.2),HDEST[23](10.2),HDEST[24](10.2),
       file=HRLYDEST.ASC

endrun
; ---------------------------------------------------------------
*copy tppl*.prn zonesprd.rpt
```

Attachment 2

Example Emission Rate Listings

Montgomery County- Case 7 Ozone Season Emission Rates

M605C7MC.R_A (Arterial Running VOC, CO, and NOx Rates by Speed Increment)

1.	4.506	20.320	2.505
2.	4.506	20.320	2.505
3.	3.529	17.444	2.399
4.	2.308	13.848	2.267
5.	1.576	11.691	2.188
6.	1.306	10.285	2.046
7.	1.113	9.282	1.945
8.	.967	8.529	1.870
9.	.855	7.944	1.811
10.	.765	7.475	1.764
11.	.705	7.127	1.690
12.	.655	6.837	1.629
13.	.613	6.592	1.577
14.	.576	6.382	1.533
15.	.545	6.199	1.495
16.	.514	6.031	1.457
17.	.487	5.883	1.423
18.	.463	5.751	1.394
19.	.441	5.633	1.367
20.	.421	5.527	1.343
21.	.405	5.460	1.321
22.	.391	5.398	1.302
23.	.378	5.342	1.283
24.	.365	5.291	1.267
25.	.354	5.244	1.251
26.	.344	5.231	1.238
27.	.334	5.219	1.227
28.	.326	5.208	1.216
29.	.319	5.198	1.205
30.	.312	5.189	1.196
31.	.304	5.222	1.191
32.	.297	5.253	1.187
33.	.291	5.282	1.183
34.	.285	5.309	1.179
35.	.280	5.335	1.175
36.	.275	5.433	1.181
37.	.271	5.524	1.187
38.	.267	5.612	1.192
39.	.265	5.694	1.197
40.	.261	5.773	1.202
41.	.258	5.873	1.212
42.	.254	5.968	1.223
43.	.252	6.058	1.232
44.	.248	6.145	1.242
45.	.246	6.227	1.250
46.	.243	6.330	1.266
47.	.240	6.428	1.280
48.	.237	6.522	1.294
49.	.235	6.612	1.308
50.	.233	6.698	1.321
51.	.231	6.803	1.342
52.	.229	6.904	1.363
53.	.227	7.002	1.382
54.	.223	7.095	1.401
55.	.222	7.185	1.420
56.	.221	7.301	1.449
57.	.219	7.412	1.477
58.	.219	7.520	1.504
59.	.217	7.623	1.531
60.	.216	7.724	1.556
61.	.215	7.844	1.597
62.	.215	7.960	1.636
63.	.214	8.073	1.674
64.	.213	8.182	1.711
65.	.213	8.287	1.746

M605C7MC.R_F (Non-Ramp Freeway Running VOC, CO, and NOx Rates by Speed Increment)

1.	5.647	22.960	1.996
2.	5.647	22.960	1.996
3.	4.390	19.544	1.901
4.	2.817	15.274	1.783
5.	1.874	12.712	1.713
6.	1.522	10.965	1.529
7.	1.271	9.718	1.398
8.	1.083	8.782	1.300
9.	.937	8.055	1.224
10.	.819	7.472	1.163
11.	.747	7.102	1.090
12.	.686	6.793	1.029
13.	.634	6.531	.978
14.	.591	6.307	.934
15.	.553	6.113	.896
16.	.526	6.053	.894
17.	.502	5.999	.892
18.	.481	5.951	.891
19.	.462	5.909	.890
20.	.445	5.870	.889
21.	.431	5.858	.887
22.	.419	5.848	.885
23.	.407	5.838	.884
24.	.398	5.829	.883
25.	.388	5.821	.882
26.	.378	5.823	.880
27.	.370	5.826	.879
28.	.363	5.828	.879
29.	.355	5.830	.878
30.	.348	5.832	.877
31.	.341	5.879	.875
32.	.334	5.924	.874
33.	.327	5.965	.873
34.	.320	6.005	.871
35.	.314	6.042	.870
36.	.310	6.159	.874
37.	.305	6.269	.877
38.	.301	6.374	.880
39.	.298	6.473	.883
40.	.294	6.568	.885
41.	.290	6.685	.891
42.	.287	6.796	.896
43.	.283	6.903	.901
44.	.280	7.004	.905
45.	.277	7.101	.910
46.	.273	7.218	.916
47.	.270	7.331	.923
48.	.267	7.438	.929
49.	.263	7.541	.934
50.	.261	7.640	.940
51.	.258	7.758	.948
52.	.255	7.871	.956
53.	.253	7.980	.963
54.	.250	8.085	.971
55.	.248	8.186	.978
56.	.246	8.313	.988
57.	.243	8.435	.998
58.	.243	8.552	1.007
59.	.241	8.666	1.016
60.	.239	8.776	1.025
61.	.238	8.903	1.038
62.	.236	9.026	1.050
63.	.234	9.145	1.063
64.	.233	9.261	1.074
65.	.233	9.373	1.086

M605C7MC.RAM (Freeway Ramp Running and Soak VOC, CO, and NOx Rates)

35.	.367	10.663	1.318	.589	.643	2.347

M605C7MC.STT (Hot Transient and Cold Start VOC, CO, and NOx Rates)

35.	.226	2.459	.183	1.352	21.028	.855