KENTUCKY TRANSPORTATION CENTER

Update of Data for Estimating ESALs

Kentucky Transportation Center Research Report — KTC15-26/PL21-13-1F
DOI: http://dx.doi.org/10.13023/KTC.RR.2015.26

UNIVERSITY OF KENTUCKY College of Engineering

KTC's Mission

We provide services to the transportation community through research, technology transfer, and education. We create and participate in partnerships to promote safe and effective transportation systems.

Kentucky Transportation Center
176 Oliver H. Raymond Building
Lexington, KY 40506-0281
(859) 257-4513

www.ktc.uky.edu

Research Report
KTC-15-26/PL21-13-1F

Update of Data for Estimating ESALs

By
Jerry G. Pigman, P.E.
Program Manager

Clark Graves, Ph.D., P.E.
Associate Director

Brad Rister, P.E.
Program Manager

David Cain
Transportation Technologist

and

Neil Tollner
IT Services Coordinator

Kentucky Transportation Center
College of Engineering
University of Kentucky
Lexington, Kentucky

In cooperation with
Kentucky Transportation Cabinet
Commonwealth of Kentucky

December 2015

1. Report No. KTC-15-26/PL21-13-1F	2. Government Accession No.	3. Recipient's Catalog No
4. Title and Subtitle Update of Data for Estimating ESALs		5. Report Date December 2015 6. Performing Organization Code
7. Author(s): Jerry Pigman, Clark Graves, Brad Rister, David Cain, and Neil Tollner		8. Performing Organization Report No. KTC-15-26/PL21-13-1F
9. Performing Organization Name and Address Kentucky Transportation Center College of Engineering University of Kentucky Lexington, KY 40506-0281		10. Work Unit No. (TRAIS) 11. Contract or Grant No. PL21-13
12. Sponsoring Agency Name and Address Kentucky Transportation Cabinet State Office Building Frankfort, KY 40622		13. Type of Report and Period Covered 14. Sponsoring Agency Code

15. Supplementary Notes
Prepared in cooperation with the Kentucky Transportation Cabinet and the Federal Highway Administration

16. Abstract

This project involved updating processing traffic characteristics data using a series of quality control and analytical programs to produce an estimate of the following parameters of interest; 1) average daily traffic, 2) percent trucks, 3) percent trucks classified as heavy/coal, 4) axles per truck, 5) axles per heavy/coal truck, 6) ESALs per truck axle, 7) ESALs per heavy/coal truck axle, and 8) total ESALs. ESAL estimation parameters used WIM data collected during 2007, 2011, and 2012–2013 (representing 41 stations); and classification data collected in 2010, 2011 and 2012 (representing 1,669 stations).

Computer programs used to process classification data, process weight data, and then combining output to calculate ESALs are contained in the three following programs; 1) CLASS SUMMARY – processes vehicle classification data and produces annual average number of vehicle types at each classification station, 2) LOADOMTR SUMMARY – processes truck weight data to produce axle load distributions by vehicle type, and 3) AGGCALC – processes output form LOADOMTR AND CLASS programs to produce ESAL-related parameters of interest. A flowchart which provides steps of processing data and calculating estimates of ESALs is included in Appendix A and computer code for each of these programs is included in Appendix B.

17. Key Words Equivalent Single Axle Loads; Weigh-in-Motion Data; Classification Data; Traffic Volume Data		18. Distribution Statement Unlimited, with approval of Kentucky Transportation Cabinet	
19. Security Classification (report) Unclassified	20. Security Classification (this page) Unclassified	21. No. of Pages 271	19. Security Classification (report)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1.0 BACKGROUND AND OVERVIEW

A revised procedure for estimating equivalent single axleloads (ESALs) was developed in 1985. This procedure used weight, classification, and traffic volume data collected by the Kentucky Transportation Cabinet's (KYTC) Division of Planning. The procedure was described and documented in Research Report UKTRP-85-30, "Estimation of Equivalent Axleloads." There have been several changes made to the revised procedure since its adoption, including its documentation, which is described in Report KTC-95-7, "Equivalent Single Axleload Computer Program Enhancements."

The results presented in Report KTC-95-7 addressed improvements in processing ESAL data. These improvements were made by aggregating functional classes to increase the accuracy of within-class estimates and converting computer programs used for processing data into versions usable on personal computers.

In addition, Report KTC-99-1, "Development of ESAL Forecasting Procedures for Superpave Pavement Design," discussed analytical methods used to prepare ESAL estimates for Superpave projects. The methodologies integrated an interactive program that let users access various databases in order to calculate ESAL Superpave estimates. Work undertaken in 2002 led to updates of this interactive program. Recent efforts have concentrated on the evaluation and calibration of WIM data collection equipment. Report KTC-04-12, "Assessment of Data Collection for ESAL Determinations for the Kentucky Transportation Cabinet, Division of Planning," was completed in 2004 and included recommendations to more intensively monitor WIM equipment.

For the past 30 years the Kentucky Transportation Center (KTC) has provided KYTC with annual update of data used to estimate ESALs. The last update came in 2008 with the publication of Report KTC 08-32/PL13-08-1F, titled "Annual Update of Data for Estimating ESALs." This update entailed processing traffic characteristics data using a series of quality control and analytical programs to produce an estimate of the following parameters of interest.

- Average daily traffic
- Percent trucks
- Percent trucks classified as heavy/coal
- Axles per truck
- Axles per heavy/coal truck
- ESALs per truck axle
- ESALs per heavy/coal truck axle
- Total ESALs

Computer programs used to process classification data, process weight data, and then combining output to calculate ESALS are contained in the three programs listed below.

- CLASS SUMMARY – processes vehicle classification data and produces annual average number of vehicle types at each classification station

- LOADOMTR SUMMARY – processes truck weight data to produce axleload distributions by vehicle type
- AGGCALC – processes output form LOADOMTR AND CLASS programs to produce ESAL-related parameters of interest

A flowchart which provides steps of processing data and calculating estimates of ESALs is included in Appendix A and computer code for the programs are included in Appendix B.

Table 1 summarizes the ESAL parameters and their associated growth rates. The research team, in conjunction with representatives from the Division of Planning, developed default values for growth rate to use when forecasting ESALs. Recommended growth rates provided by the Kentucky Transportation Cabinet's Division of Planning were based on review of monthly ATR data, traffic count data from approximately 13,000 stations, and calculated growth rates based on trend lines of historical data. These default values have also been included in the update of the ESAL Forecasting Program used by KYTC to select Superpave mix parameters.

This report documents and summarizes ESAL estimation parameters using WIM data collected during 2007, 2011, and 2012–2013 (representing 41 stations). Classification data for this current update were collected in 2010, 2011 and 2012 (representing 1,669 stations).

Tracking the WIM calibration process has continued with evaluations of the 2012–2013 data and by highlighting trends within the six aggregate classes. In addition, the historical trends of traffic characteristics data used to estimate ESALs have been updated using data collected through the end of 2013.

To facilitate the interpretation and use of data presented in this report Section 2.0 includes reference tables. These include the following information:

- Functional classifications by number and the associated description as well as cross-references from functional class to aggregate class
- Descriptions and diagrams for each the 13 Federal Highway Administration vehicle classifications
- Kentucky counties and their respective numbers

To represent the inter-annual trends in ESAL factors, the research team has assembled summaries for each aggregate class from 1993 to 2013. Section 3.0 contains this information. Figures 1 through 6 present these summaries. Each figure contains tabular data for AADT, percent trucks, axles/truck, ESAL's/axle, and ESAL's/vehicle. Graphs inform on the percent trucks, axles/truck, and ESAL's/vehicle. Data for 2002, 2003, and 2006 were not included due to equipment calibration and issues with data collection from the WIM sites. Additionally, in other years there are data that appear to be somewhat anomalous based on historical trends. WIM data were available for the year labeled 2007 for Aggregate ESAL Classes I, II, IV, and V (however, most of these data were collected in May of 2008). Data were not available for Aggregate ESAL Classes III or VI. For these years the 2005 data were replicated as a placeholder for the 2007 data. Figures 1-6 also display the number of WIM stations associated with the 2013 data.

The primary objective of analyzing and evaluating traffic characteristics data was to interpret and summarize parameters typically used to estimate total ESALs. Section 4.0 includes vehicle classification and weight data collected in 2013 that were combined with previous years' data to produce average values for each of the six aggregate classes.

In addition, regression analyses were conducted to produce smoothed values for each parameter of interest to eliminate inter-annual variability that results from site-specific influences. These influences stem from limited coverage of the functional and aggregate classes. This information is presented in Section 6.0 for each of the six aggregate classes.

Table 1 Summary of ESAL Forecasting Parameters--2013 Update

Functional Class	Functional Class Description	AADT	Growth AADT (%)	Percent Trucks (%)	Growth Percent Trucks (%)	Axles/ Truck	Growth Axles/ Truck (%)	ESAL's/ Axle	Growth ESAL's/ Axle (%)	Axles/ Coal Truck	Growth Axles/ Coal Truck (%)	ESAL's/ Coal Axle	Growth ESAL's/ Coal Axle (%)
1	Rural Interstate	30,107	1.00	25.3	0.50	4.377	0.00	.280	2.00	0.000	0.00	0.000	0.00
2	Rural Principal Arterial	6,621	0.50	13.2	1.00	3.504	0.00	.291	1.00	5.077	0.00	3.448	1.50
6	Rural Minor Arterial	6,621	0.50	13.2	0.00	3.504	0.00	.291	1.00	5.077	0.00	3.448	1.50
7	Rural Major Collector	2,489	0.50	10.0	0.00	3.077	0.00	.423	2.00	4.000	-2.00	3.500	2.00
8	Rural Minor Collector	2,489	0.50	10.0	0.00	3.077	0.00	.423	2.00	4.000	-2.00	3.500	2.00
9	Rural Local	2,489	0.00	10.0	0.00	3.077	0.00	.423	2.00	4.000	-2.00	3.500	2.00
11	Urban Interstate	38,648	1.00	13.8	0.50	4.212	0.00	.293	1.00	4.808	0.00	1.726	0.00
12	Urban Freeway or Expressway	18,333	0.50	9.4	0.50	3.736	1.00	.305	2.00	4.487	1.00	2.683	2.00
14	Urban Principal Arterial	18,333	0.50	9.4	0.50	3.736	1.00	.305	2.00	4.487	1.00	2.683	2.00
16	Urban Minor Arterial	7,370	0.50	7.9	0.00	3.088	0.50	.821	2.00	5.000	0.00	2.275	2.00
17	Urban Collector	7,370	0.50	7.9	0.00	3.088	0.50	.821	2.00	5.000	0.00	2.275	2.00
19	Urban Local	7,370	0.00	7.9	0.00	3.088	0.50	.821	2.00	5.000	0.00	2.275	2.00

2.0 DEFINITIONS AND REFERENCE TABLES

Table 2 Functional Classification

Rural	Urban
01 Principal Arterial – Interstate	11 Principal Arterial – Interstate
02 Principal Arterial – Other Minor	12 Principal Arterial – Other Freeways/Expressways
06 Arterial	14 Other Principal Arterial
07 Major Collector	16 Minor Arterial Collectors
08 Minor Collector	17 Collector
09 Local	19 Local

Table 3 Aggregate ESAL Groups

Rural		Urban	
I	FC 1	IV	FC 11
II	FC 2 and 6	V	FC 12 and 14
III	FC 7, 8, and 9	VI	FC 16, 17, and 19

Figure 1 FHWA Vehicle Classifications

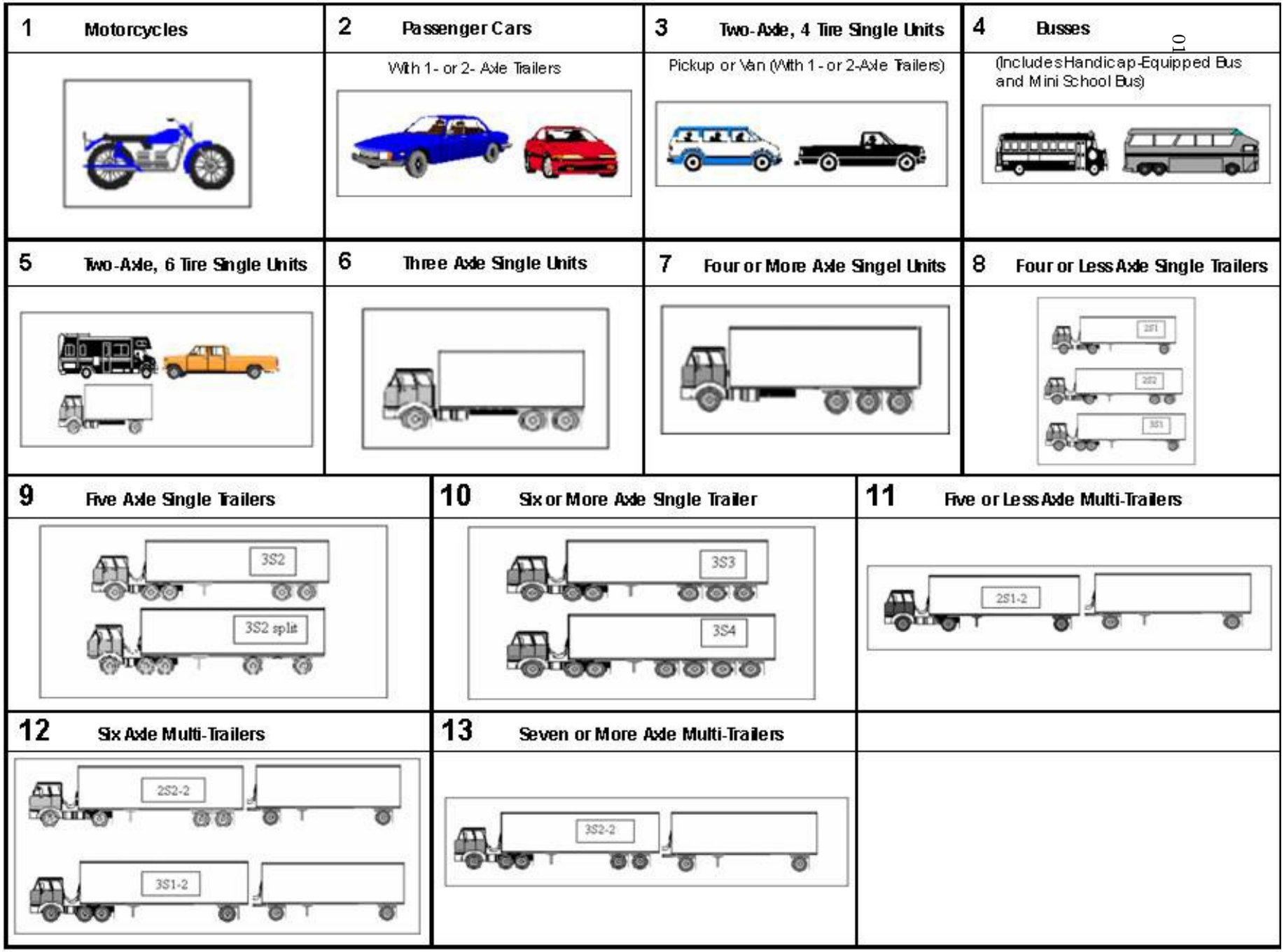


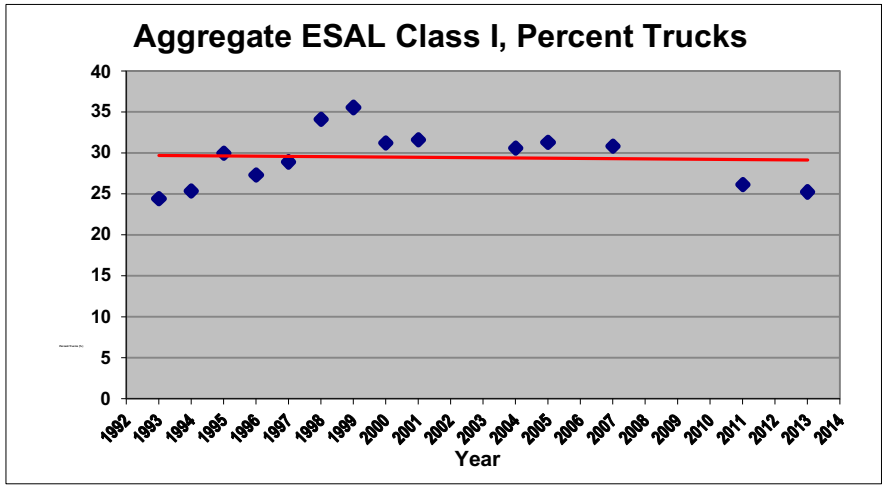
Table 4 Kentucky Counties

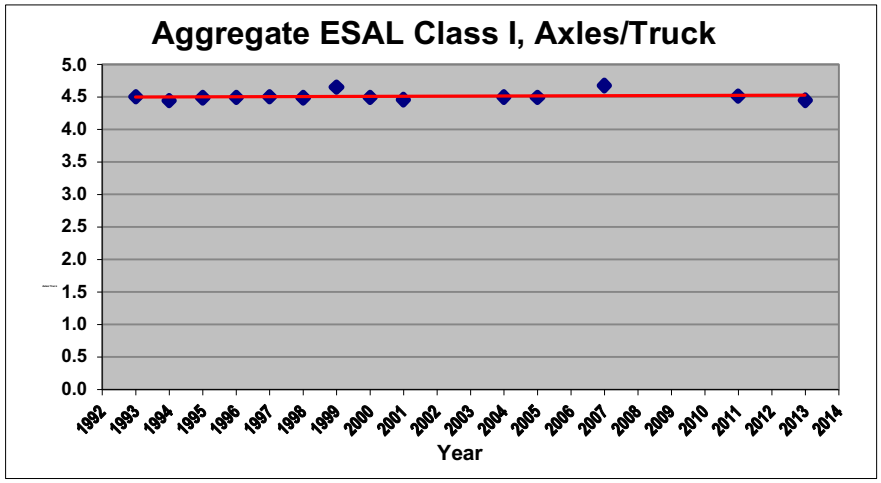
No.	County	No.	County	No.	County
1	Adair	41	Grant	81	Mason
2	Allen	42	Graves	82	Meade
3	Anderson	43	Grayson	83	Menifee
4	Ballard	44	Green	84	Mercer
5	Barren	45	Greenup	85	Metcalf
6	Bath	46	Hancock	86	Monroe
7	Bell	47	Hardin	87	Montgomery
8	Boone	48	Harlan	88	Morgan
9	Bourbon	49	Harrison	89	Muhlenburg
10	Boyd	50	Hart	90	Nelson
11	Boyle	51	Henderson	91	Nicholas
12	Bracken	52	Henry	92	Ohio
13	Breathitt	53	Hickman	93	Oldham
14	Breckenridge	54	Hopkins	94	Owen
15	Bullitt	55	Jackson	95	Owsley
16	Butler	56	Jefferson	96	Pendleton
17	Caldwell	57	Jessamine	97	Perry
18	Calloway	58	Johnson	98	Pike
19	Campbell	59	Kenton	99	Powell
20	Carlisle	60	Knott	100	Pulaski
21	Carroll	61	Knox	101	Robertson
22	Carter	62	Larue	102	Rockcastle
23	Casey	63	Laurel	103	Rowan
24	Christian	64	Lawrence	104	Russell
25	Clark	65	Lee	105	Scott
26	Clay	66	Leslie	106	Shelby
27	Clinton	67	Letcher	107	Simpson
28	Crittenden	68	Lewis	108	Spencer
29	Cumberland	69	Lincoln	109	Taylor
30	Daviess	70	Livingston	110	Todd
31	Edmonson	71	Logan	111	Trigg
32	Elliott	72	Lyon	112	Trimble
33	Estill	73	McCracken	113	Union
34	Fayette	74	McCreary	114	Warren
35	Fleming	75	McLean	115	Washington
36	Floyd	76	Madison	116	Wayne
37	Franklin	77	Magoffin	117	Webster
38	Fulton	78	Marion	118	Whitley
39	Gallatin	79	Marshall	119	Wolfe
40	Garrard	80	Martin	120	Woodford

3.0 HISTORICAL DATA TRENDS BY AGGREGATE CLASS

Figure 2 Aggregate ESAL Class I (FC 01), 2013 -- 1 site

Year	AADT	% Trucks	Axles/Truck	EAL's/Axle	EAL's/veh
1993	23,935	24.46	4.502	0.201	0.90
1994	27,754	25.38	4.442	0.206	0.92
1995	25,427	29.99	4.485	0.215	0.96
1996	22,292	27.30	4.489	0.231	1.04
1997	34,777	28.93	4.501	0.208	0.94
1998	23,935	34.12	4.486	0.226	1.01
1999	26,600	35.56	4.648	0.220	1.02
2000	28,424	31.22	4.489	0.258	1.16
2001	31,586	31.64	4.456	0.239	1.06
2004	33,594	30.63	4.495	0.322	1.45
2005	35,488	31.31	4.492	0.302	1.36
2007	32,616	30.86	4.673	0.194	0.91
2011	34,277	26.15	4.510	0.267	1.20
2013	34,220	25.26	4.447	0.332	1.48





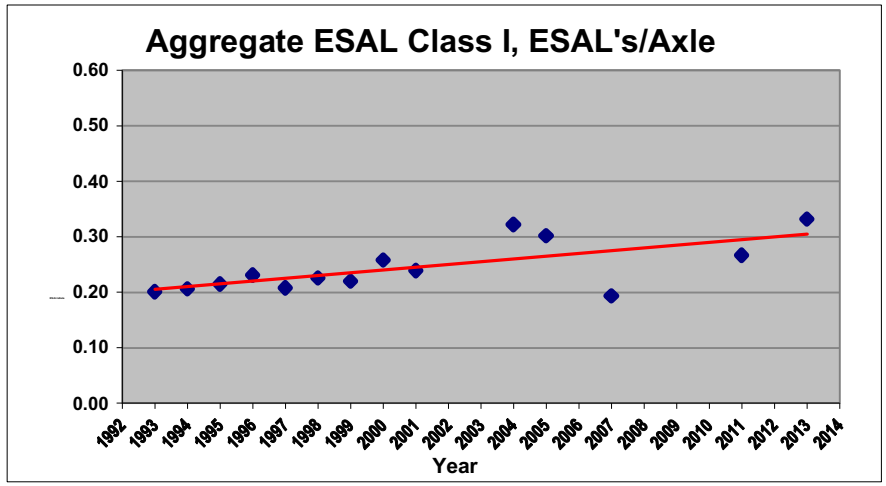
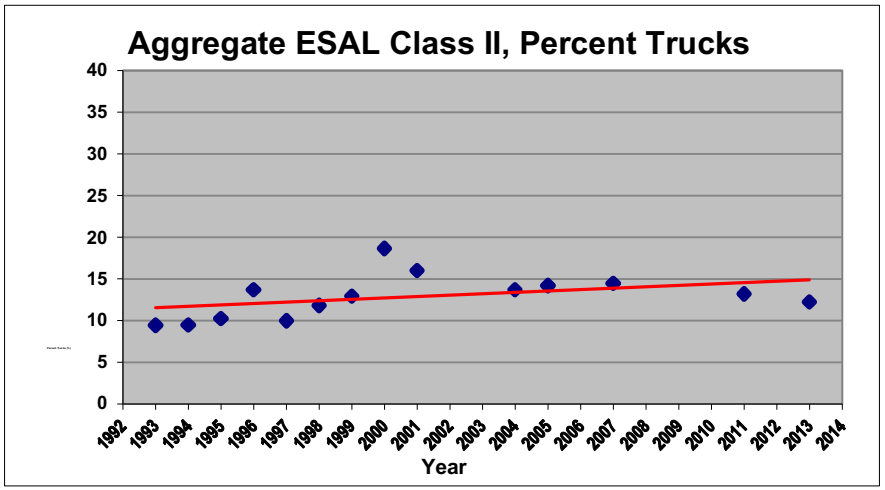
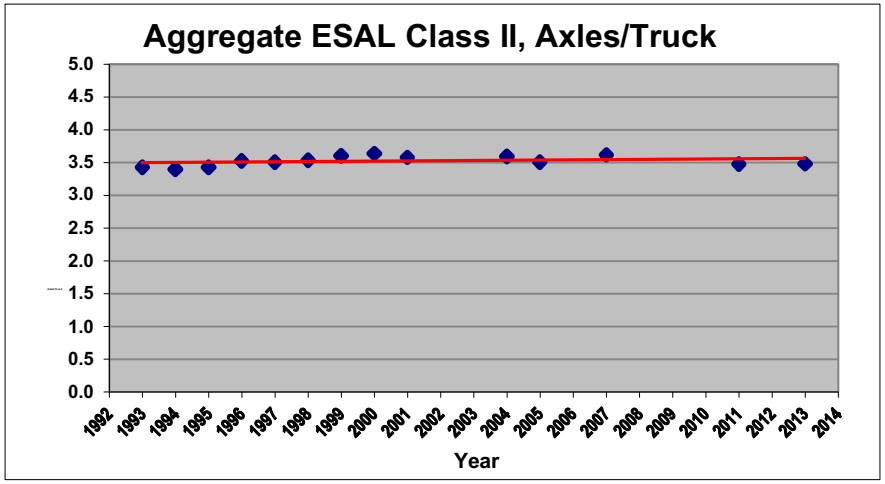


Figure 3 Aggregate ESAL Class II (FC 02, 06), 2013 -- 6 sites

Year	AADT	% Trucks	Axles/Truck	EAL's/Axle	EAL's/veh
1993	5,566	9.48	3.426	0.205	0.70
1994	5,730	9.51	3.399	0.214	0.73
1995	5,453	10.26	3.426	0.235	0.81
1996	6,439	13.72	3.522	0.248	0.87
1997	8,954	10.00	3.504	0.271	0.95
1998	7,603	11.84	3.532	0.275	0.97
1999	7,879	12.96	3.603	0.284	1.02
2000	6,242	18.69	3.637	0.286	1.04
2001	7,545	16.02	3.577	0.283	1.01
2004	7,081	13.75	3.591	0.236	0.85
2005	7,749	14.23	3.505	0.270	0.95
2007	7,057	14.48	3.617	0.278	1.01
2011	6,688	13.24	3.473	0.260	0.90
2013	6,557	12.29	3.479	0.346	1.20





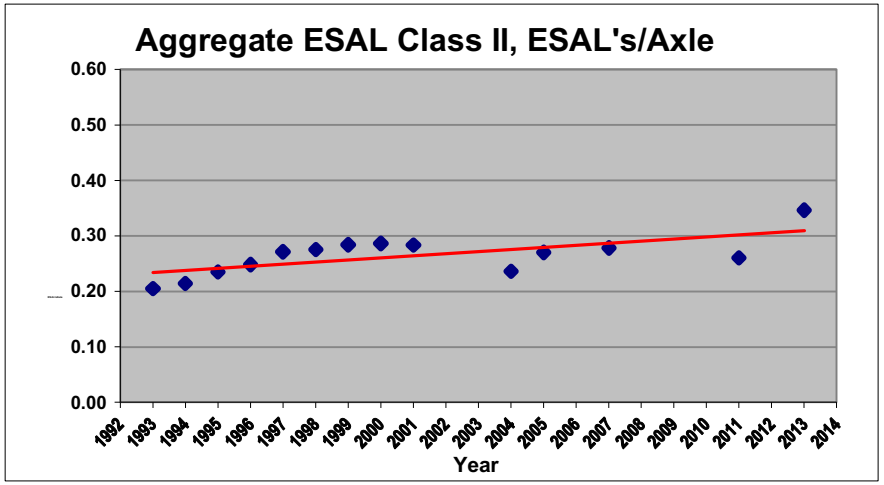
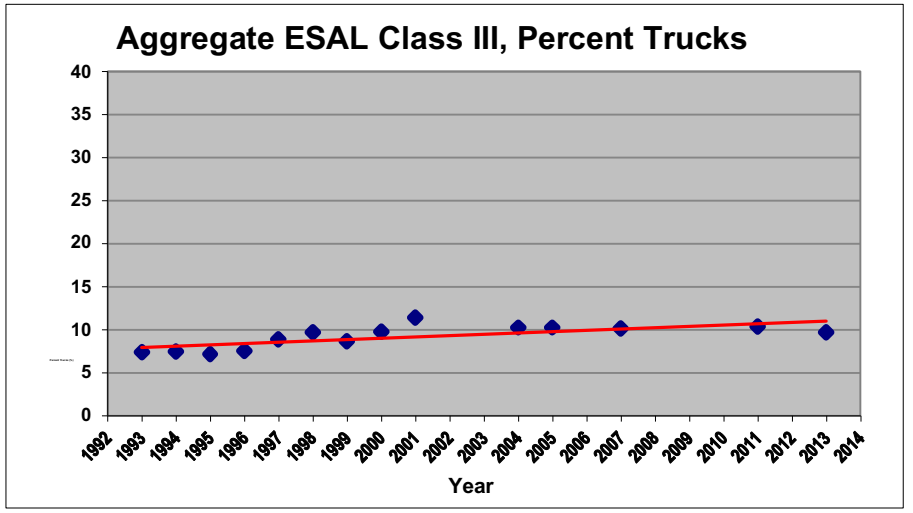
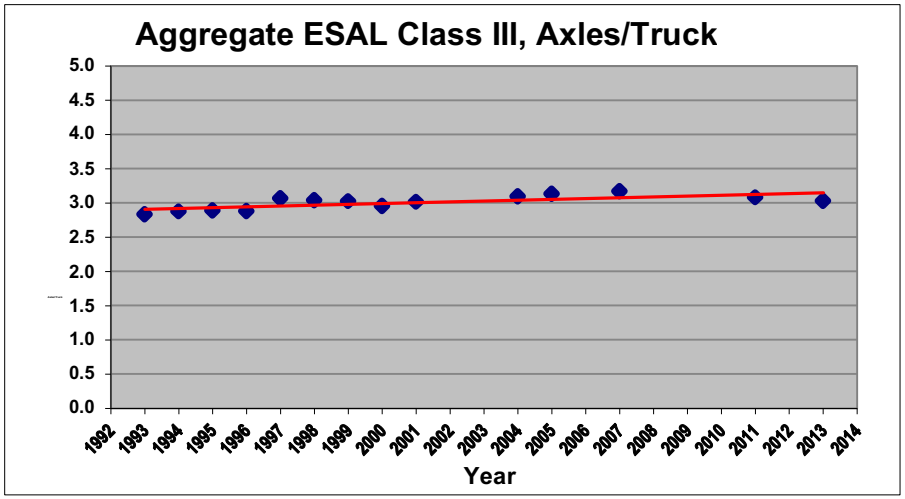


Figure 4 Aggregate ESAL Class III (FC 07, 08, 09), 2013 -- 3 sites

Year	AADT	% Trucks	Axles/Truck	EAL's/Axle	EAL's/veh
1993	3,048	7.34	2.832	0.265	0.75
1994	3,037	7.42	2.872	0.249	0.72
1995	3,334	7.12	2.886	0.219	0.63
1996	3,078	7.48	2.877	0.214	0.62
1997	4,959	8.85	3.061	0.226	0.69
1998	4,285	9.69	3.038	0.222	0.67
1999	3,337	8.60	3.023	0.194	0.59
2000	2,040	9.75	2.953	0.233	0.69
2001	2,625	11.36	3.011	0.234	0.70
2004	2,506	10.21	3.095	0.505	1.56
2005	2,672	10.21	3.129	0.550	1.72
2007	2,557	10.10	3.165	0.571	1.81
2011	2,510	10.31	3.079	0.348	1.07
2013	2,418	9.63	3.027	0.452	1.37





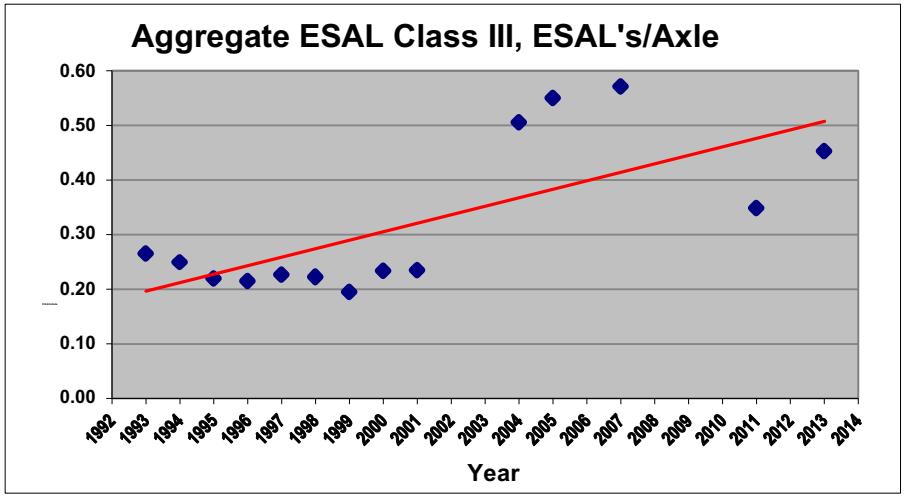
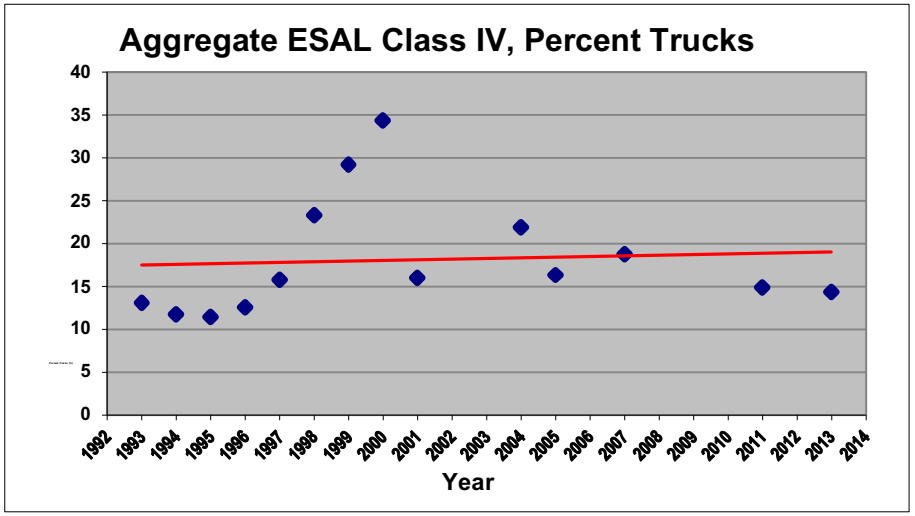
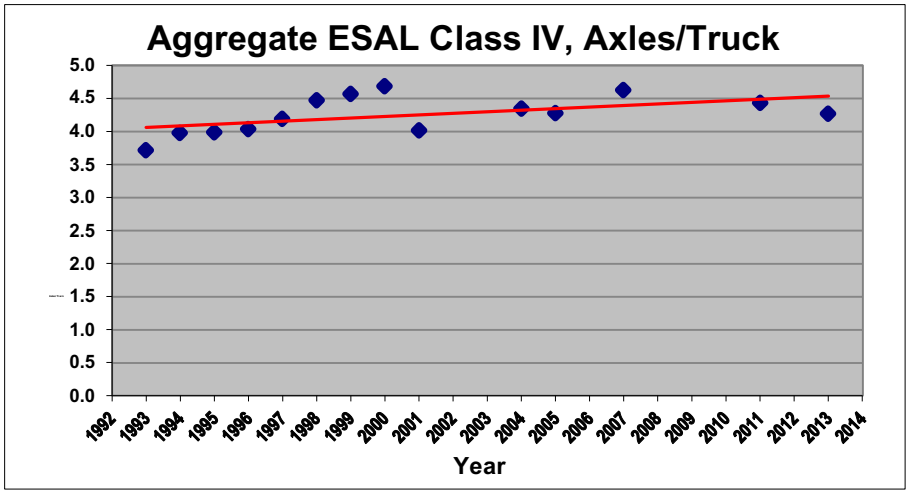


Figure 5 Aggregate ESAL Class IV (FC 11), 2013 -- 2 sites

Year	AADT	% Trucks	Axles/Truck	EAL's/Axle	EAL's/veh
1993	49,584	13.02	3.715	0.181	0.67
1994	64,412	11.73	3.975	0.187	0.74
1995	99,690	11.39	3.981	0.179	0.71
1996	65,325	12.52	4.033	0.181	0.73
1997	66,585	15.71	4.185	0.187	0.78
1998	34,475	23.29	4.470	0.231	1.03
1999	28,000	29.14	4.563	0.241	1.10
2000	39,600	34.33	4.683	0.245	1.15
2001	45,087	15.93	4.012	0.339	1.36
2004	44,333	21.82	4.341	0.272	1.18
2005	44,802	16.29	4.270	0.268	1.14
2007	42,863	18.70	4.624	0.218	1.01
2011	45,498	14.87	4.431	0.282	1.25
2013	45,556	14.32	4.267	0.351	1.50





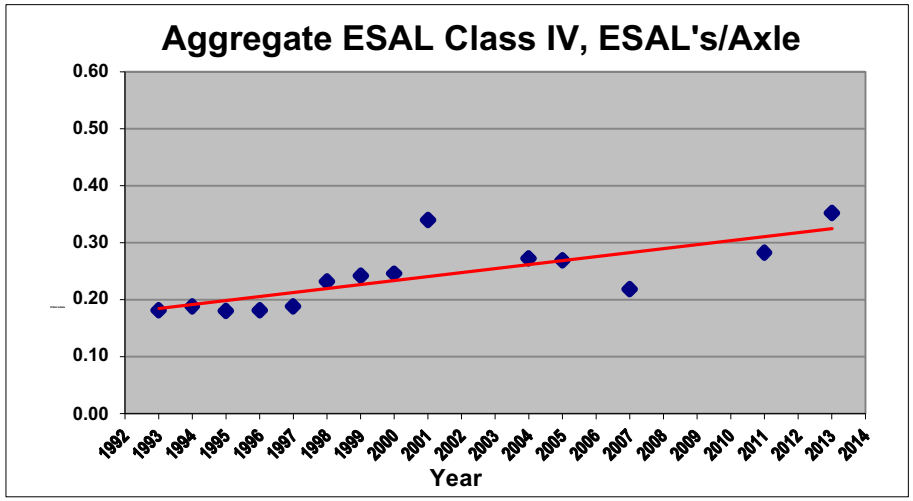
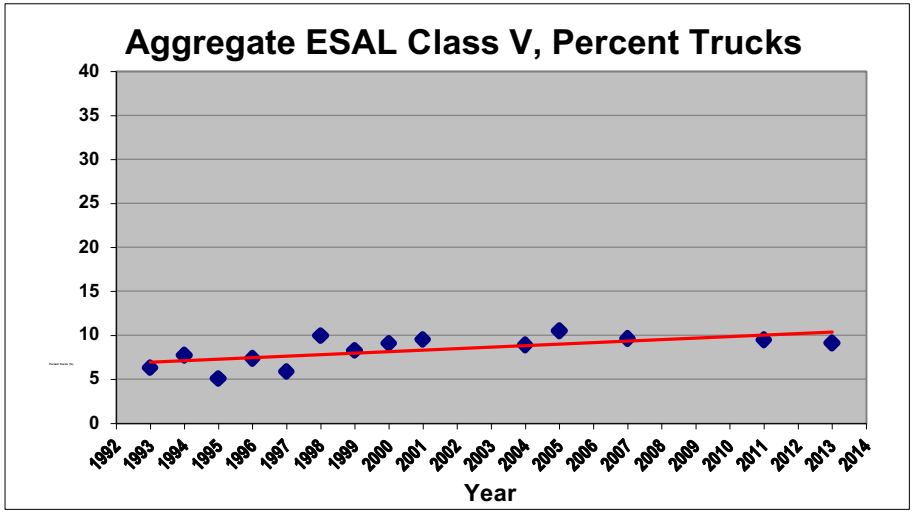
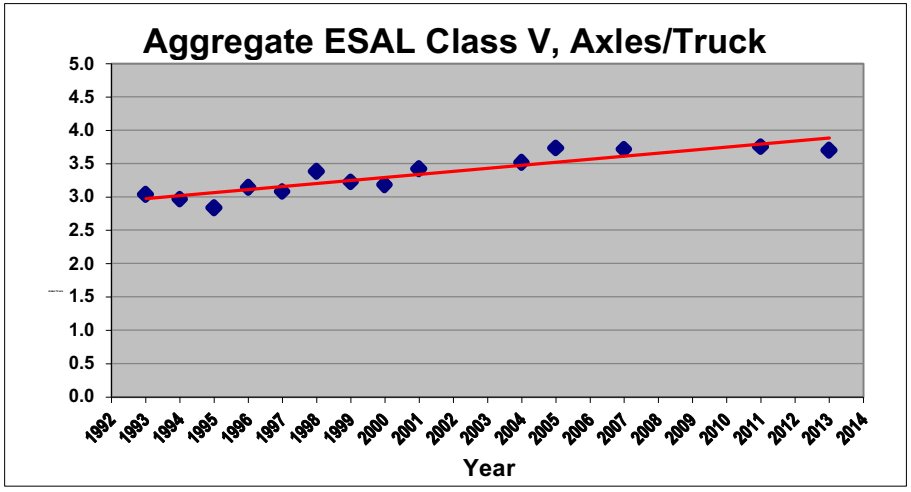


Figure 6 Aggregate ESAL Class V (FC 12, 14), 2013 -- 2 sites

Year	AADT	% Trucks	Axles/Truck	EAL's/Axle	EAL's/veh
1993	19,626	6.31	3.041	0.196	0.60
1994	16,519	7.74	2.972	0.214	0.64
1995	16,319	5.07	2.838	0.211	0.60
1996	21,135	7.37	3.144	0.204	0.64
1997	26,925	5.88	3.083	0.213	0.66
1998	20,421	9.96	3.386	0.243	0.82
1999	15,462	8.27	3.228	0.257	0.83
2000	25,643	9.08	3.186	0.273	0.87
2001	26,231	9.51	3.427	0.258	0.88
2004	19,655	8.88	3.519	0.304	1.07
2005	19,072	10.49	3.735	0.321	1.20
2007	18,940	9.62	3.718	0.312	1.16
2011	18,568	9.46	3.757	0.307	1.15
2013	17,346	9.09	3.704	0.293	1.09





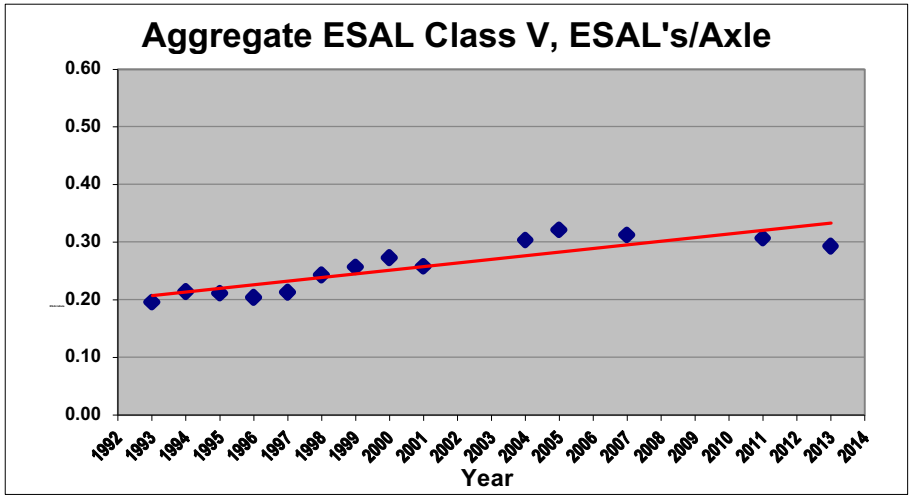
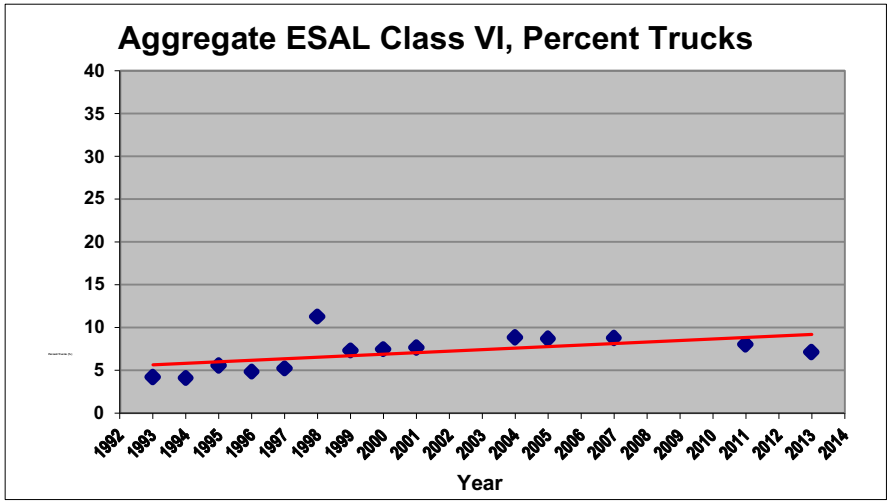
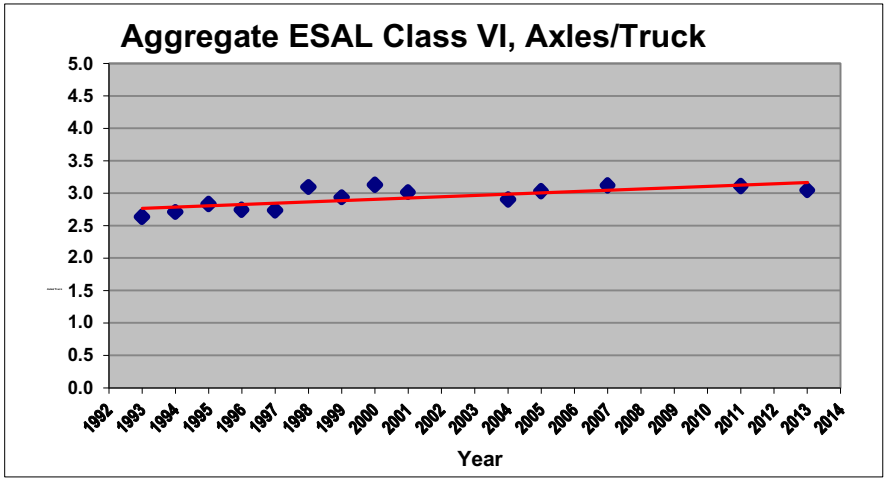
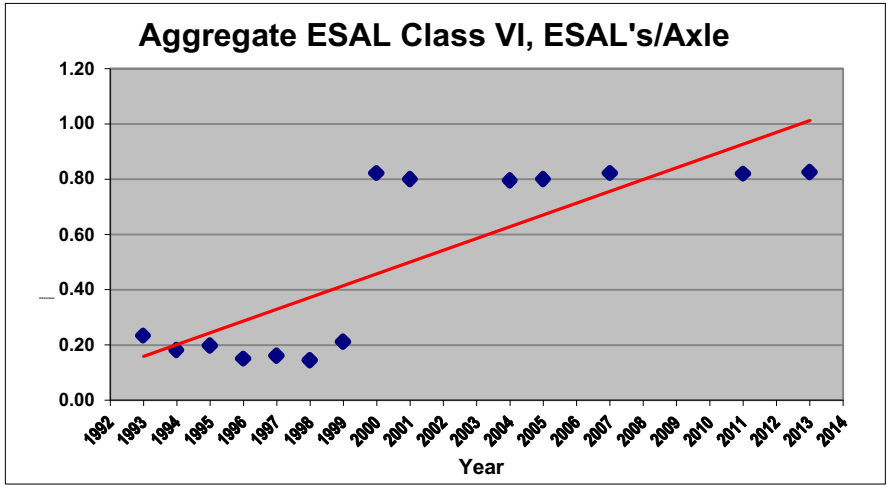


Figure 7 Aggregate ESAL Class VI (FC 16, 17, 19), 2013 -- 0 sites

Year	AADT	% Trucks	Axles/Truck	EAL's/Axle	EAL's/veh
1993	11,811	4.22	2.635	0.233	0.61
1994	12,275	4.11	2.711	0.181	0.49
1995	10,381	5.56	2.828	0.197	0.56
1996	11,221	4.85	2.741	0.150	0.41
1997	13,803	5.23	2.731	0.160	0.44
1998	10,994	11.28	3.095	0.145	0.45
1999	8,809	7.29	2.933	0.211	0.62
2000	11,235	7.46	3.125	0.821	2.57
2001	10,619	7.63	3.016	0.800	2.41
2004	6,908	8.85	2.901	0.795	2.31
2005	6,745	8.71	3.028	0.799	2.42
2007	7,587	8.76	3.117	0.821	2.56
2011	7,363	8.00	3.107	0.819	2.54
2013	7,259	7.14	3.043	0.825	2.51







4.0 HISTORICAL TRUCK VOLUME TRENDS

4.0 Truck Volume Summary

The research team used historical classification data from the development of ESAL factors from 1994 to 2013 to evaluate trends in truck volume. Using raw classification counts as a starting point, the ESAL processing algorithm extrapolates an equivalent 24-hr traffic count. These counts are then adjusted based on the season data were collected in to establish an equivalent annual count and the associated truck percentage. For this analysis the truck percentage and the station AADT were included in the classification count to obtain the truck volume.

The resultant truck volumes and truck percentages were averaged by functional class and aggregate ESAL class, for each year from 1994 to 2013, with the exceptions Aggregate ESAL Classes I and IV, which are the same as and are represented as Functional Class 1 and 11. Figures 7–22 contain these results.

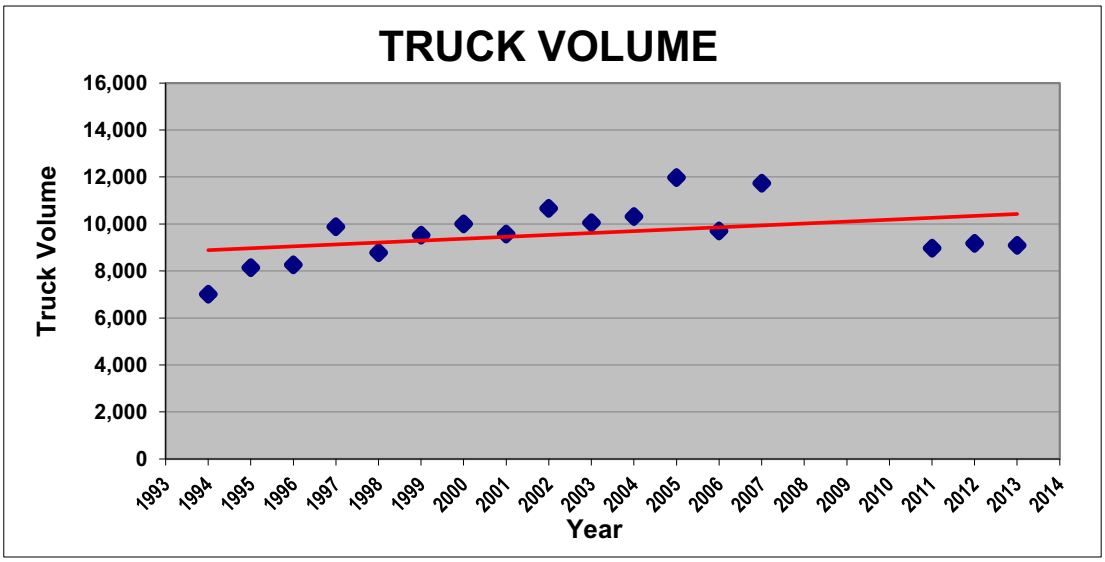
Using the WIM analysis, linear regression was used to calculate the growth rate; the slope of a linear regression line through the data represents growth. This slope was divided by the predicted value for 2013 to derive growth rate. The same procedure was adopted in KTC-01-15, "Analysis of Traffic Growth Rates".

Some of the classes reveal significant scatter in the data. Sampling issues may be the primary cause of this, given that a different number of class counts may be taken from year to year. To determine the sample size needed to obtain a stable mean value for a specific group of stations, 2004 data were used to determine a cumulative mean value.

A mean traffic count from random samples with increasing sample size was calculated for each data grouping and plotted versus the number of samples collected. Figures 23–27 summarize the results of this analysis. Functional Class 19 was not included in this analysis because of the small number of data collection sites available. These figures may be used to evaluate whether historical data contain a sufficient number of samples to ensure a stable mean value. They may also be used to decide how many samples should be collected during future efforts. In general, fewer samples are required to obtain a stable mean truck percentage than to obtain a stable mean total truck volume.

Figure 8 Functional Class 01

Year	Sites	AADT	Trucks	% Trucks
1994	11	27,755	7,010	25.48
1995	12	26,008	8,135	31.23
1996	10	28,780	8,257	28.71
1997	20	34,777	9,879	28.97
1998	17	23,935	8,776	34.15
1999	9	26,600	9,518	35.60
2000	10	32,630	10,002	31.28
2001	22	30,877	9,572	30.67
2002	22	32,686	10,669	33.14
2003	15	33,068	10,054	31.46
2004	33	36,738	10,312	29.30
2005	41	39,145	11,972	31.38
2006	19	34,976	9,703	31.33
2007	31	38,524	11,739	31.25
2011	39	34,277	8,964	26.15
2012	39	39,135	9,169	23.98
2013	30	36,767	9,085	25.20



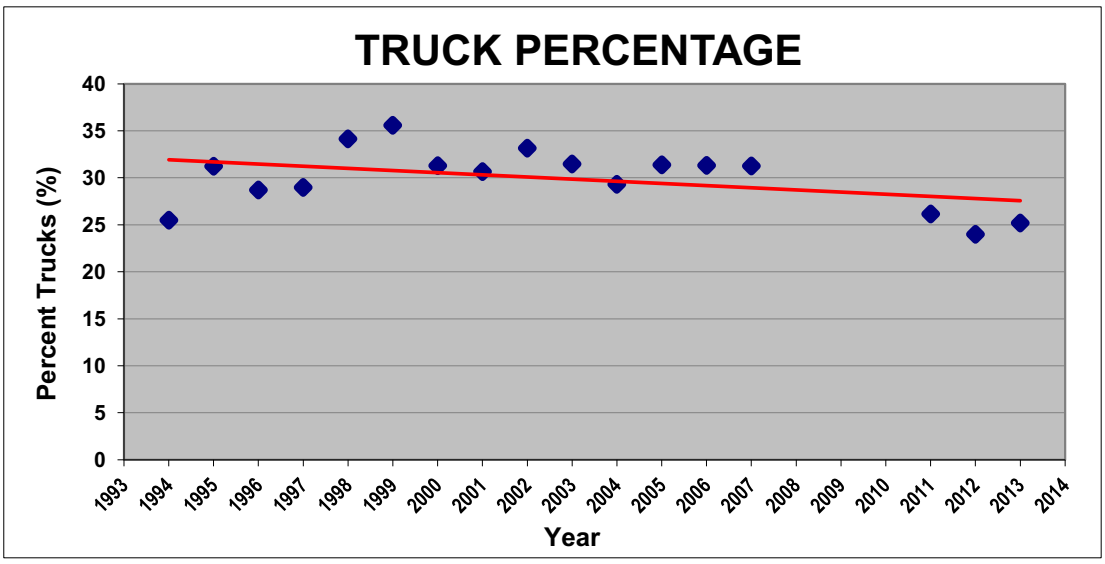
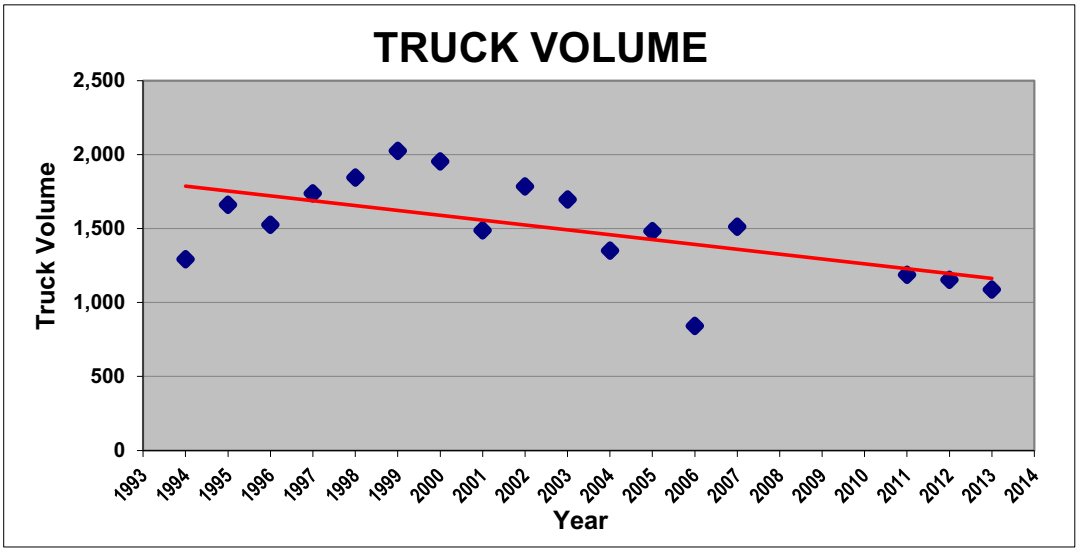


Figure 9 Functional Class 02

Year	Sites	AADT	Trucks	% Trucks
1994	42	7,924	1,292	15.41
1995	46	8,338	1,660	17.86
1996	53	8,561	1,526	18.60
1997	50	11,769	1,737	14.94
1998	49	9,231	1,845	18.74
1999	39	10,308	2,025	18.85
2000	22	8,292	1,954	21.67
2001	90	9,122	1,487	17.10
2002	67	9,963	1,784	19.62
2003	76	9,475	1,697	18.55
2004	119	8,955	1351	16.17
2005	146	9,235	1,483	16.43
2006	67	4,797	841	18.86
2007	125	8,785	1,513	17.96
2011	287	8,751	1,187	14.07
2012	191	8,656	1,153	13.81
2013	168	8,778	1,087	13.19



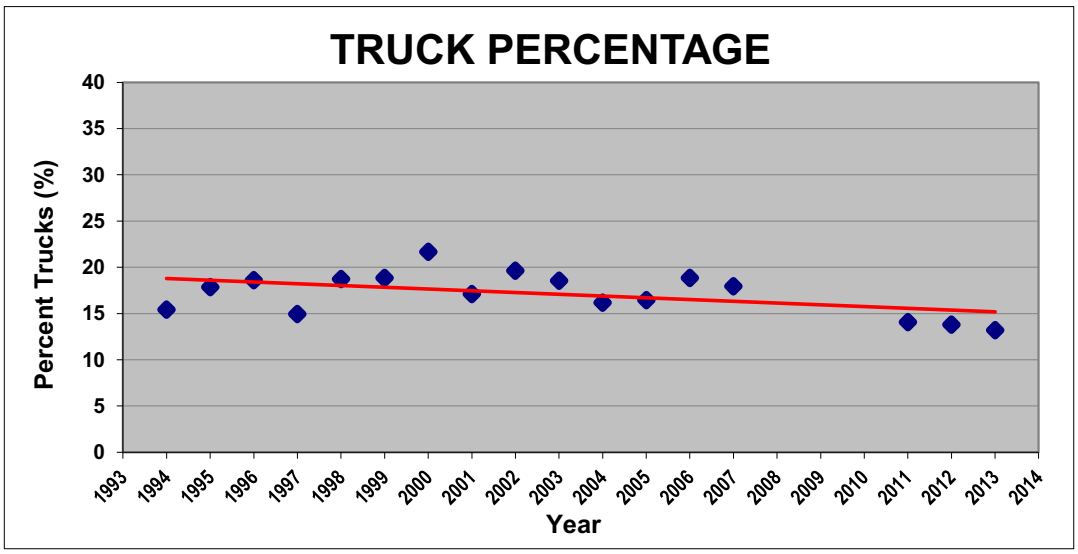
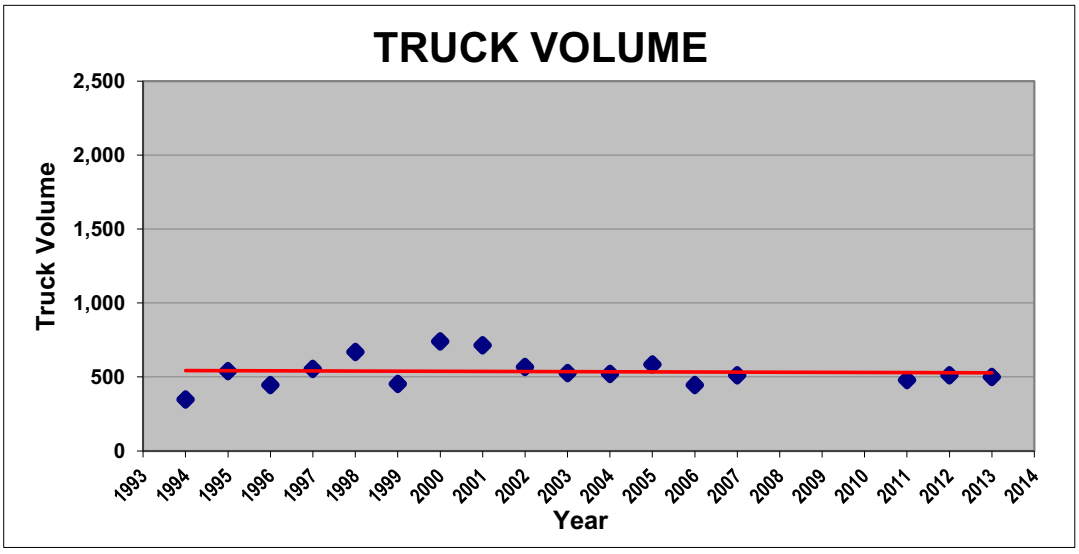


Figure 10 Functional Class 06

Year	Sites	AADT	Trucks	% Trucks
1994	49	4,345	347	8.41
1995	42	5,508	540	9.86
1996	46	5,419	446	9.27
1997	31	6,281	555	9.90
1998	42	7,567	668	10.28
1999	18	6,766	453	6.85
2000	15	5,566	740	15.19
2001	49	4,663	714	14.18
2002	29	4,446	567	14.24
2003	28	4,421	526	12.00
2004	108	5,151	520	10.96
2005	99	5,636	585	11.00
2006	44	4,423	445	11.48
2007	110	5,089	512	10.61
2011	259	5,372	480	9.82
2012	209	5,526	511	10.18
2013	200	5,616	500	9.74



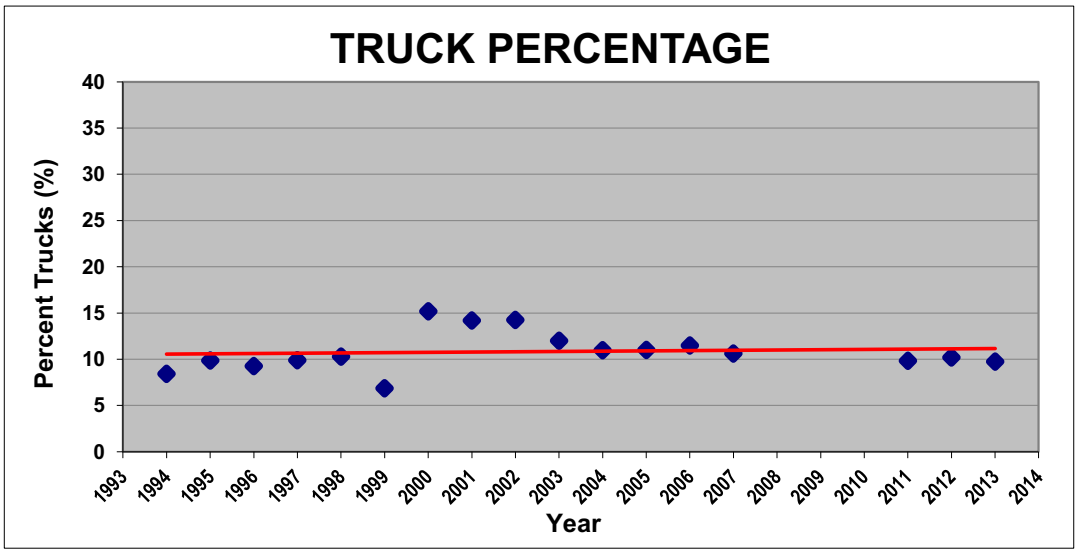
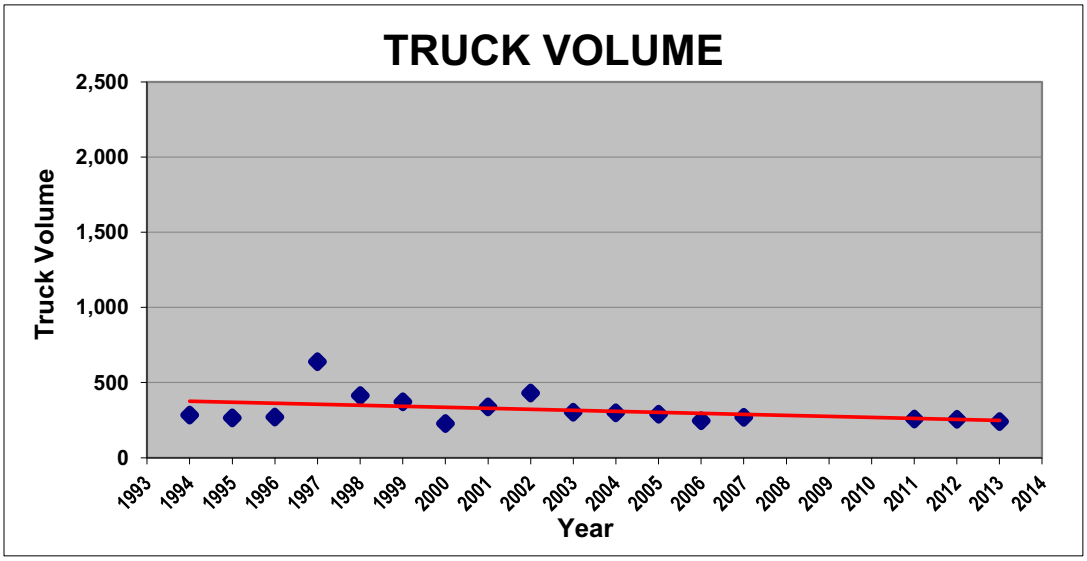


Figure 11 Functional Class 07

Year	Sites	AADT	Trucks	% Trucks
1994	75	3,145	284	9.63
1995	53	3,728	266	8.23
1996	72	3,844	271	8.12
1997	55	5,770	638	10.12
1998	78	5,032	413	10.09
1999	53	3,947	373	9.26
2000	27	2,616	227	8.65
2001	129	3,090	338	11.53
2002	76	2,959	431	13.80
2003	64	2,825	303	10.36
2004	307	3,013	298	9.85
2005	296	2,977	290	10.16
2006	112	2,598	247	9.97
2007	301	2,722	269	10.02
2011	808	2,953	257	9.02
2012	649	2,936	256	9.14
2013	529	2,910	241	8.60



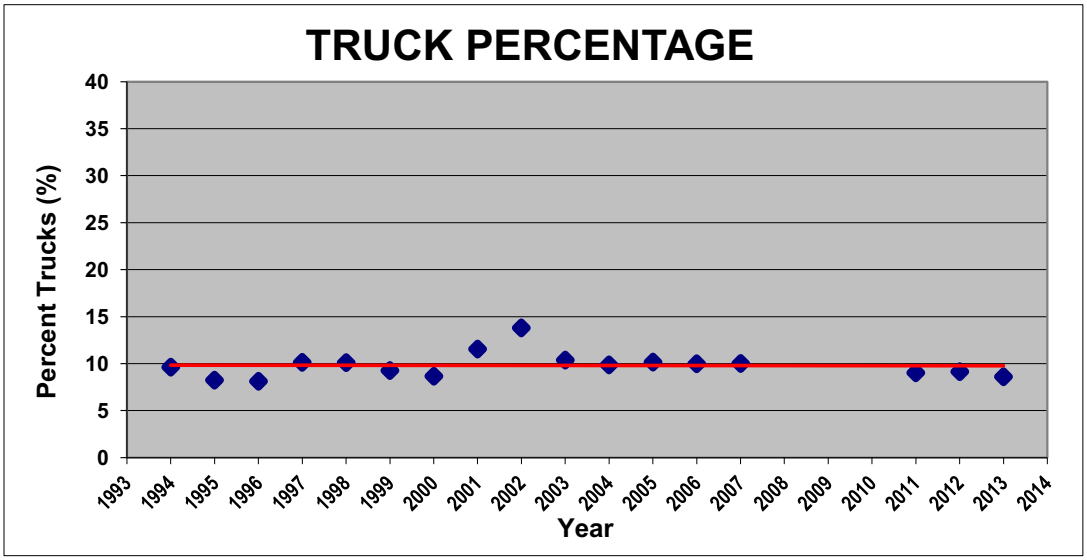
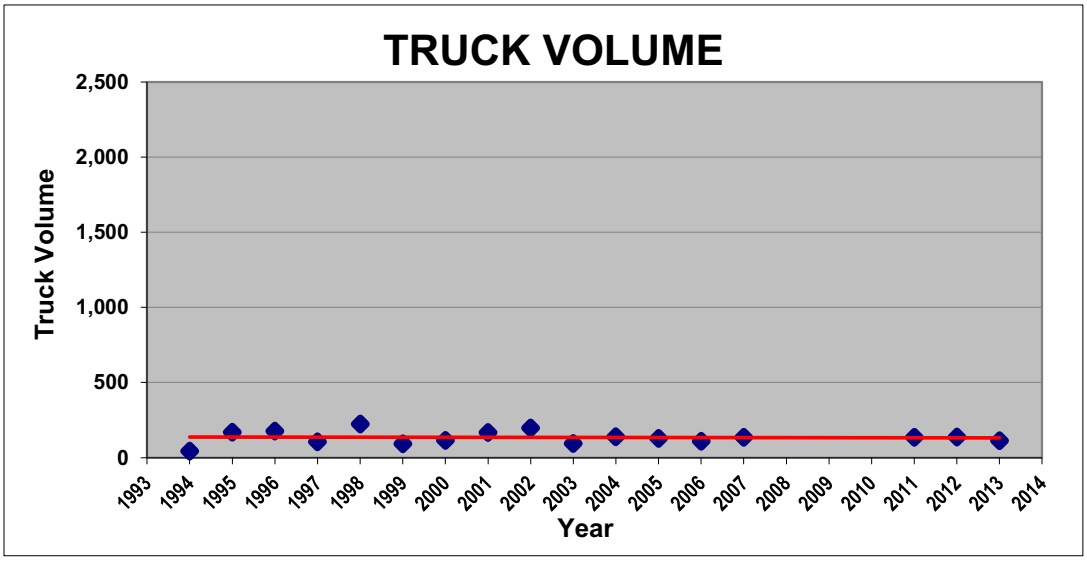


Figure 12 Functional Class 08

Year	Sites	AADT	Trucks	% Trucks
1994	9	1,067	43	4.77
1995	16	1,623	170	7.73
1996	13	1,581	177	12.94
1997	10	2,718	106	4.76
1998	13	2,181	224	11.31
1999	17	969	93	8.75
2000	10	1,237	115	9.19
2001	41	1,995	167	8.27
2002	17	1,457	198	12.38
2003	59	921	94	11.31
2004	71	1,328	140	11.14
2005	55	1,299	128	10.27
2006	22	1,270	109	9.66
2007	41	1,398	136	11.13
2011	115	1,538	136	10.09
2012	93	1,571	138	9.80
2013	92	1,305	113	9.29



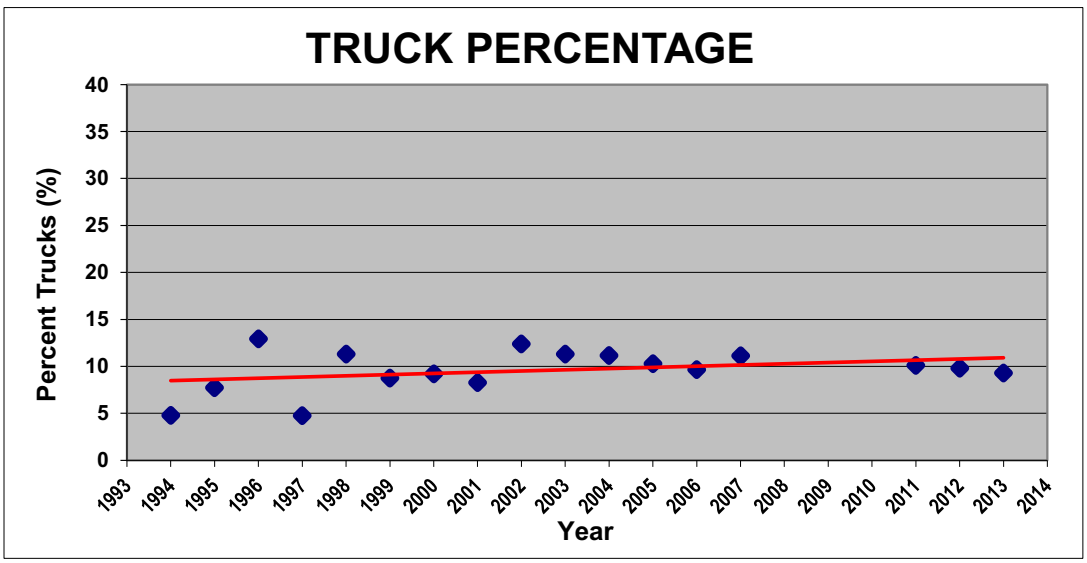
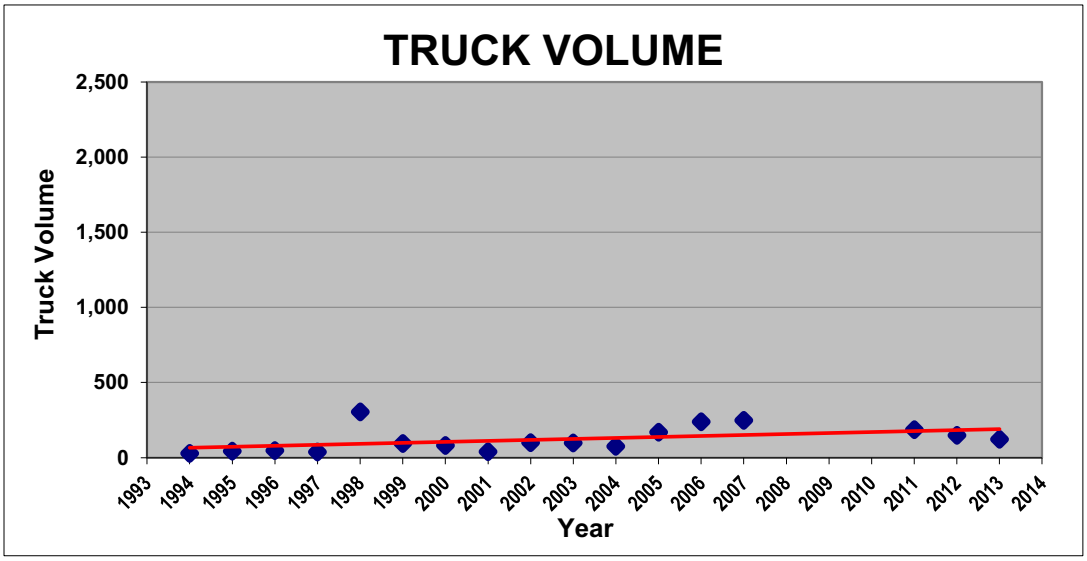


Figure 13 Functional Class 09

Year	Sites	AADT	Trucks	% Trucks
1994	1	915	28	3.03
1995	1	950	44	4.60
1996	1	1,000	48	4.80
1997	2	1,151	37	3.28
1998	13	2,472	305	14.26
1999	1	2,470	94	3.80
2000	2	563	81	13.93
2001	10	424	39	14.84
2002	5	951	99	10.67
2003	26	833	97	13.65
2004	27	674	76	11.51
2005	12	1,701	169	10.84
2006	2	1,785	239	14.07
2007	8	2,176	249	8.02
2011	21	1,771	186	10.38
2012	21	1,641	148	10.31
2013	15	2,063	123	9.60



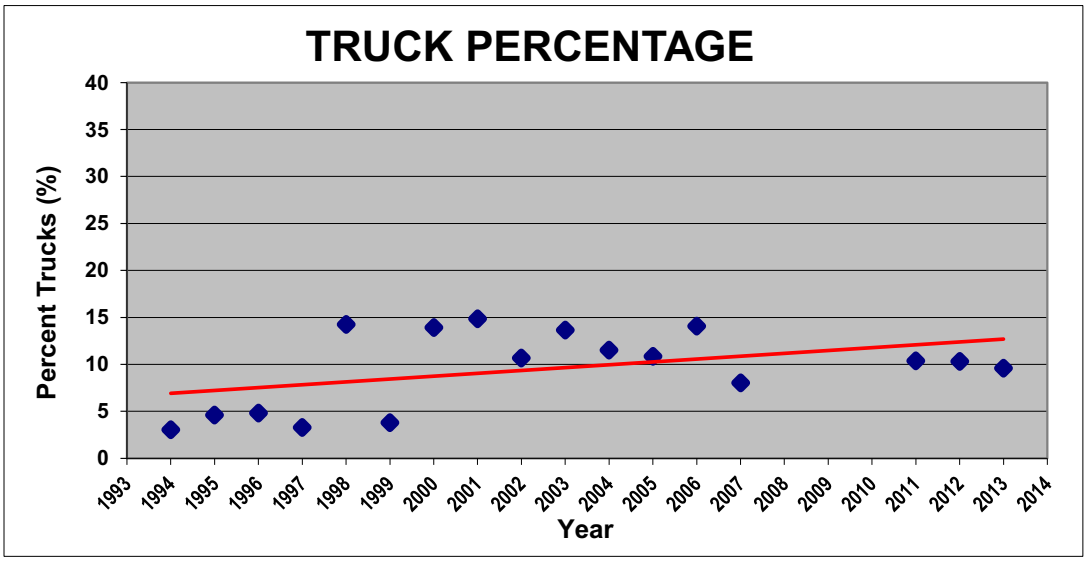
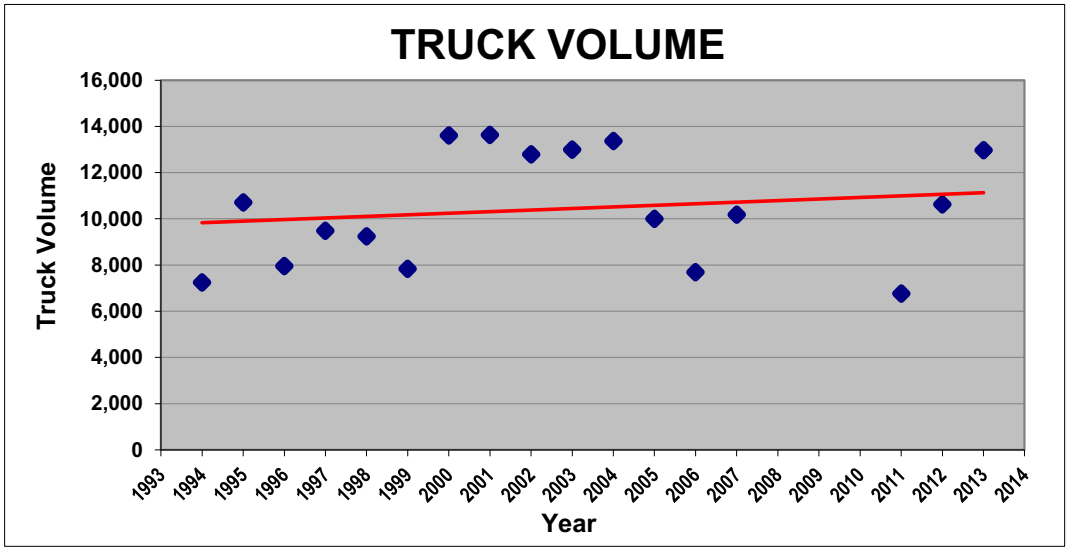


Figure 14 Functional Class 11

Year	Sites	AADT	Trucks	% Trucks
1994	16	64,413	7,246	11.73
1995	11	99,691	10,714	11.39
1996	18	68,578	7,959	12.06
1997	17	66,586	9,478	15.71
1998	3	35,800	9,244	24.52
1999	2	28,000	7,835	29.14
2000	1	39,600	13,604	34.35
2001	26	81,454	13,626	17.75
2002	35	80,003	12,783	16.63
2003	17	84,597	12,989	17.88
2004	40	74,535	13,370	19.73
2005	32	68,696	10,001	15.66
2006	20	63,113	7,687	15.73
2007	36	67,185	10,175	18.41
2011	82	45,498	6,763	14.87
2012	47	86,843	10,621	13.64
2013	28	96,514	12,969	14.83



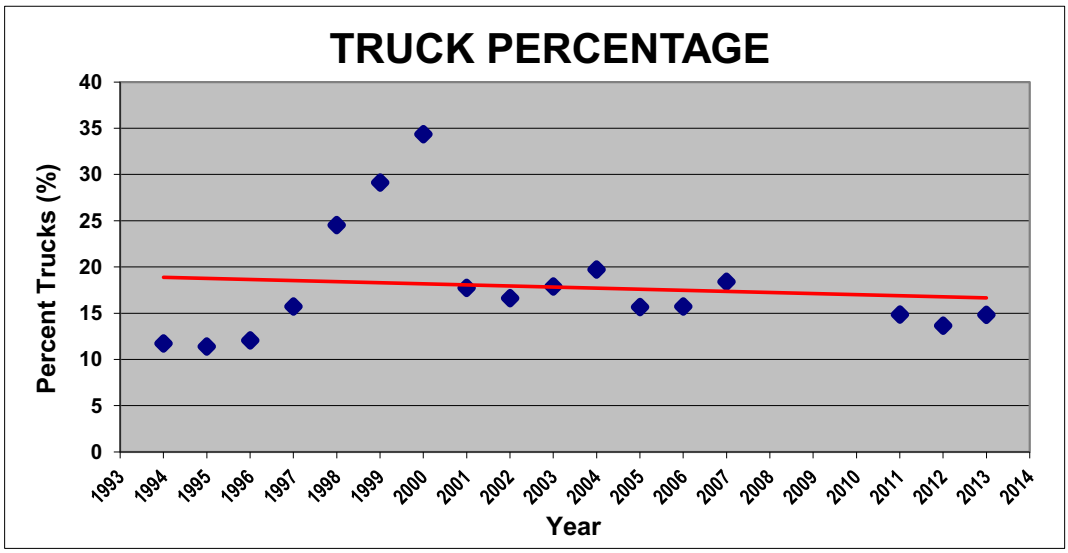
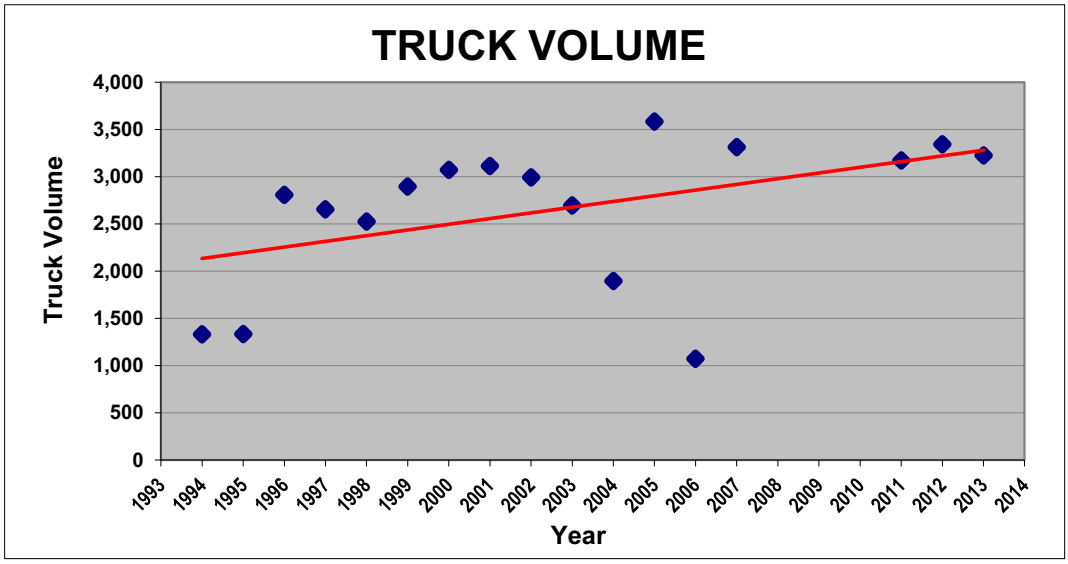


Figure 15 Functional Class 12

Year	Sites	AADT	Trucks	% Trucks
1994	1	9,360	1,331	14.03
1995	5	16,480	1,333	8.99
1996	11	26,128	2,808	13.85
1997	7	37,716	2,654	8.72
1998	10	20,577	2,525	17.43
1999	4	27,705	2,895	18.21
2000	7	40,580	3,072	12.11
2001	10	35,670	3,113	11.22
2002	9	25,988	2,992	18.21
2003	15	27,156	2,696	13.70
2004	49	21,857	1,894	10.83
2005	12	28,205	3,583	18.15
2006	19	13,930	1,071	13.34
2007	6	21,087	3,313	19.60
2011	17	34,141	3,172	11.98
2012	8	34,208	3,341	12.08
2013	9	30,972	3,223	12.64



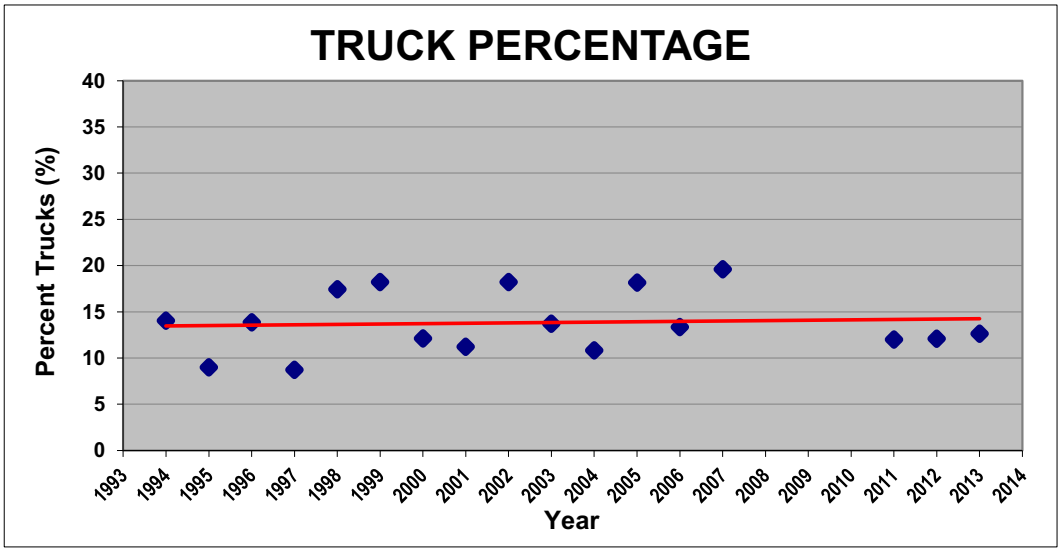
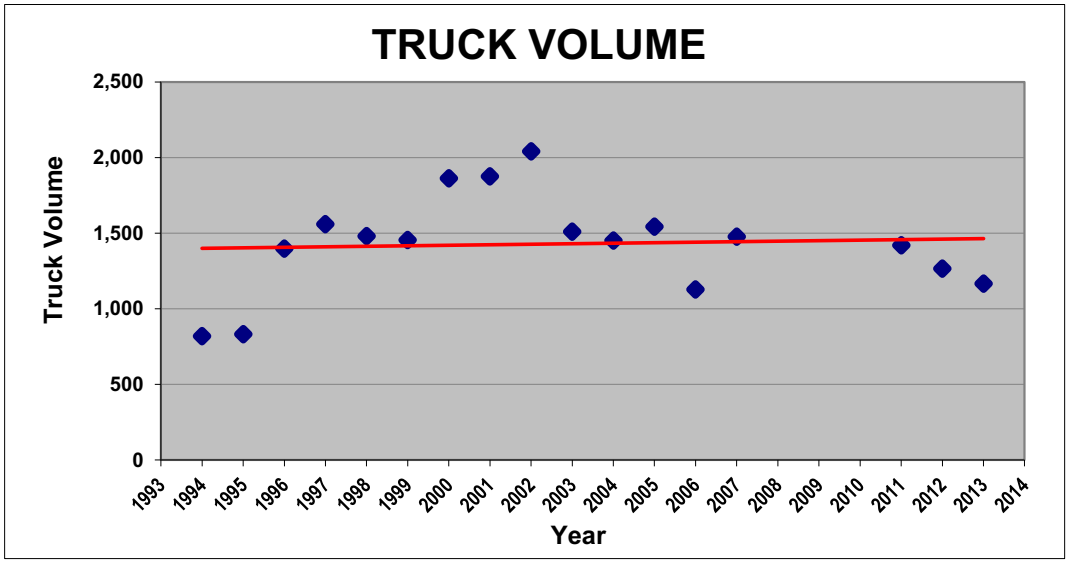


Figure 16 Functional Class 14

Year	Sites	AADT	Trucks	% Trucks
1994	50	17,093	819	5.24
1995	35	16,697	831	5.36
1996	46	21,441	1,398	6.65
1997	46	25,106	1,559	6.57
1998	33	20,999	1,480	7.81
1999	21	18,510	1,455	8.15
2000	18	24,177	1,863	8.68
2001	37	24,367	1,876	8.77
2002	52	23,891	2,041	9.84
2003	24	25,097	1,509	6.73
2004	109	20,266	1,450	8.10
2005	87	17,985	1,542	9.45
2006	51	15,828	1,127	9.51
2007	99	18,934	1,477	9.08
2011	275	18,796	1,419	8.29
2012	200	17,844	1,266	8.02
2013	125	17,274	1,166	7.89



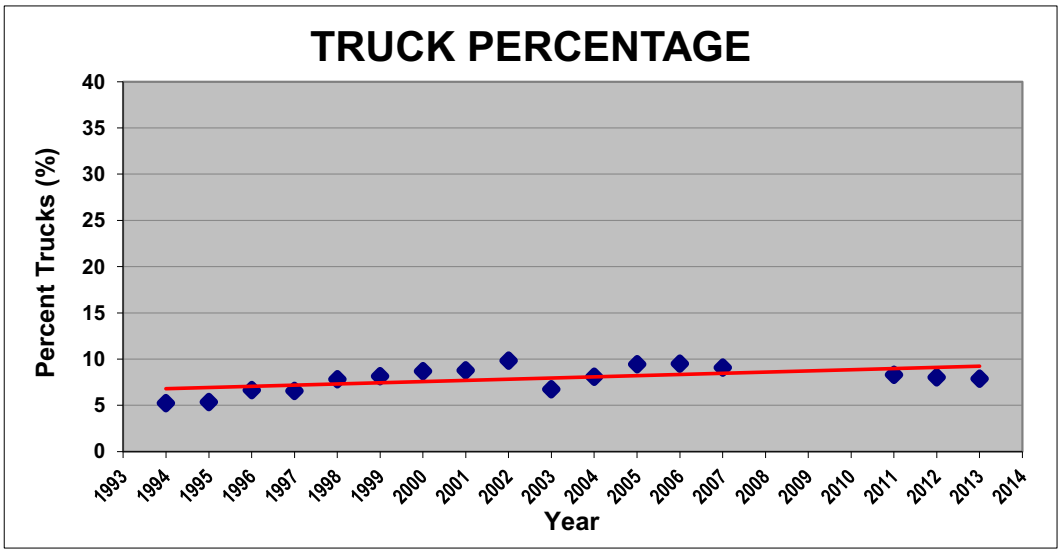
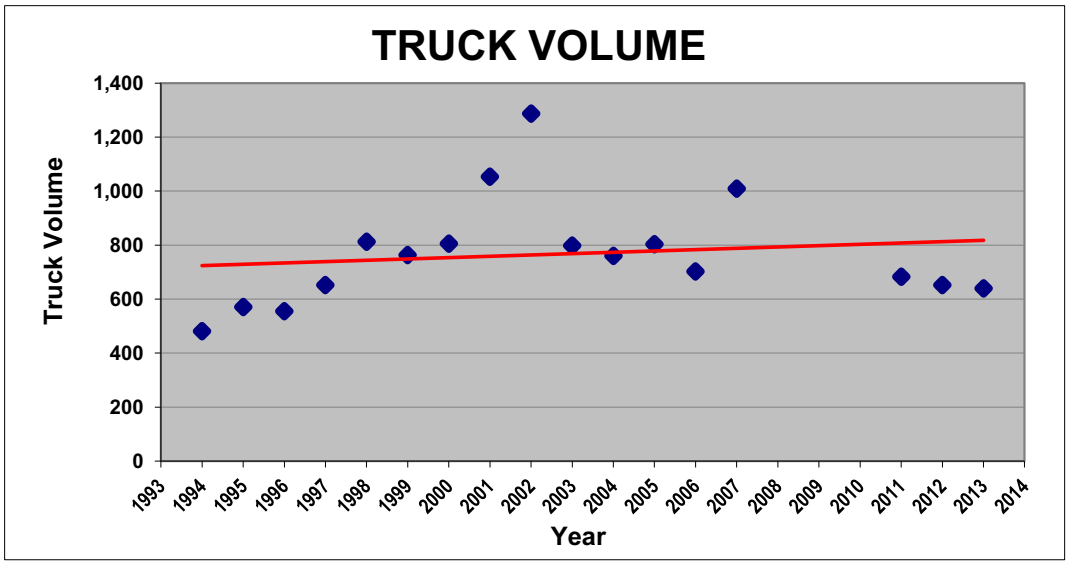


Figure 17 Functional Class 16

Year	Sites	AADT	Trucks	% Trucks
1994	31	14,035	481	4.29
1995	33	11,331	571	5.45
1996	29	12,576	555	5.17
1997	26	16,596	652	4.74
1998	23	14,895	813	5.34
1999	12	10,293	763	6.40
2000	14	12,416	805	7.59
2001	20	11,733	1,053	9.07
2002	24	13,216	1,287	10.23
2003	17	9,585	798	8.94
2004	117	9,188	760	8.72
2005	103	8,258	803	9.14
2006	58	8,972	702	7.88
2007	108	9,704	1,009	9.74
2011	320	9,090	683	7.77
2012	233	9,426	652	7.49
2013	192	9,577	640	7.08



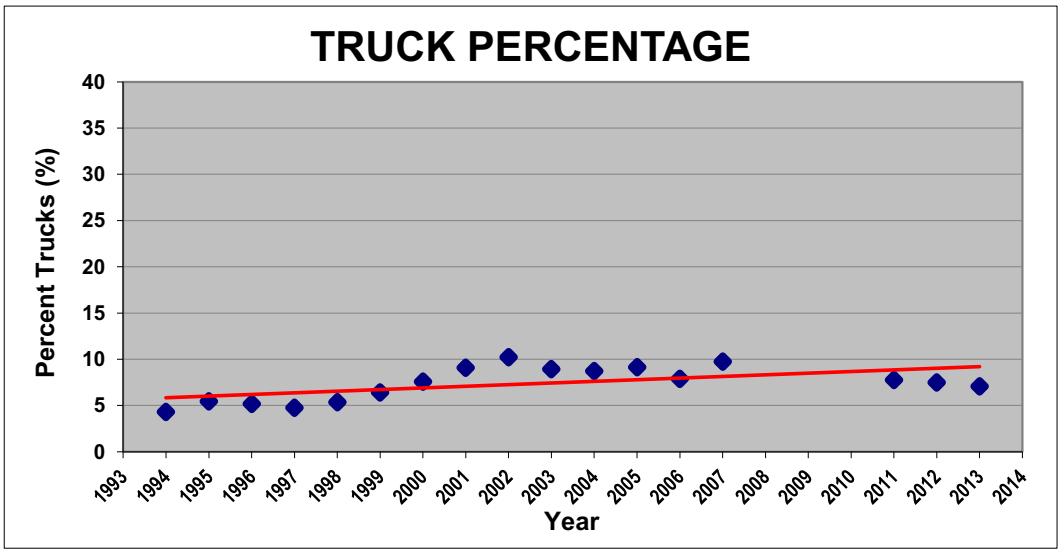
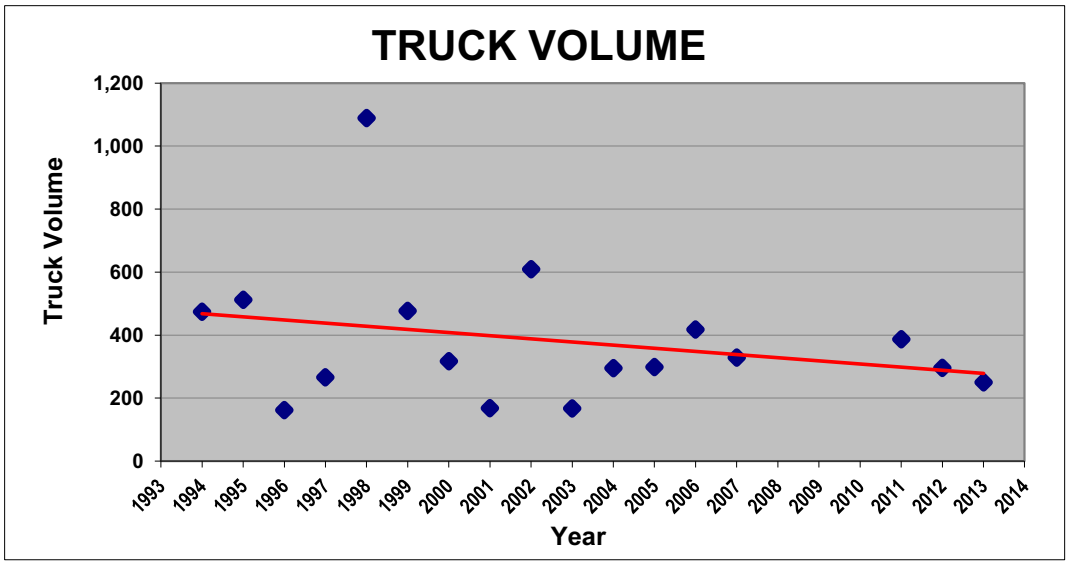


Figure 18 Functional Class 17

Year	Sites	AADT	Trucks	% Trucks
1994	4	11,715	474	5.18
1995	10	9,142	512	6.76
1996	8	7,128	162	2.14
1997	9	7,376	266	3.73
1998	10	7,381	1,089	14.69
1999	3	6,653	477	8.93
2000	3	5,730	317	6.89
2001	9	5,069	168	3.91
2002	15	9,243	609	8.72
2003	7	2,867	167	6.23
2004	79	3,787	295	7.76
2005	50	3,703	299	8.39
2006	26	5,329	418	7.98
2007	68	4,510	329	7.65
2011	158	4,941	387	6.08
2012	128	4,882	296	5.53
2013	118	4,592	250	5.34



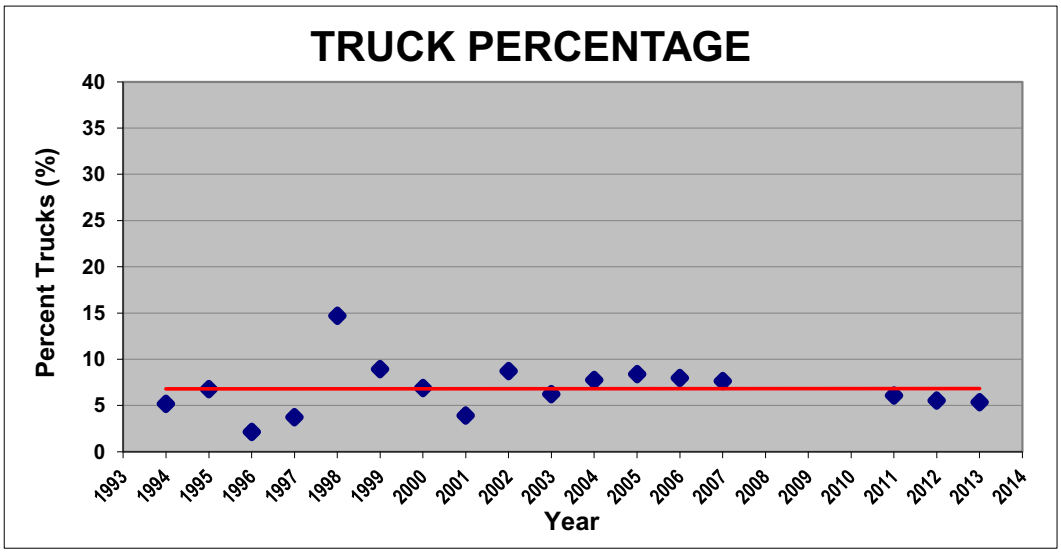
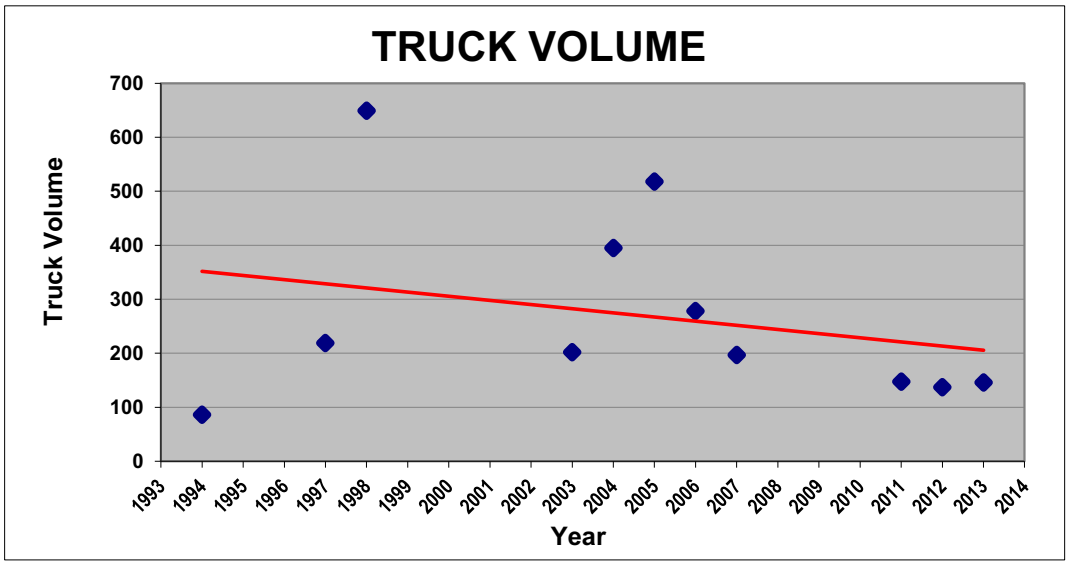


Figure 19 Functional Class 19

Year	Sites	AADT	Trucks	% Trucks
1994	1	1,740	86	4.93
1997	1	656	219	33.44
1998	8	3,180	649	25.71
2003	1	1,436	202	14.10
2004	4	4,002	395	9.23
2005	5	6,295	518	6.25
2006	2	5,384	278	4.44
2007	2	3,122	197	8.14
2011	12	3,495	147	4.22
2012	7	2,848	137	4.76
2013	4	3,300	146	4.47



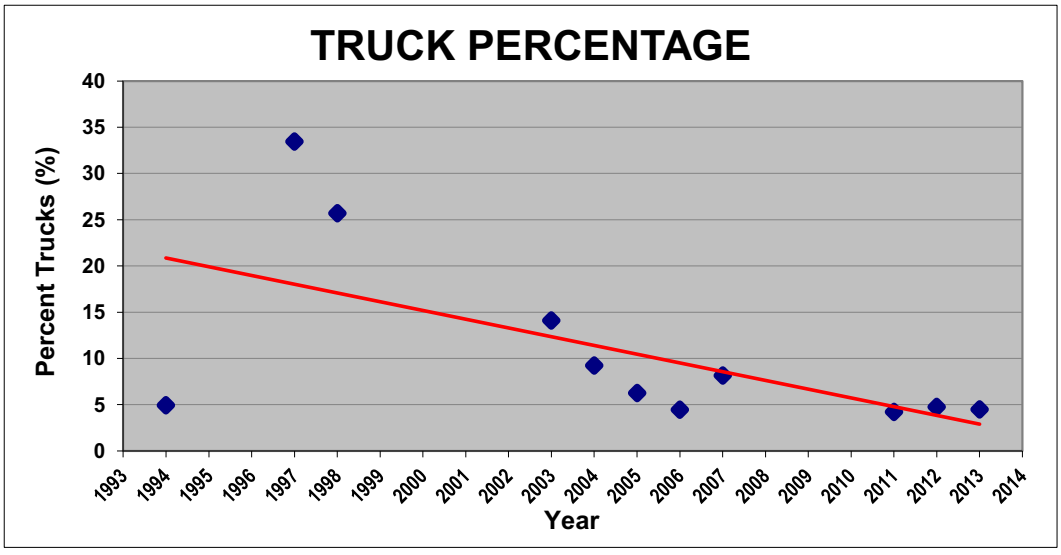
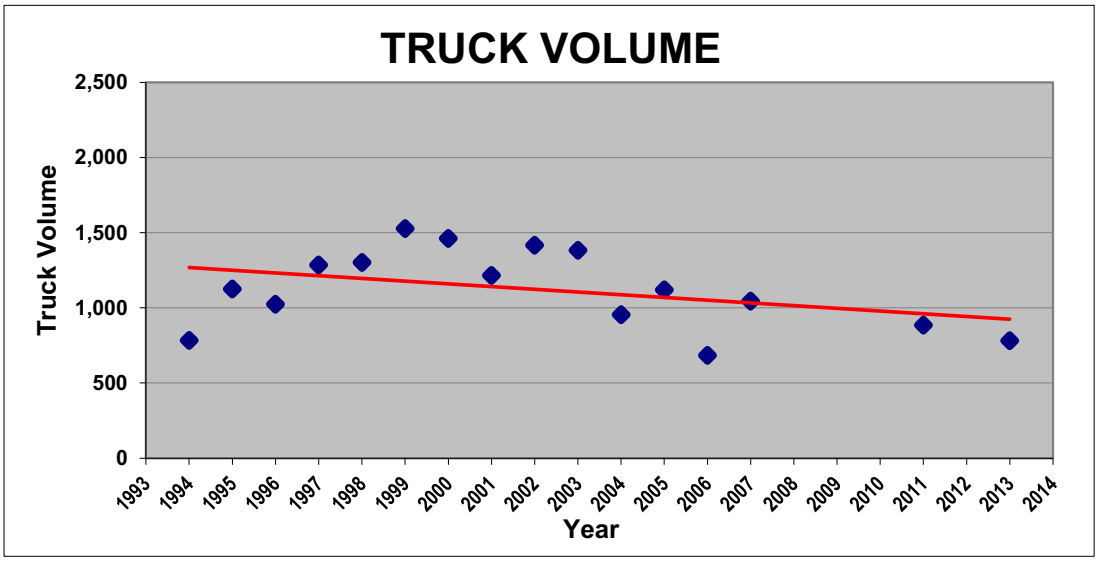


Figure 20 Aggregate ESAL Class II (FC 02, 06)

Year	Sites	AADT	Trucks	% Trucks
1994	91	5,997	783	11.64
1995	88	6,987	1,126	14.04
1996	99	7,101	1,024	14.26
1997	81	9,669	1,285	13.01
1998	91	8,463	1,302	14.83
1999	57	9,189	1,528	15.06
2000	37	7,187	1,462	19.05
2001	139	7,550	1,215	16.07
2002	96	8,296	1,417	17.99
2003	104	8,114	1,382	16.78
2004	227	7,145	955	13.69
2005	245	7,781	1,120	14.23
2006	111	4,649	684	15.93
2007	235	7,055	1,044	14.52
2011	546	6,688	886	13.24
2013	388	6,262	781	12.48



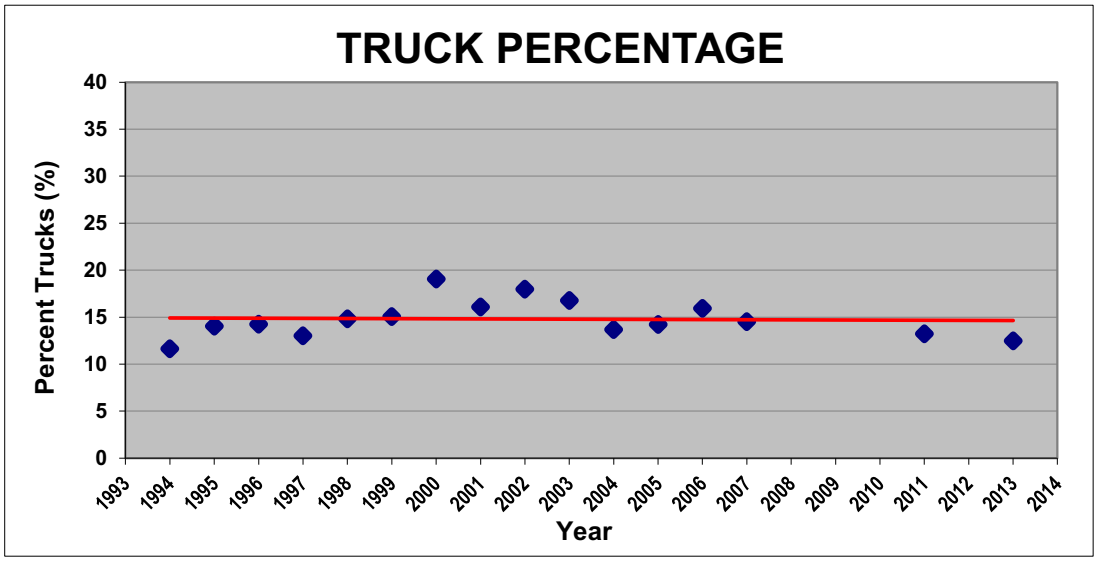
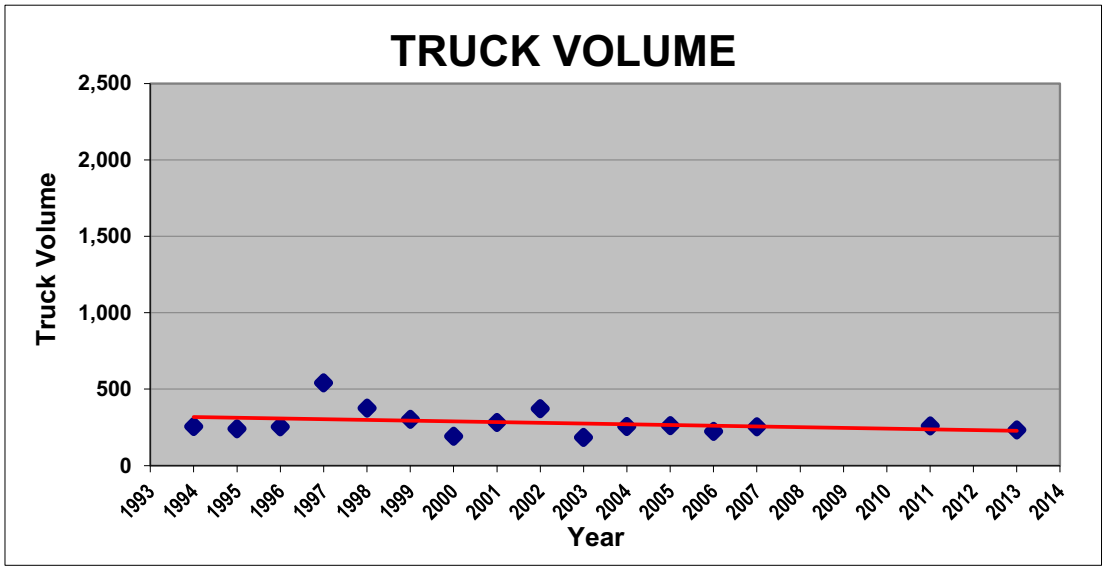


Figure 21 Aggregate ESAL Class III (FC 07, 08, 09)

Year	Sites	AADT	Trucks	% Trucks
1994	85	2,899	255	9.04
1995	70	3,207	241	8.06
1996	86	3,468	254	8.81
1997	67	5,177	541	9.12
1998	104	4,356	376	10.77
1999	71	3,213	302	9.06
2000	39	2,157	191	9.06
2001	180	2,692	282	10.97
2002	98	2,596	373	13.40
2003	149	1,723	184	11.31
2004	405	2,561	256	10.19
2005	363	2,680	262	10.20
2006	136	2,371	224	9.98
2007	350	2,555	253	10.11
2011	944	2,510	259	10.31
2013	636	2,418	233	9.63



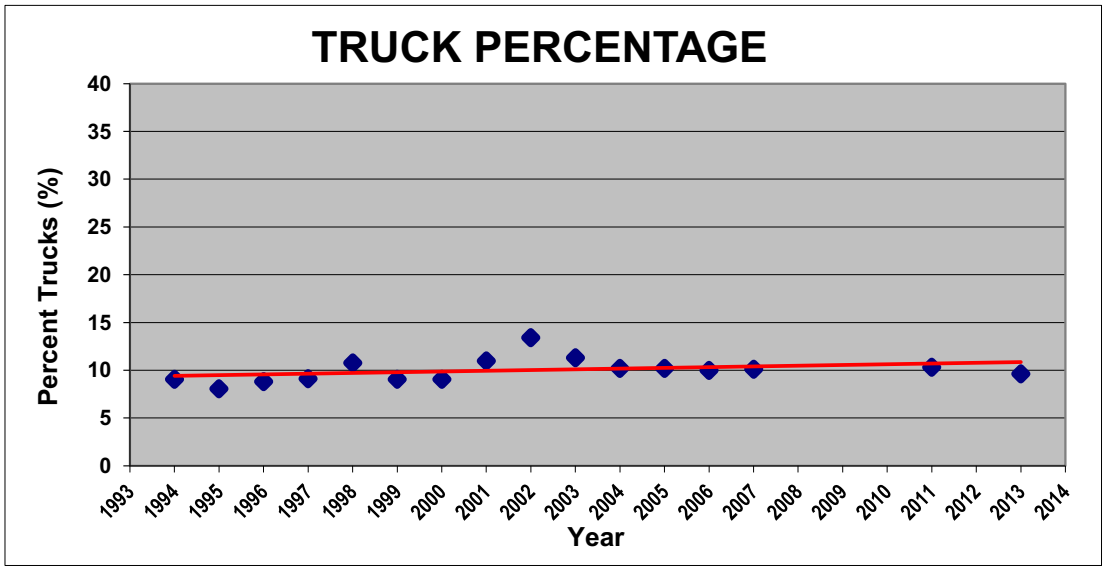
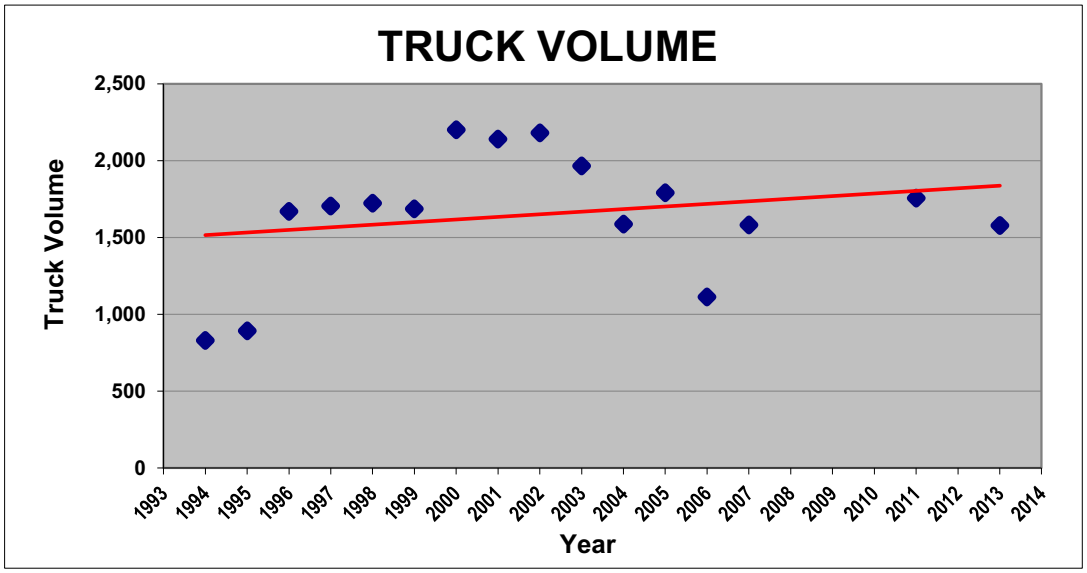


Figure 22 Aggregate ESAL Class V (FC 12, 14)

Year	Sites	AADT	Trucks	% Trucks
1994	51	16,941	829	5.42
1995	40	16,670	893	5.81
1996	57	22,345	1,670	8.04
1997	53	26,772	1,704	6.86
1998	43	20,901	1,723	10.05
1999	25	19,981	1,686	9.76
2000	25	28,770	2,201	9.64
2001	47	26,772	2,140	9.29
2002	61	24,200	2,181	11.08
2003	39	25,889	1,965	9.41
2004	158	20,759	1,587	8.95
2005	99	19,224	1,790	10.51
2006	70	15,313	1,112	10.55
2007	105	19,057	1,582	9.68
2011	291	18,568	1,757	9.46
2013	133	17,346	1,577	9.09



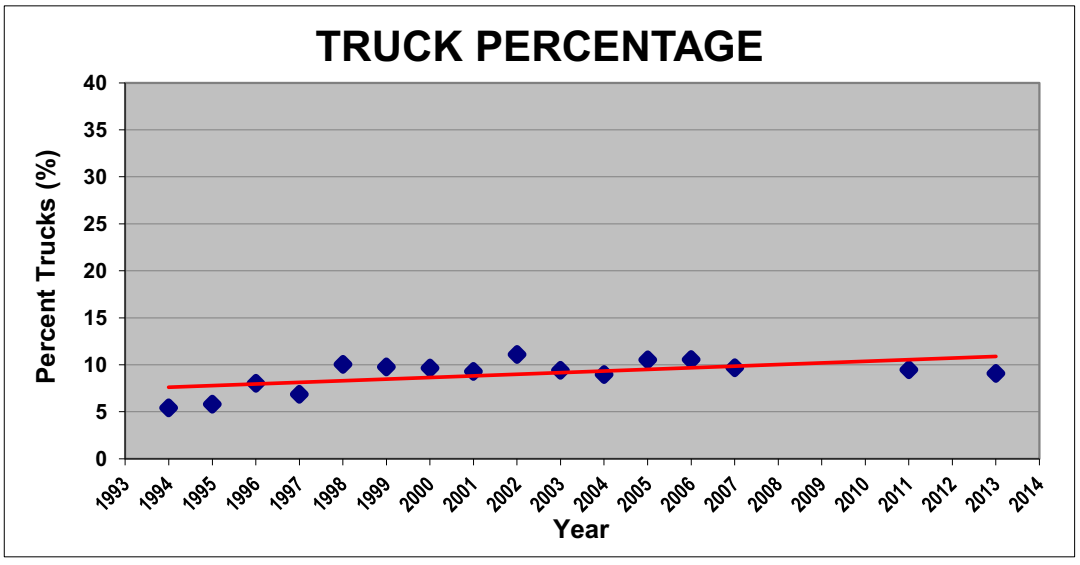
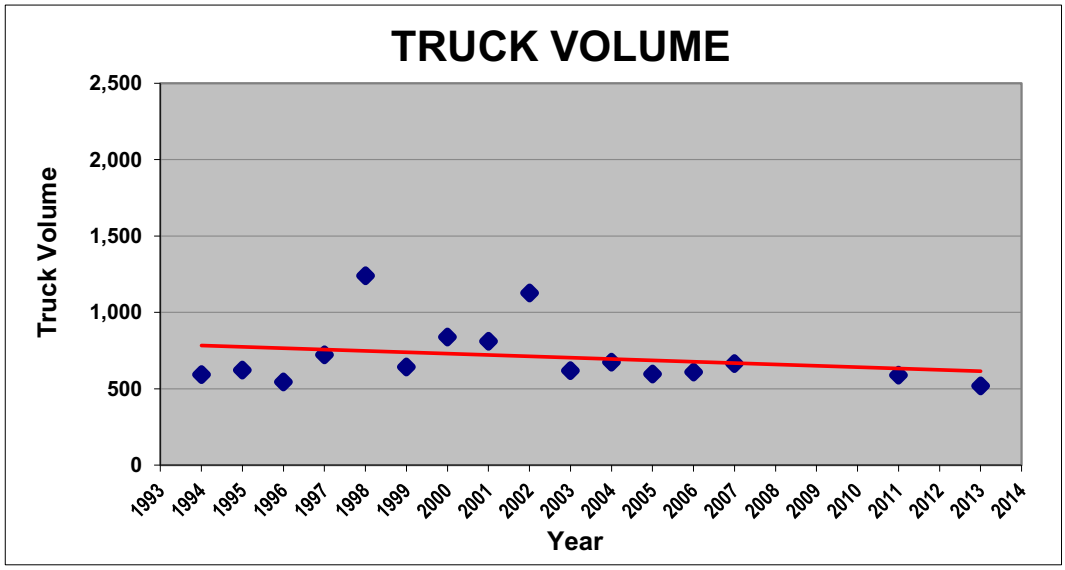
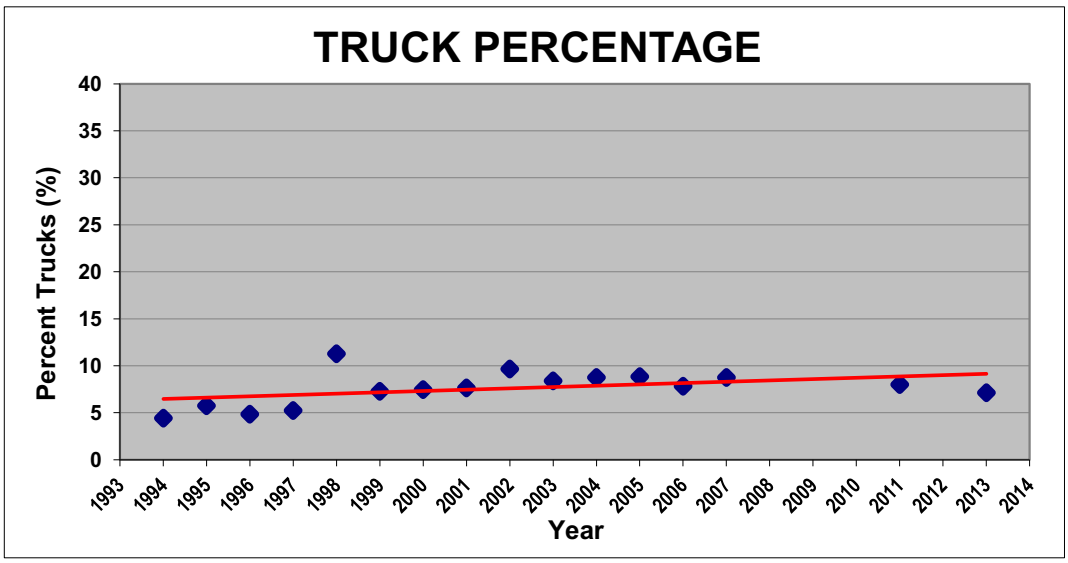


Figure 23 Aggregate ESAL Class VI (FC 16, 17, 19)

Year	Sites	AADT	Trucks	% Trucks
1994	36	13,435	592	4.41
1995	43	10,822	622	5.75
1996	39	11,221	544	4.85
1997	36	13,803	722	5.23
1998	32	10,994	1240	11.28
1999	17	8,809	642	7.29
2000	17	11,235	838	7.46
2001	32	10,619	810	7.63
2002	39	11,688	1127	9.64
2003	25	7,378	619	8.39
2004	222	7,713	675	8.75
2005	159	6,745	596	8.84
2006	86	7,787	610	7.83
2007	177	7,587	665	8.76
2011	489	7,363	589	8.00
2013	314	7,259	518	7.14





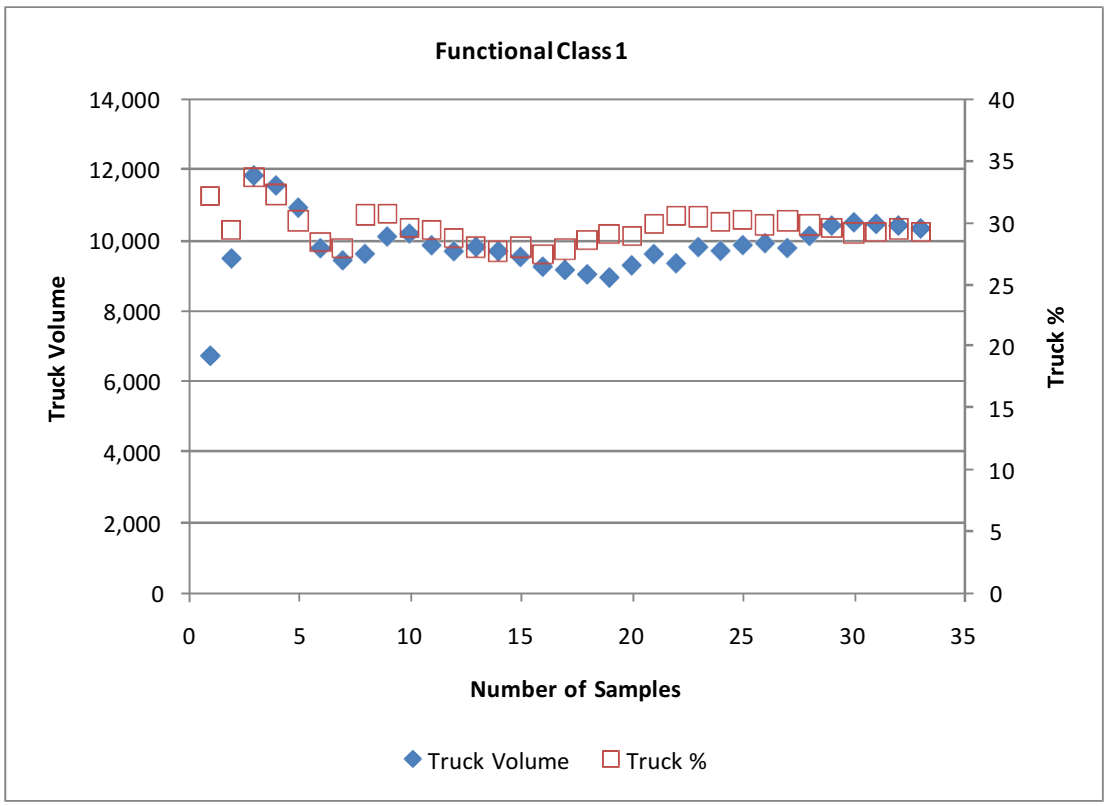


Figure 24 Functional Class 01, Sample Size

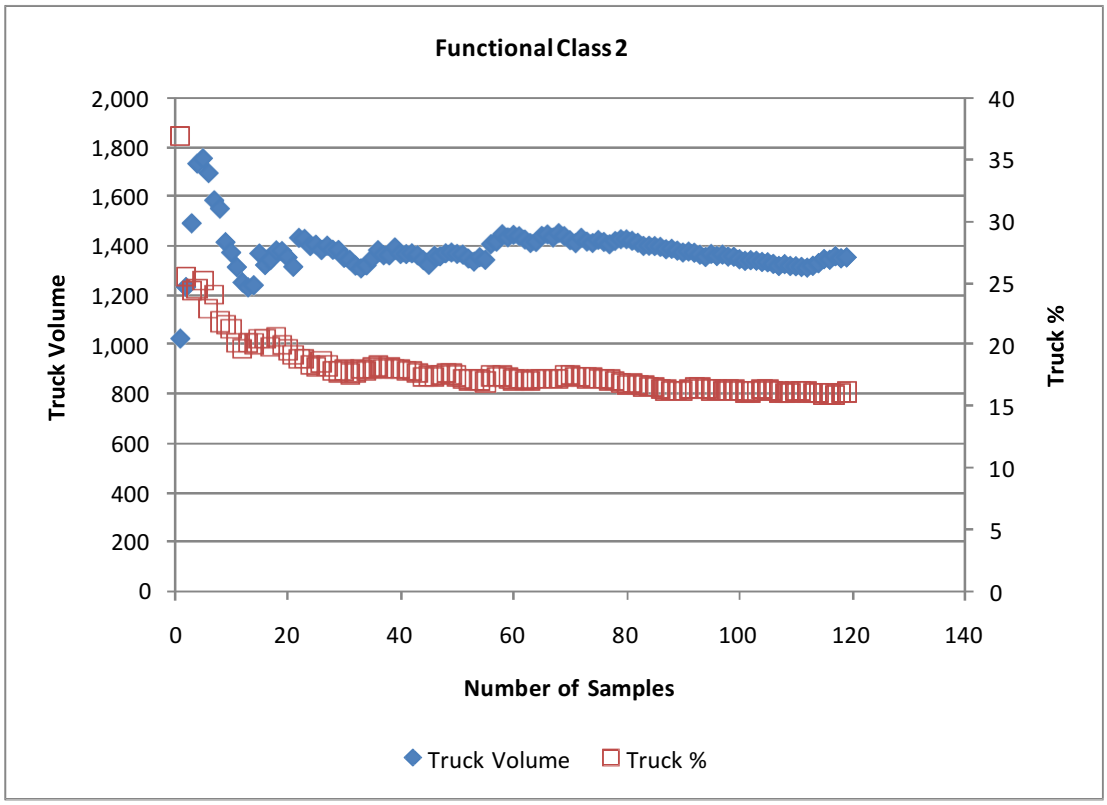


Figure 25 Functional Class 02, Sample Size

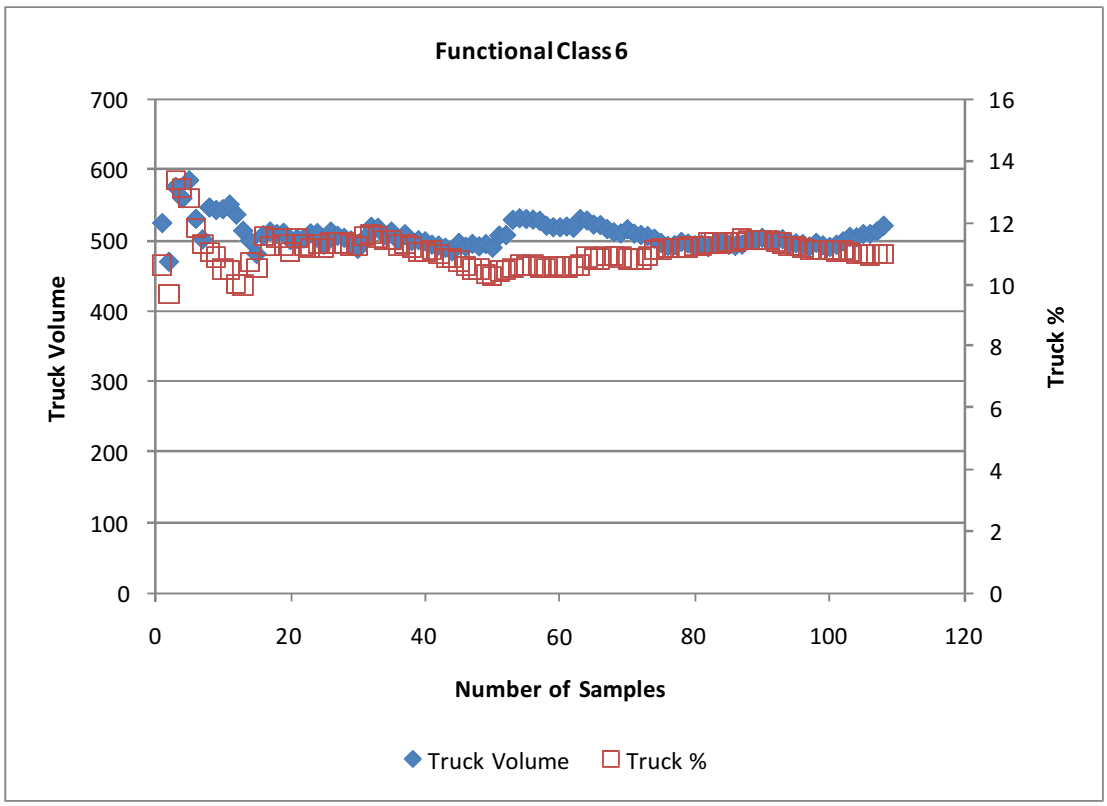


Figure 26 Functional Class 06, Sample Size

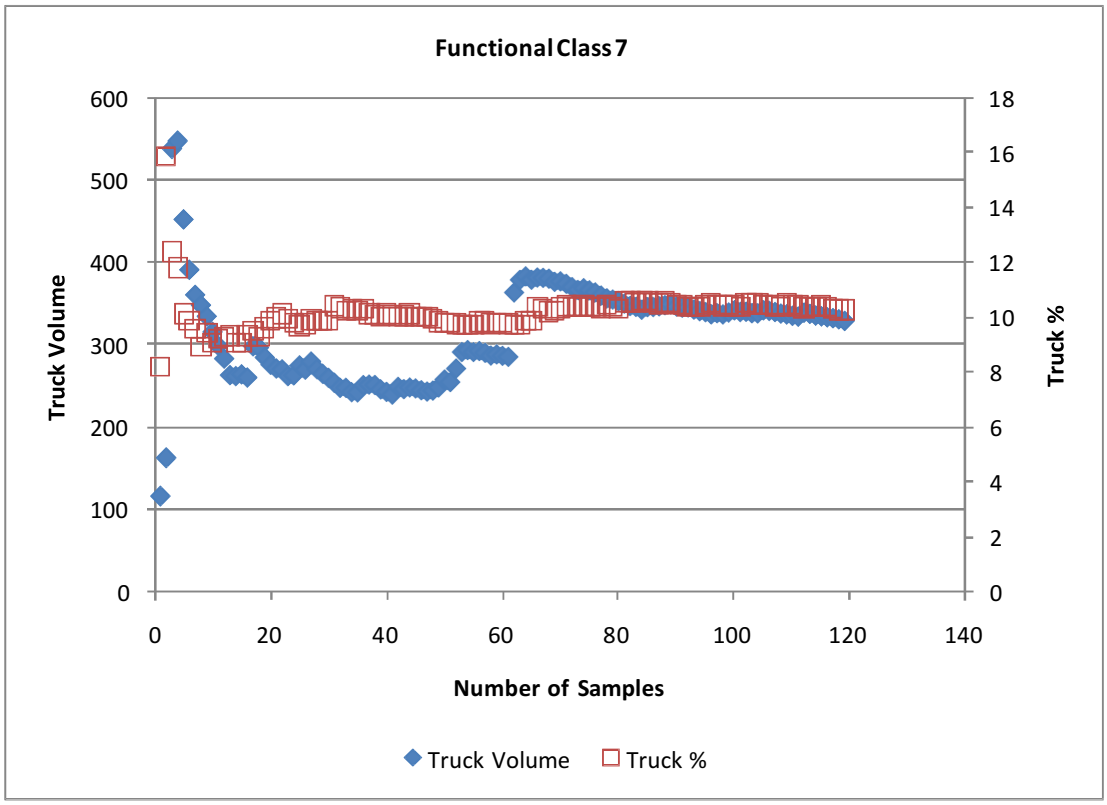


Figure 27 Functional Class 07, Sample Size

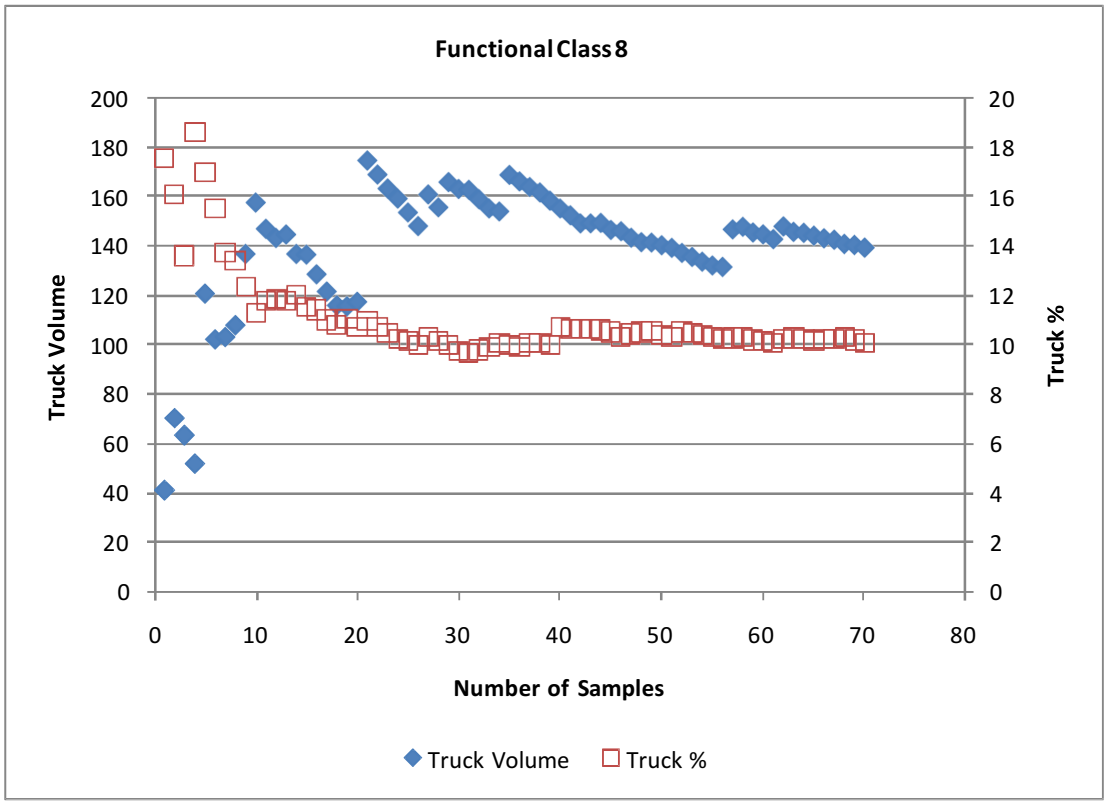


Figure 28 Functional Class 08, Sample Size

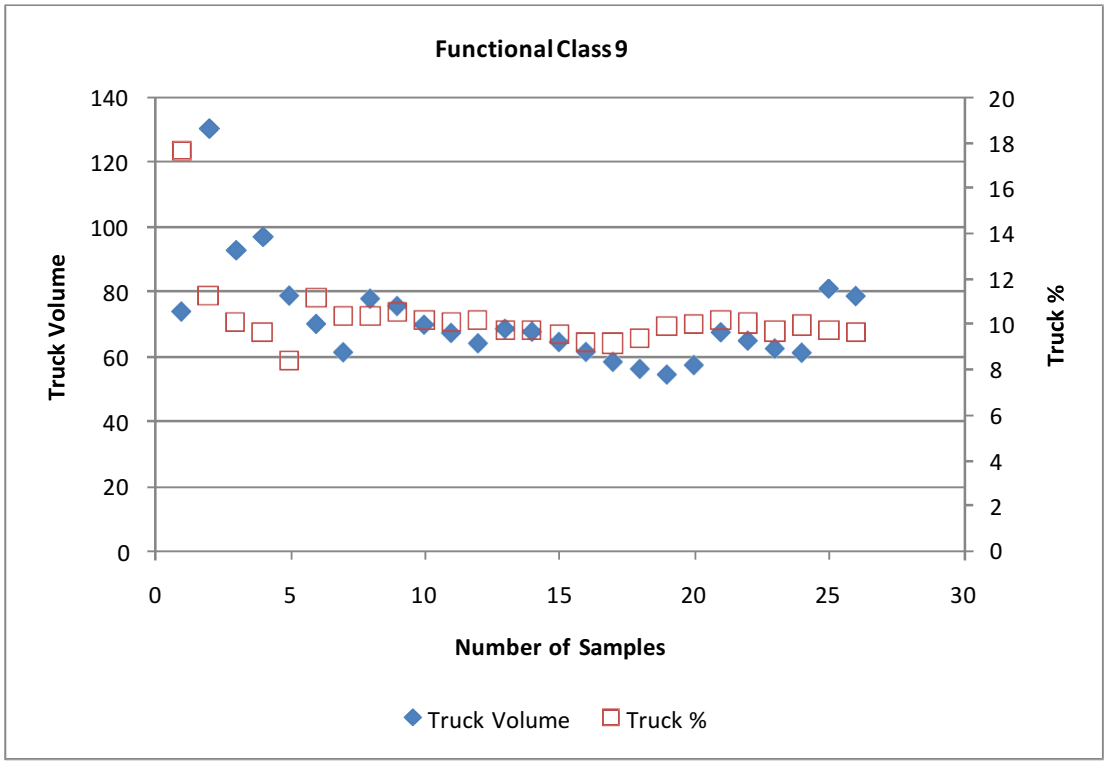


Figure 29 Functional Class 09, Sample Size

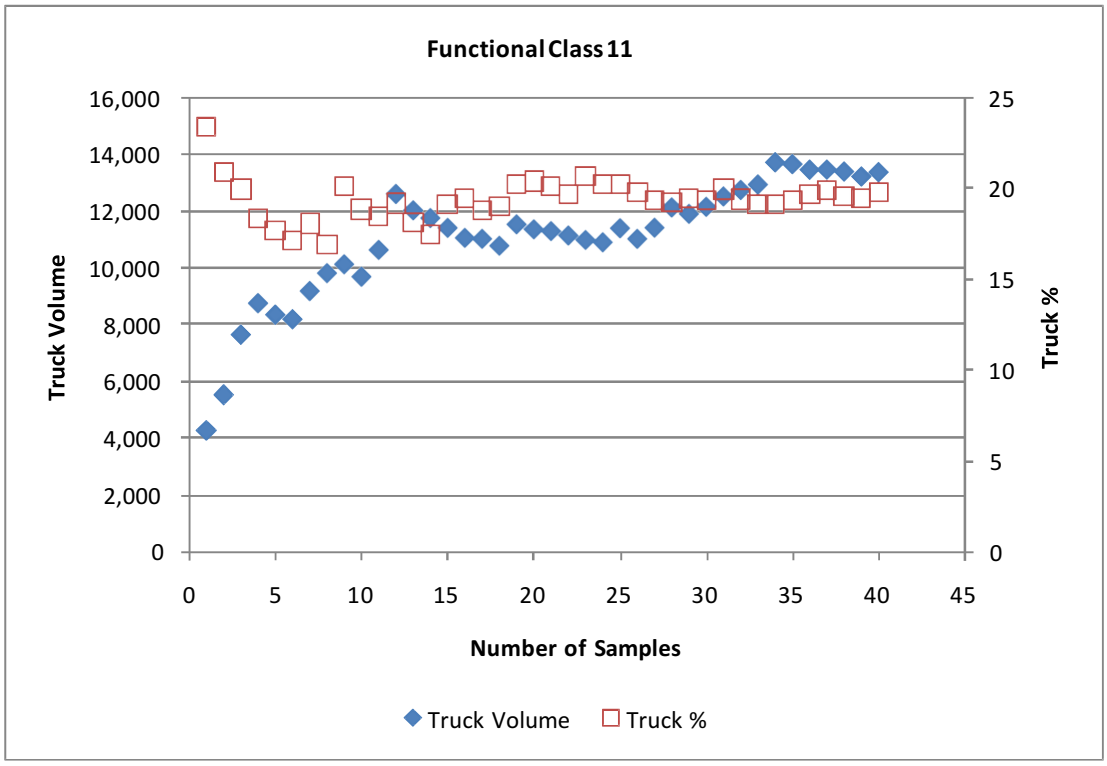


Figure 30 Functional Class 11, Sample Size

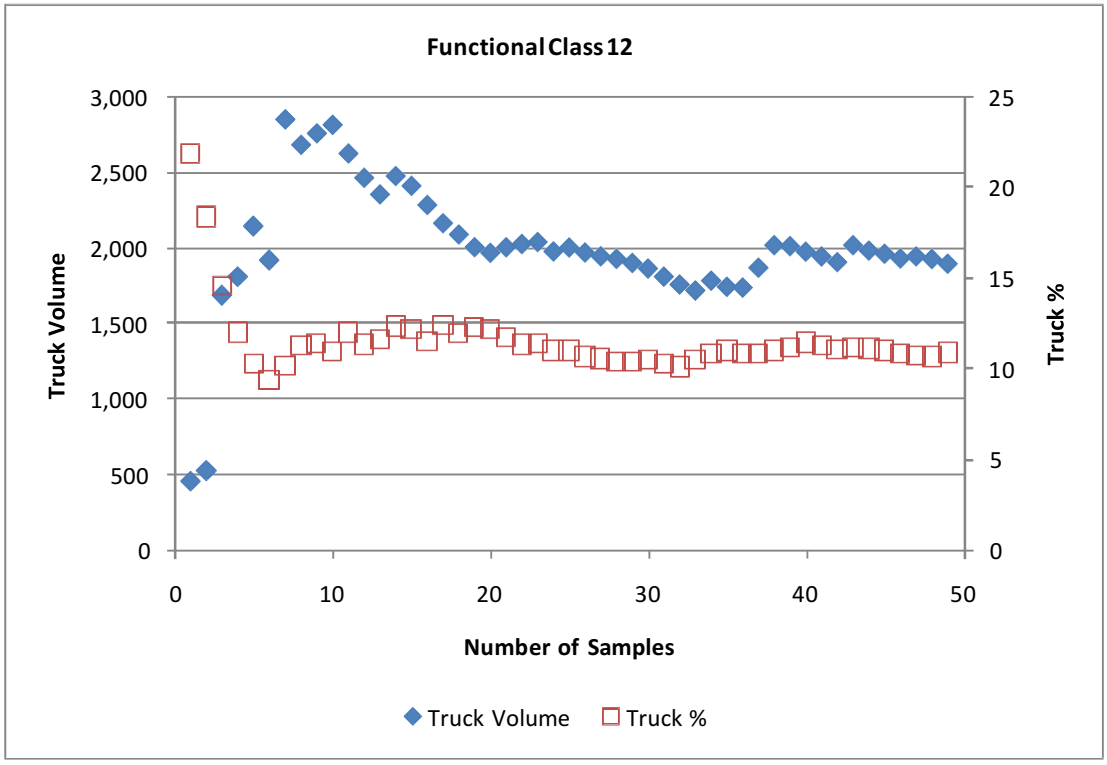


Figure 31 Functional Class 12, Sample Size

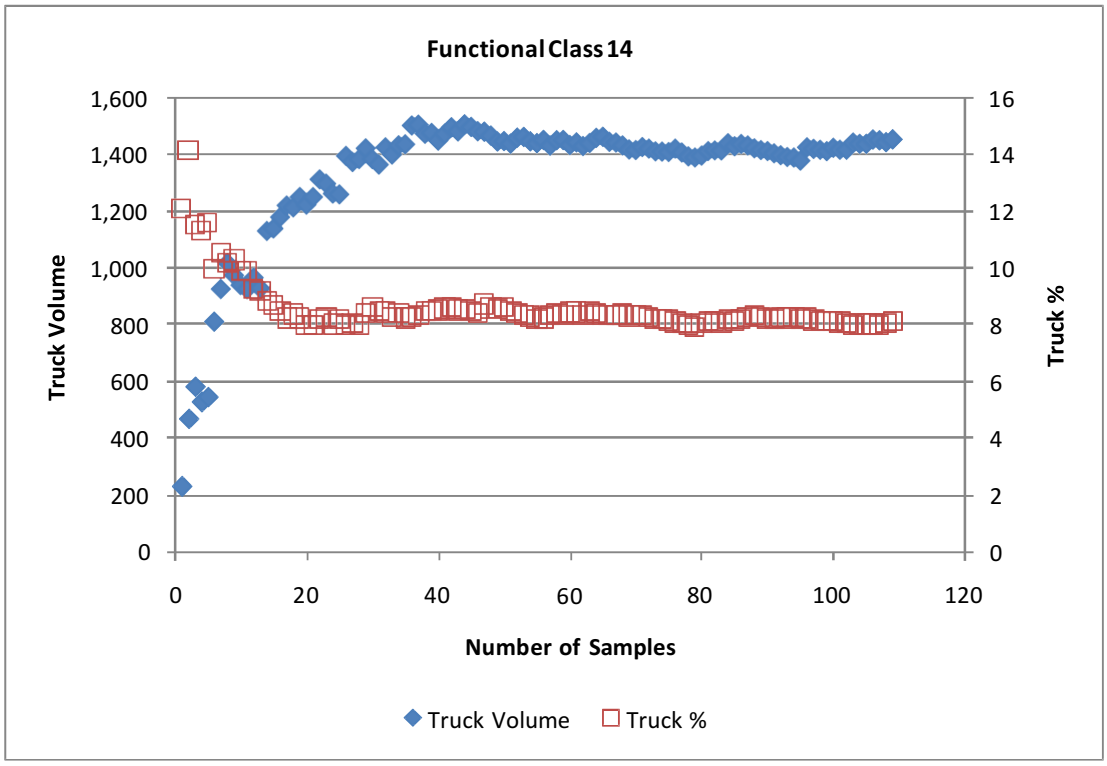


Figure 32 Functional Class 14, Sample Size

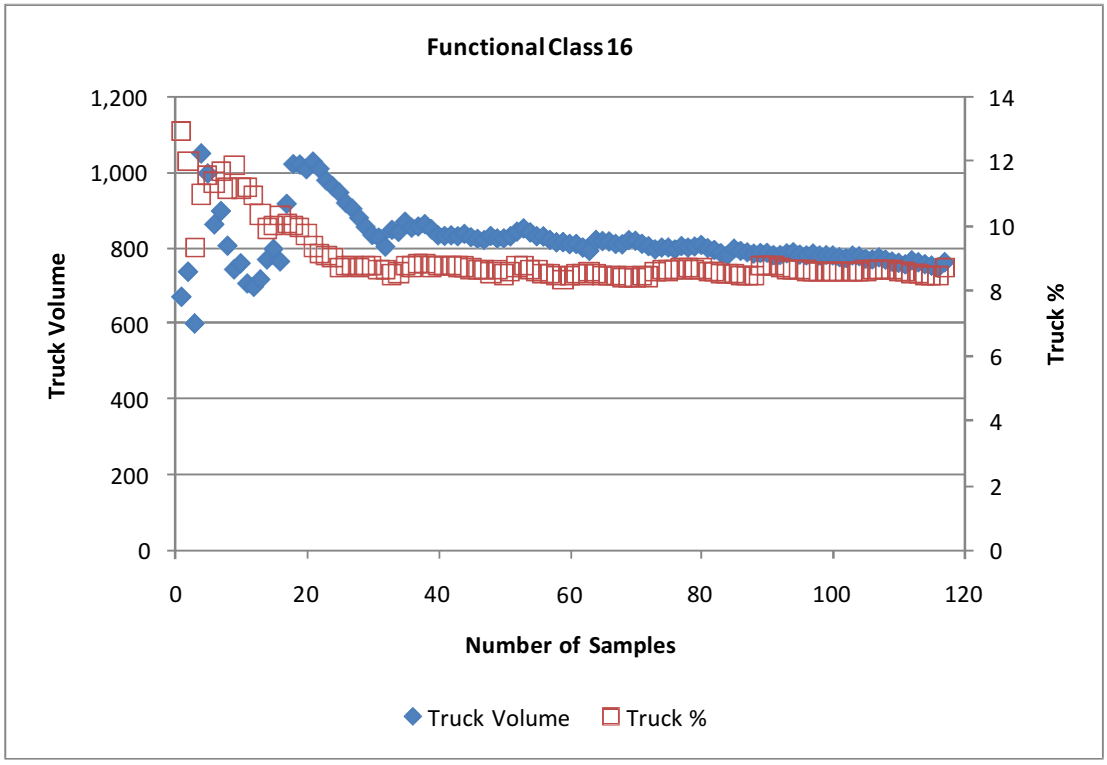


Figure 33 Functional Class 16, Sample Size

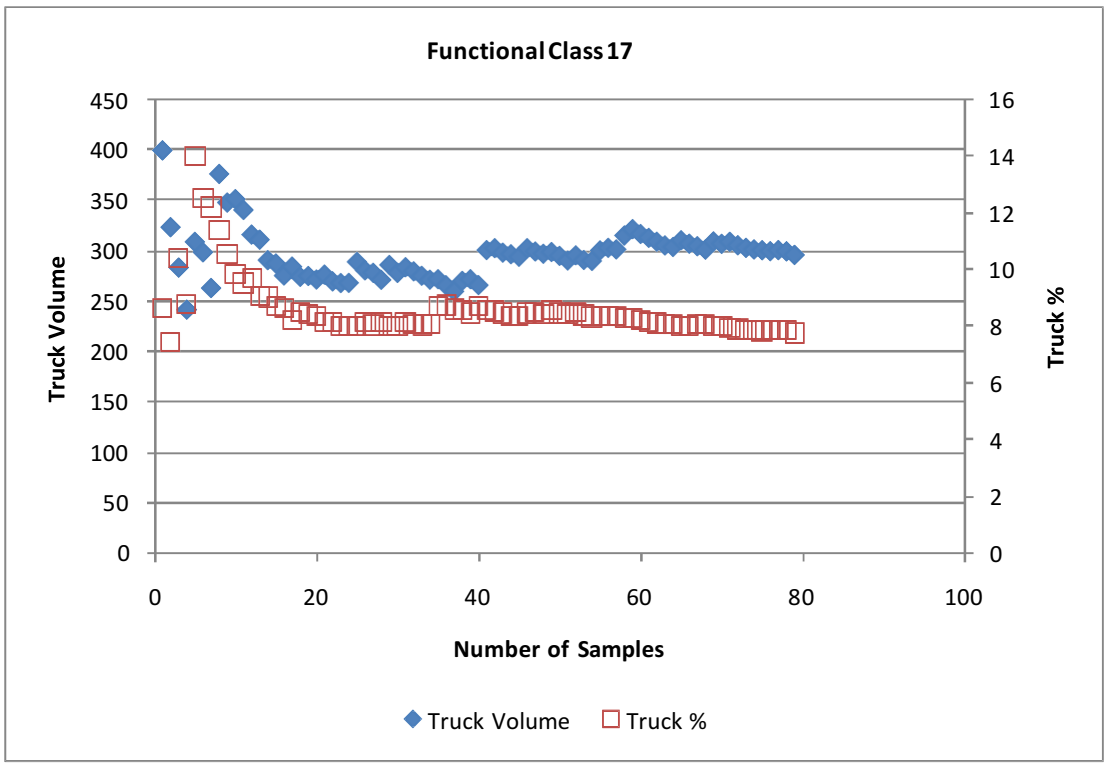


Figure 34 Functional Class 17, Sample Size

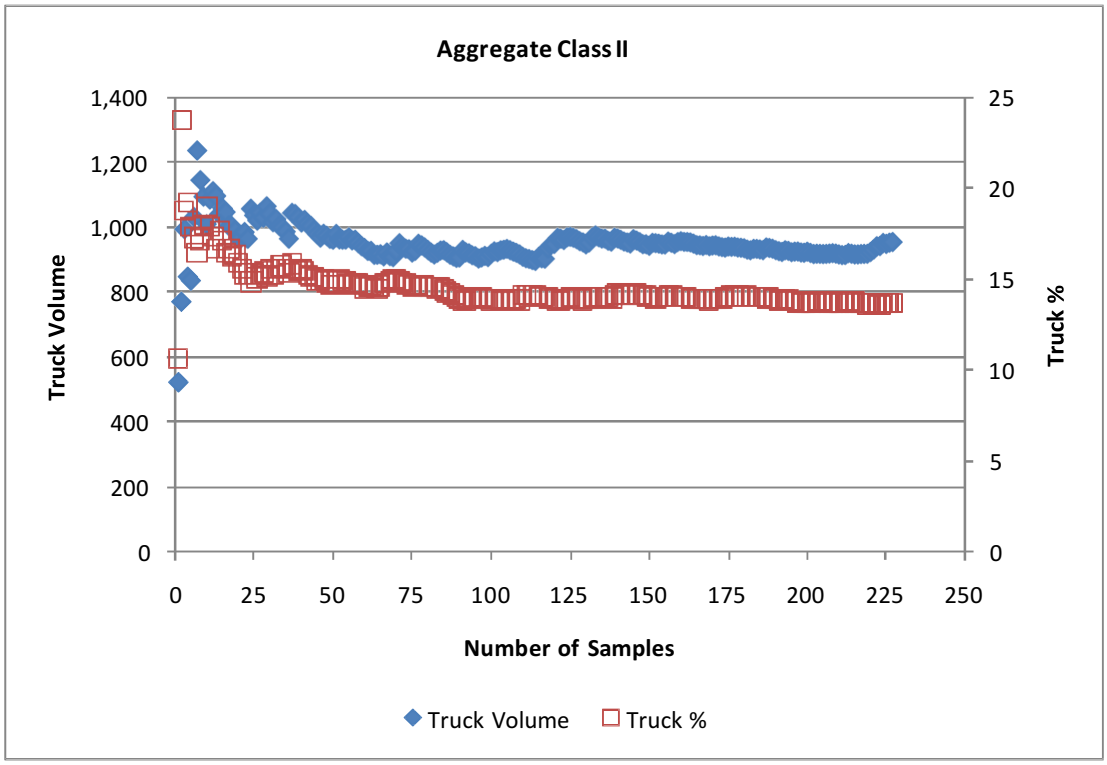


Figure 35 Aggregate ESAL Class II, Sample Size

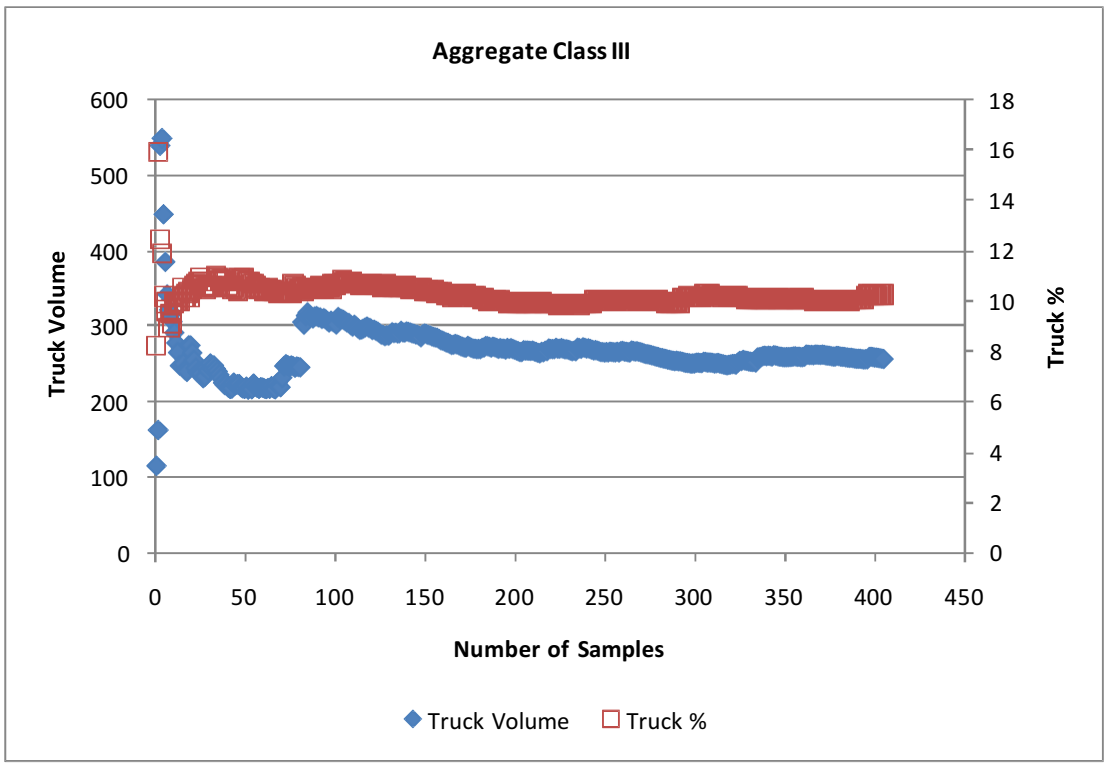


Figure 36 Aggregate ESAL Class III, Sample Size

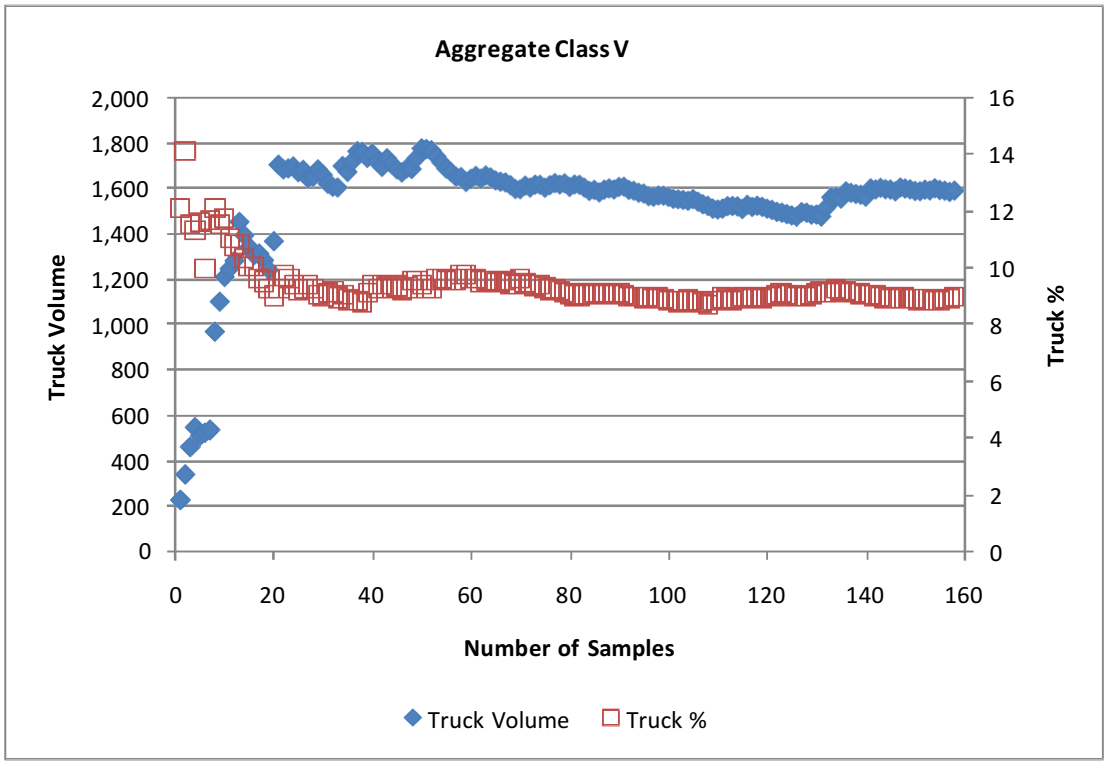


Figure 37 Aggregate ESAL Class V, Sample Size

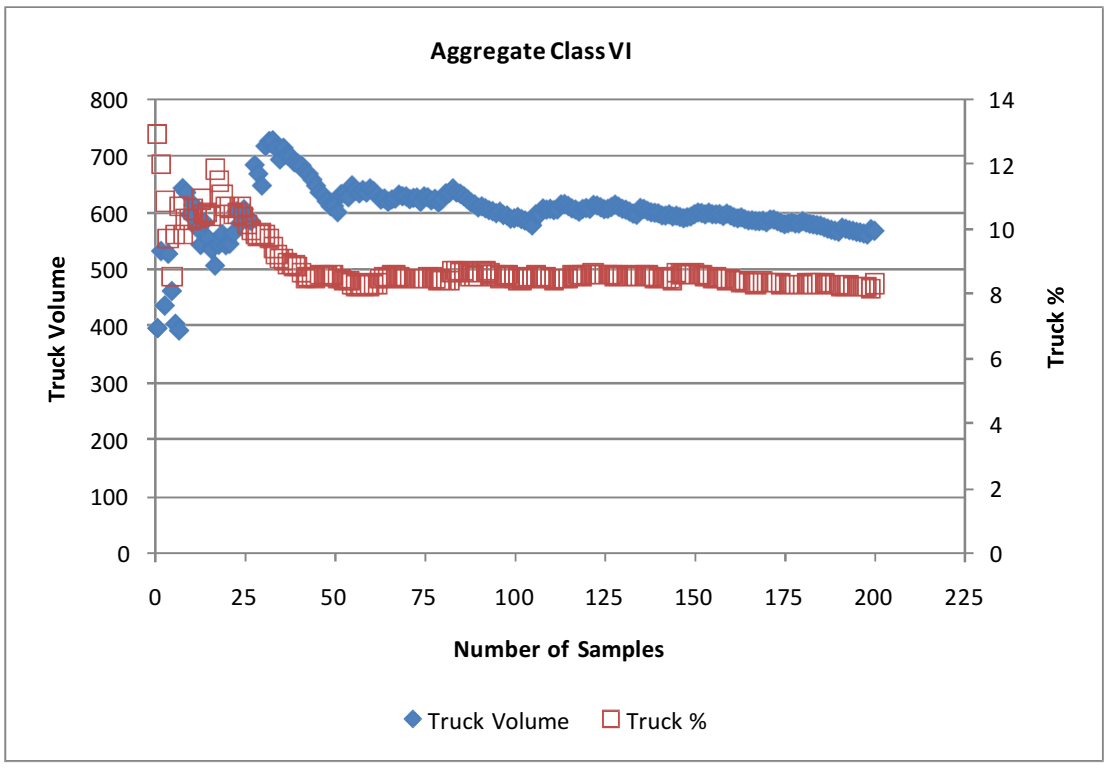


Figure 38 Aggregate ESAL Class VI, Sample Size

5.0 AGGREGATE CLASS SITE ESTIMATES AND AVERAGES

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS

2013

AGGREGATE CLASS I -- RURAL INTERSTATE

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	2-DIRECTION EAL'S IN 1000'S NON-COAL TRUCKS	2-DIRECTION EAL'S IN 1000'S COAL TRUCKS	2-DIRECTION EAL'S IN 1000'S TOTAL	CLASS	YEAR
8	340	I 71	72.0	34561	.297	.000	4.599	.000	.347	.000	44.	5980.	0.	6024.		2012
10	017	I 64	191.0	23561	.216	.000	4.335	.000	.370	.000	34.	2989.	0.	3022.		
10	022	I 64	185.0	19159	.235	.000	4.300	.000	.336	.000	27.	2376.	0.	2403.		
15	503	I 65	103.0	52017	.303	.000	4.526	.000	.327	.000	66.	8515.	0.	8581.		
22	031	I 64	172.0	20245	.204	.000	4.303	.000	.324	.000	29.	2098.	0.	2127.		
24	328	I 24	86.0	33287	.249	.000	4.450	.000	.330	.000	45.	4440.	0.	4485.		
25	001	I 64	98.0	27255	.177	.000	4.198	.000	.324	.000	41.	2390.	0.	2430.		
25	004	I 64	96.0	36917	.137	.000	4.158	.000	.328	.000	58.	2510.	0.	2567.		
34	336	I 75	104.0	46102	.223	.000	4.500	.000	.325	.000	65.	5478.	0.	5543.		
34	782	I 75	118.0	51295	.202	.000	4.493	.000	.323	.000	74.	5485.	0.	5560.		
34	784	I 75	115.0	77508	.146	.000	4.358	.000	.322	.000	121.	5792.	0.	5912.		
39	257	I 71	62.0	32403	.313	.000	4.480	.000	.321	.000	40.	5321.	0.	5362.		
41	251	I 75	156.0	39813	.256	.000	4.493	.000	.333	.000	54.	5571.	0.	5625.		
47	330	I 65	79.0	39489	.365	.000	4.513	.000	.329	.000	45.	7803.	0.	7847.		
50	038	I 65	70.0	35971	.394	.000	4.706	.000	.329	.000	39.	8026.	0.	8065.		
50	042	I 65	64.0	35870	.388	.000	4.755	.000	.334	.000	40.	8076.	0.	8116.		
56	019	I 64	19.0	54047	.166	.000	4.212	.000	.320	.000	82.	4423.	0.	4504.		
59	521	I 75	166.0	49703	.213	.000	4.426	.000	.335	.000	71.	5722.	0.	5793.		
63	827	I 75	41.0	33678	.281	.000	4.461	.000	.321	.000	44.	4955.	0.	4998.		
76	637	I 75	77.0	47210	.232	.000	4.553	.000	.335	.000	66.	6102.	0.	6167.		
79	044	I 24	25.0	25999	.306	.000	4.513	.000	.330	.000	33.	4329.	0.	4361.		
79	049	I 24	27.0	26309	.295	.000	4.616	.000	.361	.000	34.	4721.	0.	4755.		
79	850	I 24	17.0	27343	.231	.000	4.617	.000	.333	.000	38.	3543.	0.	3581.		
102	068	I 75	59.0	31219	.294	.000	4.644	.000	.338	.000	40.	5256.	0.	5296.		
102	070	I 75	62.0	38147	.281	.000	4.515	.000	.327	.000	50.	5772.	0.	5822.		
103	027	I 64	137.0	12987	.220	.000	4.313	.000	.322	.000	18.	1447.	0.	1465.		
105	002	I 75	136.0	32896	.363	.000	4.712	.000	.370	.000	38.	7597.	0.	7635.		
106	254	I 64	43.0	41154	.198	.000	4.086	.000	.315	.000	60.	3825.	0.	3885.		
5	831	I 65	48.0	26137	.318	.000	4.529	.000	.326	.000	32.	4488.	0.	4520.		2009
5	836	I 65	44.7	36854	.307	.000	4.567	.000	.328	.000	46.	6179.	0.	6224.		
8	266	I 75	174.1	86448	.121	.000	4.548	.000	.329	.000	139.	5729.	0.	5867.		
8	340	I 71	76.2	33890	.288	.000	4.729	.000	.333	.000	44.	5611.	0.	5655.		
8	766	I 275	8.4	37216	.137	.000	4.135	.000	.359	.000	57.	2765.	0.	2822.		
8	767	I 275	12.4	32430	.125	.000	4.129	.000	.319	.000	51.	1951.	0.	2002.		
22	031	I 64	177.4	20790	.158	.000	4.462	.000	.325	.000	32.	1735.	0.	1767.		

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS I -- RURAL INTERSTATE

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL	
22	050	I 64	180.1	21164	.217	.000	4.032	.000	.338	.000	28.	2287.	0.	2316.	
22	554	I 64	158.1	13324	.209	.000	3.799	.000	.322	.000	19.	1242.	0.	1261.	
25	001	I 64	99.7	26891	.169	.000	3.642	.000	.315	.000	40.	1905.	0.	1946.	
25	755	I 64	89.1	35722	.146	.000	4.535	.000	.328	.000	56.	2820.	0.	2876.	
39	522	I 71	58.8	28580	.429	.000	4.619	.000	.328	.000	29.	6793.	0.	6822.	
39	525	I 71	53.5	29085	.331	.000	4.136	.000	.315	.000	33.	4568.	0.	4601.	
39	537	I 71	55.3	31542	.315	.000	3.934	.000	.307	.000	36.	4382.	0.	4418.	
41	251	I 75	156.5	39902	.144	.000	4.229	.000	.345	.000	62.	3051.	0.	3113.	
41	261	I 75	153.6	33941	.157	.000	4.257	.000	.324	.000	52.	2681.	0.	2733.	
41	368	I 75	155.0	38040	.149	.000	4.185	.000	.328	.000	59.	2845.	0.	2904.	
47	174	I 65	100.5	45381	.312	.000	4.819	.000	.338	.000	57.	8418.	0.	8474.	
50	038	I 65	73.9	36285	.314	.000	4.627	.000	.329	.000	45.	6318.	0.	6363.	
50	042	I 65	66.2	35967	.404	.000	4.849	.000	.338	.000	39.	8696.	0.	8735.	
52	782	I 71	31.1	31081	.299	.000	4.735	.000	.335	.000	40.	5392.	0.	5432.	
56	019	I 64	20.7	51360	.145	.000	4.157	.000	.352	.000	80.	3981.	0.	4060.	
59	521	I 75	170.0	49192	.098	.000	4.542	.000	.332	.000	81.	2653.	0.	2734.	
63	279	I 75	50.1	32036	.338	.000	4.608	.000	.331	.000	38.	6032.	0.	6070.	
63	547	I 75	30.7	37512	.244	.000	4.410	.000	.322	.000	52.	4741.	0.	4792.	
63	827	I 75	45.9	32139	.258	.000	4.406	.000	.324	.000	43.	4330.	0.	4374.	
102	068	I 75	59.9	31535	.256	.000	4.541	.000	.331	.000	43.	4418.	0.	4460.	
102	070	I 75	68.3	36484	.278	.000	4.792	.000	.336	.000	48.	5957.	0.	6005.	
102	279	I 75	50.1	32036	.336	.000	4.576	.000	.330	.000	39.	5930.	0.	5969.	
103	027	I 64	145.9	13140	.172	.000	4.651	.000	.332	.000	20.	1276.	0.	1296.	
103	769	I 64	134.1	19518	.171	.000	4.668	.000	.333	.000	30.	1892.	0.	1921.	
106	509	I 64	33.7	46453	.153	.000	4.093	.000	.332	.000	72.	3533.	0.	3605.	
107	285	I 65	1.0	38721	.349	.000	4.844	.000	.337	.000	46.	8056.	0.	8102.	
114	059	I 65	42.0	43296	.339	.000	4.809	.000	.338	.000	52.	8713.	0.	8765.	
114	065	I 65	30.7	46849	.314	.000	4.818	.000	.337	.000	59.	8716.	0.	8775.	
114	573	I 65	20.0	40343	.392	.000	4.709	.000	.337	.000	44.	9148.	0.	9192.	
120	798	I 64	63.5	35232	.185	.000	3.725	.000	.327	.000	51.	2894.	0.	2945.	
8	340	I 71	76.2	33890	.300	.000	4.720	.000	.332	.000	43.	5824.	0.	5867.	2010
39	522	I 71	58.8	28580	.363	.000	4.570	.000	.334	.000	33.	5765.	0.	5798.	

SUMMARY OF AVERAGE VALUES FOR
AGGREGATE CLASS I -- RURAL INTERSTATE

YEAR	3 YR AVG	13	11	07	05	04	01	00	99	98	97	96
UNCLASSIFIED ROADS (ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)												
NO OF CLASSIFICATION STA.	168	67	68	33	42	36	24	10	9	17	20	15
AADT	33928	34220	34277	32616	35488	33594	31586	28424	26600	23935	34777	22292
PERCENT TRUCKS	26.721	25.262	26.152	30.857	31.312	30.361	31.636	31.224	35.555	34.124	28.932	27.300
AXLES PER TRUCK	4.517	4.447	4.510	4.673	4.492	4.495	4.456	4.489	4.648	4.486	4.501	4.489
EAL'S PER TRUCK AXLE	.278	.332	.267	.194	.302	.322	.239	.258	.220	.226	.208	.231
CLASSIFIED ROADS (MANUAL LOCATION WITH 3% OR MORE OF TRUCKS CLASSIFIED AS HEAVY/COAL)												
NO OF CLASSIFICATION STA.	0	0	0	0	0	0	1	0	0	0	0	1
AADT	0	0	0	0	0	0	21300	0	0	0	0	3687
PERCENT TRUCKS	.000	.000	.000	.000	.000	.000	23.543	.000	.000	.000	.000	17.738
PERCENT OF TRUCKS CLASSIFIED AS HEAVY/COAL	.000	.000	.000	.000	.000	.000	3.878	.000	.000	.000	.000	4.128
AXLES PER TRUCK NORMAL	.000	.000	.000	.000	.000	.000	4.252	.000	.000	.000	.000	5.165
AXLES PER TRUCK HEAVY/COAL	.000	.000	.000	.000	.000	.000	4.637	.000	.000	.000	.000	4.778
EAL'S PER TRUCK AXLE NORMAL	.000	.000	.000	.000	.000	.000	.244	.000	.000	.000	.000	.292
EAL'S PER TRUCK AXLE HEAVY/COAL	.000	.000	.000	.000	.000	.000	1.870	.000	.000	.000	.000	.880

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS

2013
AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/
RURAL MINOR ARTERIAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
1	008	KY 55	2.0	8741	.087	.000	3.340	.000	.362	.000	15.	335.	0.	350.
1	047	KY 55	15.0	6162	.131	.000	3.830	.000	.286	.000	10.	324.	0.	334.
1	A61	KY 80	12.0	5419	.050	.000	3.109	.000	.333	.000	9.	103.	0.	112.
1	A62	KY 61	11.0	7112	.098	.000	3.722	.000	.347	.000	12.	327.	0.	339.
1	A68	KY 55	13.0	3311	.165	.000	3.891	.000	.293	.000	5.	227.	0.	232.
5	003	US 31	17.0	4819	.067	.000	3.343	.000	.415	.000	8.	164.	0.	172.
7	003	US 119	8.0	4994	.117	.000	4.632	.000	.359	.000	8.	355.	0.	363.
8	R04	US 25	.0	6837	.129	.000	3.091	.000	.421	.000	11.	420.	0.	431.
11	006	KY 52	.0	5671	.080	.000	4.731	.000	.482	.000	9.	376.	0.	386.
11	036	US 150	8.0	6634	.091	.000	2.526	.000	.172	.000	11.	96.	0.	107.
11	037	US 127	8.0	13827	.100	.000	3.362	.000	.319	.000	22.	543.	0.	565.
12	016	KY 19	14.0	1744	.094	.000	2.909	.000	.214	.000	3.	37.	0.	40.
15	846	KY 61	10.0	3646	.067	.000	2.584	.000	.273	.000	6.	63.	0.	69.
18	643	KY 80	5.0	4019	.127	.000	3.507	.000	.316	.000	6.	206.	0.	212.
19	033	KY 547	2.0	2267	.171	.000	2.526	.000	.293	.000	3.	105.	0.	108.
20	006	KY 121	3.0	1562	.272	.000	3.985	.000	.222	.000	2.	137.	0.	140.
21	263	KY 227	1.0	3453	.097	.000	2.531	.000	.470	.000	6.	145.	0.	151.
22	044	KY 9	.0	4428	.262	.000	4.075	.000	.392	.000	6.	675.	0.	681.
23	037	US 127	19.0	4754	.109	.000	3.584	.000	.285	.000	8.	193.	0.	201.
23	A51	US 127	12.0	7485	.101	.000	3.437	.000	.297	.000	12.	281.	0.	293.
24	B31	US 68	7.0	4148	.141	.000	2.758	.000	.210	.000	6.	123.	0.	130.
27	758	KY 90	2.0	2195	.219	.000	4.307	.000	.283	.000	3.	214.	0.	217.
28	A44	US 641	7.0	5195	.125	.000	3.304	.000	.444	.000	8.	349.	0.	357.
28	A58	US 60	8.0	4113	.096	.000	3.223	.000	.670	.000	7.	312.	0.	319.
29	257	KY 61	.0	1348	.223	.000	3.730	.000	.258	.000	2.	106.	0.	107.
30	800	US 60	1.0	2640	.145	.000	4.102	.000	.256	.000	4.	147.	0.	151.
30	822	AU9005	16.0	8932	.207	.000	4.103	.000	.253	.000	13.	703.	0.	716.
30	A99	US 231	11.0	5683	.314	.000	4.611	.000	.263	.000	7.	792.	0.	799.
31	794	KY 259	17.0	2809	.083	.000	3.153	.000	.273	.000	5.	73.	0.	78.
31	A14	KY 70	10.0	6945	.060	.000	3.205	.000	.336	.000	12.	163.	0.	175.
34	296	US 60	13.0	13650	.069	.000	2.476	.000	.199	.000	23.	168.	0.	191.
36	507	KY 80	6.0	12022	.097	.000	3.431	.000	.373	.000	20.	547.	0.	567.
36	758	KY 3	1.0	9004	.111	.000	2.767	.000	.297	.000	15.	300.	0.	315.
36	768	KY 321	1.0	5987	.032	.000	2.643	.000	.293	.000	10.	55.	0.	65.
36	791	US 23	16.0	15235	.119	.000	4.070	.000	.373	.000	24.	1003.	0.	1027.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/
RURAL MINOR ARTERIAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
37	271	US 421	.0	5674	.058	.000	3.280	.000	.533	.000	10.	211.	0.	221.
40	500	KY1295	.0	1842	.077	.000	2.437	.000	.157	.000	3.	20.	0.	23.
40	A10	US 27	3.0	10347	.105	.000	3.372	.000	.368	.000	17.	491.	0.	508.
41	354	KY 36	19.0	1140	.085	.000	2.756	.000	.259	.000	2.	25.	0.	27.
43	560	WK9001	88.0	6562	.351	.000	4.302	.000	.246	.000	8.	890.	0.	897.
44	040	KY 61	10.0	4086	.079	.000	3.151	.000	.407	.000	7.	150.	0.	157.
45	002	US 23	18.0	7277	.168	.000	3.894	.000	.277	.000	11.	482.	0.	493.
47	753	KY 86	12.0	3215	.085	.000	2.977	.000	.284	.000	5.	85.	0.	90.
48	256	US 421	.0	1906	.078	.000	3.025	.000	.178	.000	3.	29.	0.	32.
48	777	US 119	16.0	5417	.084	.000	3.723	.000	.340	.000	9.	209.	0.	219.
51	155	AU9005	.0	8355	.225	.000	4.020	.000	.245	.000	12.	676.	0.	687.
51	554	US 60	.0	5562	.122	.000	3.629	.000	.268	.000	9.	240.	0.	249.
52	506	US 421	10.0	1564	.082	.000	2.735	.000	.417	.000	3.	53.	0.	56.
53	501	KY 239	.0	856	.145	.000	3.548	.000	.284	.000	1.	46.	0.	47.
53	A59	US 51	7.0	2895	.176	.000	4.280	.000	.277	.000	4.	221.	0.	225.
54	063	EB9004	45.0	15875	.263	.000	4.015	.000	.267	.000	20.	1632.	0.	1653.
55	308	US 421	4.0	3266	.053	.000	2.819	.000	.280	.000	6.	50.	0.	55.
55	337	KY 30	2.0	2599	.071	.000	3.268	.000	.291	.000	4.	64.	0.	69.
57	A86	US 27	6.0	18027	.121	.000	3.102	.000	.316	.000	29.	782.	0.	811.
58	303	US 23	2.0	9877	.157	.000	3.806	.000	.322	.000	15.	695.	0.	710.
58	A82	KY 321	8.0	13711	.084	.000	3.238	.000	.404	.000	22.	553.	0.	575.
60	774	KY 80	6.0	6673	.119	.000	3.750	.000	.324	.000	11.	353.	0.	364.
61	C33	KY3041	2.0	7227	.052	.000	2.908	.000	.247	.000	12.	99.	0.	112.
62	318	US 31	7.0	8728	.086	.000	2.773	.000	.308	.000	14.	235.	0.	250.
62	750	KY 61	9.0	9876	.098	.000	3.479	.000	.303	.000	16.	372.	0.	388.
62	A13	US 31	10.0	7467	.053	.000	2.993	.000	.429	.000	13.	187.	0.	200.
63	255	US 25	.0	13342	.082	.000	3.819	.000	.467	.000	22.	715.	0.	738.
63	828	KY 80	.0	6124	.243	.000	4.017	.000	.438	.000	8.	955.	0.	964.
64	260	US 23	7.0	8446	.237	.000	4.437	.000	.351	.000	12.	1140.	0.	1152.
64	A37	KY 3	15.0	9364	.145	.000	3.822	.000	.370	.000	14.	701.	0.	716.
64	A59	KY 3	16.0	8596	.094	.000	3.498	.000	.366	.000	14.	376.	0.	390.
65	272	KY 11	2.0	5281	.061	.000	3.366	.000	.314	.000	9.	124.	0.	133.
65	A28	KY 11	3.0	6066	.076	.000	2.703	.000	.260	.000	10.	117.	0.	128.
67	255	US 119	20.0	10758	.068	.000	3.517	.000	.362	.000	18.	342.	0.	360.
67	258	US 119	9.0	2002	.140	.000	3.941	.000	.333	.000	3.	134.	0.	138.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/
RURAL MINOR ARTERIAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
67	760	KY 7	13.0	8427	.112	.000	3.615	.000	.393	.000	14.	489.	0.	503.
67	768	KY 15	9.0	6110	.148	.000	3.584	.000	.352	.000	9.	417.	0.	427.
68	041	KY 10	17.0	1352	.154	.000	3.655	.000	.270	.000	2.	75.	0.	77.
69	315	US 27	3.0	6008	.090	.000	3.726	.000	.334	.000	10.	245.	0.	255.
69	329	US 150	16.0	3088	.166	.000	3.751	.000	.254	.000	5.	179.	0.	183.
69	A21	US 150	4.0	12558	.102	.000	3.661	.000	.374	.000	20.	639.	0.	660.
69	A56	US 150	6.0	5369	.103	.000	3.792	.000	.268	.000	9.	206.	0.	215.
71	255	US 68	25.0	10027	.161	.000	3.707	.000	.268	.000	15.	586.	0.	601.
71	502	KY 79	2.0	3702	.073	.000	3.261	.000	.281	.000	6.	91.	0.	97.
71	781	US 431	23.0	4743	.093	.000	3.341	.000	.247	.000	8.	133.	0.	141.
74	258	US 27	.0	5135	.132	.000	3.879	.000	.325	.000	8.	311.	0.	319.
75	271	US 431	.0	3622	.171	.000	4.421	.000	.273	.000	5.	273.	0.	278.
76	521	KY1295	.0	1768	.062	.000	3.071	.000	.333	.000	3.	41.	0.	44.
77	294	US 460	15.0	4407	.128	.000	4.290	.000	.355	.000	7.	313.	0.	320.
77	A14	US 460	12.0	8933	.044	.000	3.013	.000	.255	.000	16.	110.	0.	125.
79	A24	US 641	8.0	3661	.035	.000	3.177	.000	.546	.000	6.	80.	0.	87.
80	502	KY 3	.0	3880	.100	.000	4.197	.000	.378	.000	6.	225.	0.	231.
80	762	KY 645	.0	3632	.149	.000	4.058	.000	.400	.000	6.	321.	0.	326.
82	264	US 60	8.0	5633	.088	.000	3.586	.000	.264	.000	9.	171.	0.	180.
82	A23	KY 79	10.0	4808	.114	.000	3.454	.000	.281	.000	8.	195.	0.	202.
82	A35	KY1051	1.0	13049	.079	.000	2.976	.000	.388	.000	22.	433.	0.	455.
83	A11	US 460	8.0	4645	.061	.000	2.843	.000	.269	.000	8.	79.	0.	87.
84	001	US 68	14.0	2942	.041	.000	2.686	.000	.307	.000	5.	36.	0.	41.
84	256	US 68	10.0	2321	.064	.000	2.854	.000	.748	.000	4.	116.	0.	120.
88	015	KY 7	3.0	1776	.114	.000	3.096	.000	.242	.000	3.	56.	0.	58.
88	253	US 460	26.0	1214	.082	.000	2.977	.000	.323	.000	2.	35.	0.	37.
88	501	US 460	11.0	2797	.059	.000	2.865	.000	.326	.000	5.	57.	0.	61.
88	A47	US 460	13.0	6983	.092	.000	3.038	.000	.513	.000	12.	367.	0.	378.
90	017	US 31	17.0	7950	.084	.000	2.932	.000	.287	.000	13.	206.	0.	219.
90	058	US 31	21.0	5633	.080	.000	3.074	.000	.276	.000	9.	140.	0.	149.
90	281	US 150	2.0	11370	.132	.000	3.542	.000	.302	.000	18.	587.	0.	605.
94	757	KY 35	.0	2111	.175	.000	3.360	.000	.317	.000	3.	144.	0.	147.
97	A84	KY 15	11.0	13260	.110	.000	3.863	.000	.447	.000	21.	922.	0.	943.
98	120	US 119	10.0	7196	.120	.000	3.374	.000	.370	.000	11.	395.	0.	406.
98	272	US 460	14.0	4787	.171	.000	4.133	.000	.391	.000	7.	483.	0.	490.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/
RURAL MINOR ARTERIAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
98	812	US 119	8.0	7281	.154	.000	3.538	.000	.437	.000	11.	631.	0.	642.
99	270	KY 11	1.0	2794	.059	.000	3.093	.000	.325	.000	5.	60.	0.	65.
99	790	KY9000	.0	1401	.048	.000	3.440	.000	.362	.000	2.	30.	0.	33.
99	791	KY9000	.0	1441	.049	.000	3.139	.000	.267	.000	3.	21.	0.	24.
99	C11	KY 11	21.0	10861	.034	.000	2.575	.000	.204	.000	19.	71.	0.	90.
100	009	KY 80	22.0	16376	.160	.000	3.929	.000	.294	.000	25.	1106.	0.	1131.
100	C00	KY 914	6.0	6664	.100	.000	3.459	.000	.318	.000	11.	267.	0.	278.
100	D04	KY 90	3.0	14462	.074	.000	3.342	.000	.230	.000	24.	300.	0.	325.
100	D09	US 27	9.0	13521	.106	.000	3.919	.000	.364	.000	22.	747.	0.	769.
100	D20	US 27	17.0	12626	.061	.000	3.396	.000	.292	.000	22.	279.	0.	300.
100	D21	US 27	19.0	12807	.062	.000	3.287	.000	.232	.000	22.	222.	0.	244.
102	794	KY 461	8.0	9236	.160	.000	4.146	.000	.295	.000	14.	658.	0.	672.
102	822	US 150	2.0	5766	.120	.000	3.607	.000	.279	.000	9.	254.	0.	263.
104	006	US 127	20.0	5868	.103	.000	3.517	.000	.330	.000	10.	256.	0.	266.
105	020	US 62	13.0	6774	.084	.000	2.794	.000	.168	.000	11.	97.	0.	108.
106	254	US 421	.0	1299	.199	.000	4.099	.000	.242	.000	2.	94.	0.	95.
106	514	KY 55	1.0	7875	.091	.000	3.032	.000	.304	.000	13.	243.	0.	256.
106	A79	KY 55	9.0	3652	.063	.000	3.090	.000	.320	.000	6.	83.	0.	89.
109	534	KY 55	5.0	7358	.123	.000	3.566	.000	.264	.000	12.	311.	0.	323.
110	298	US 41	.0	4523	.109	.000	3.377	.000	.372	.000	7.	226.	0.	233.
110	519	US 79	.0	7076	.131	.000	3.758	.000	.338	.000	11.	430.	0.	441.
111	529	US 68	.0	2918	.136	.000	3.438	.000	.300	.000	5.	150.	0.	154.
113	780	KY 56	8.0	1753	.214	.000	3.306	.000	.265	.000	2.	120.	0.	122.
113	B16	KY3393	.0	1264	.241	.000	3.522	.000	.311	.000	2.	122.	0.	124.
113	B19	US 60	1.0	2982	.152	.000	3.625	.000	.361	.000	5.	216.	0.	220.
114	008	US 31	21.0	6912	.068	.000	3.140	.000	.363	.000	12.	195.	0.	207.
114	795	WN9007	10.0	10292	.214	.000	3.717	.000	.277	.000	14.	828.	0.	842.
115	255	US 150	15.0	2173	.183	.000	3.895	.000	.390	.000	3.	220.	0.	223.
117	032	US 41	16.0	2970	.118	.000	3.859	.000	.335	.000	5.	165.	0.	170.
119	258	KY 15	.0	3899	.108	.000	3.618	.000	.517	.000	6.	288.	0.	294.
119	504	KY 11	.0	2581	.127	.000	3.307	.000	.538	.000	4.	212.	0.	216.
119	507	KY 15	.0	6215	.099	.000	3.407	.000	.487	.000	10.	371.	0.	381.
120	023	US 60	11.0	38672	.115	.000	3.174	.000	.265	.000	62.	1362.	0.	1425.
120	082	BG9002	.0	2806	.046	.000	2.397	.000	.162	.000	5.	18.	0.	23.
24	011	US 68	14.8	5200	.162	.000	3.534	.000	.266	.000	8.	290.	0.	297.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/
RURAL MINOR ARTERIAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
27	032	KY 90	8.8	4185	.166	.000	3.981	.000	.304	.000	6.	308.	0.	314.
27	289	US 127	3.5	3970	.199	.000	4.051	.000	.420	.000	6.	491.	0.	496.
27	A08	US 127	6.2	12129	.169	.005	4.166	5.200	.418	4.064	18.	1295.	77.	1390.
29	A24	KY 61	10.8	2708	.108	.000	3.580	.000	.266	.000	4.	103.	0.	107.
30	A99	US 231	13.8	5770	.395	.000	4.052	.000	.247	.000	6.	832.	0.	839.
36	276	US 23	1.1	20064	.115	.000	3.561	.000	.429	.000	32.	1284.	0.	1316.
36	535	KY 80	8.9	11624	.102	.000	3.620	.000	.382	.000	19.	600.	0.	619.
36	779	US 23	13.3	23570	.095	.000	4.515	.000	.234	.000	39.	860.	0.	899.
36	A13	KY 321	.1	9882	.057	.000	3.141	.000	.461	.000	17.	297.	0.	314.
39	502	KY 35	1.7	2720	.130	.000	3.208	.000	.361	.000	4.	149.	0.	153.
41	502	KY 22	5.7	1907	.075	.000	2.485	.000	.178	.000	3.	23.	0.	27.
46	764	US 60	.8	7837	.230	.000	2.734	.000	.284	.000	10.	510.	0.	520.
51	158	KY9004	75.0	11342	.218	.009	3.894	5.182	.280	3.911	16.	977.	163.	1155.
51	B01	KY9004	68.3	1440	.192	.000	4.592	.000	.339	.000	2.	156.	0.	158.
51	B02	KY9004	68.3	1395	.180	.000	4.882	.000	.322	.000	2.	145.	0.	147.
55	251	KY 30	9.8	1282	.054	.000	3.462	.000	.432	.000	2.	38.	0.	40.
55	337	KY 30	3.0	2709	.046	.000	3.296	.000	.590	.000	5.	89.	0.	94.
55	A19	US 421	15.0	4020	.035	.000	3.118	.000	.686	.000	7.	110.	0.	117.
58	003	US 23	9.9	11000	.116	.005	4.147	5.333	.394	3.556	17.	754.	42.	813.
58	277	KY 321	2.3	5702	.063	.000	2.852	.000	.337	.000	10.	127.	0.	136.
58	518	US 460	6.0	5398	.132	.000	4.366	.000	.383	.000	9.	437.	0.	446.
58	A82	KY 321	8.6	17469	.045	.000	3.467	.000	.459	.000	30.	457.	0.	487.
58	W02	KY 321	7.8	14547	.092	.009	3.502	5.250	.425	3.983	24.	718.	92.	833.
58	W06	KY 321	8.6	17469	.093	.002	3.575	5.333	.372	3.556	28.	790.	21.	839.
58	W07	KY 40	9.6	5787	.054	.000	2.796	.000	.312	.000	10.	99.	0.	109.
60	523	KY 160	7.0	3385	.076	.000	3.468	.000	.373	.000	6.	121.	0.	127.
61	278	US 25E	6.2	13273	.123	.000	3.782	.000	.344	.000	21.	774.	0.	795.
64	A37	KY 3	15.0	8922	.168	.006	4.267	5.222	.414	4.128	13.	960.	71.	1044.
64	A64	KY 3S	.1	7919	.136	.013	4.577	5.143	.456	4.132	12.	808.	109.	929.
67	285	US 119	17.5	9186	.099	.000	3.279	.000	.418	.000	15.	454.	0.	469.
67	508	US 119	3.4	2426	.261	.000	4.807	.000	.422	.000	3.	469.	0.	472.
67	C26	KY 15	.3	12859	.072	.000	3.423	.000	.451	.000	21.	520.	0.	541.
80	750	KY 40	12.4	6637	.101	.000	3.883	.000	.380	.000	11.	359.	0.	370.
92	A02	US 231	7.4	6676	.140	.000	3.298	.000	.295	.000	10.	331.	0.	342.
92	A08	US 62	10.4	10598	.079	.001	3.948	5.000	.606	3.459	18.	730.	6.	754.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/
RURAL MINOR ARTERIAL

						FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
92	C19	KY 69	14.0	7208	.133	.000	3.160	.000	.400	.000	11.	440.	0.	452.
94	792	US 127	16.4	4111	.043	.000	3.767	.000	.389	.000	7.	94.	0.	101.
100	C06	KY 914	13.1	8380	.162	.000	3.939	.000	.276	.000	13.	537.	0.	550.
100	C13	KY 914	3.9	7343	.107	.000	4.006	.000	.384	.000	12.	442.	0.	454.
100	C17	KY 914	.8	4882	.146	.000	3.821	.000	.342	.000	8.	341.	0.	348.
100	D22	US 27	21.8	12264	.066	.000	3.555	.000	.425	.000	20.	450.	0.	470.
102	500	KY 461	3.0	6225	.145	.000	4.013	.000	.272	.000	10.	361.	0.	371.
102	A19	US 25	13.7	6771	.086	.000	3.032	.000	.452	.000	11.	292.	0.	303.
113	280	KY 56	18.0	1846	.219	.000	2.513	.000	.240	.000	3.	89.	0.	91.
113	501	US 60	.8	1638	.230	.000	2.871	.000	.448	.000	2.	177.	0.	179.
115	Z01	US 150	7.5	7203	.161	.000	4.053	.000	.278	.000	11.	475.	0.	486.
116	022	KY 90	18.0	7968	.127	.000	3.835	.000	.315	.000	13.	447.	0.	459.
117	044	KY9004	63.7	9643	.251	.000	3.675	.000	.274	.000	13.	891.	0.	904.
117	571	US 41A	6.0	2780	.242	.000	3.386	.000	.282	.000	4.	234.	0.	238.
117	A15	US 41A	.3	4648	.158	.000	3.133	.000	.276	.000	7.	232.	0.	239.
1	750	KY 61	16.2	2589	.087	.000	3.125	.000	.303	.000	4.	78.	0.	83.
1	A22	KY 55	11.2	12301	.122	.001	3.086	5.000	.287	3.459	19.	487.	6.	513.
1	A26	KY 55X	1.2	12349	.072	.000	3.392	.000	.328	.000	21.	359.	0.	380.
1	A65	KY 55	10.4	1297	.197	.000	3.468	.000	.484	.000	2.	157.	0.	159.
1	A68	KY 55	14.3	3215	.156	.000	3.326	.000	.284	.000	5.	173.	0.	178.
4	252	US 60	10.1	6174	.173	.000	4.425	.000	.410	.000	9.	707.	0.	716.
4	501	KY 121	4.0	1224	.300	.000	4.406	.000	.263	.000	2.	155.	0.	157.
4	506	US 60	4.0	3956	.281	.000	4.352	.000	.269	.000	5.	475.	0.	480.
5	250	KY 90	12.7	8578	.107	.000	3.834	.000	.332	.000	14.	427.	0.	440.
5	D35	KY 90	.7	7811	.147	.000	3.559	.000	.326	.000	11.	486.	0.	497.
6	774	KY 11	2.8	3955	.097	.000	3.214	.000	.293	.000	6.	131.	0.	138.
7	761	US 25E	11.4	18431	.131	.000	3.987	.000	.348	.000	29.	1223.	0.	1251.
9	A32	KY 627	6.0	2172	.117	.000	3.031	.000	.232	.000	3.	66.	0.	69.
11	752	US 68	8.7	1342	.074	.000	2.519	.000	.169	.000	2.	16.	0.	18.
11	770	US 150	6.0	5474	.125	.000	3.244	.000	.347	.000	9.	281.	0.	289.
13	755	KY 15	14.4	7480	.105	.000	3.639	.000	.324	.000	12.	339.	0.	351.
13	774	KY 15	19.9	6938	.095	.000	3.998	.000	.395	.000	11.	379.	0.	391.
13	784	KY 205	3.4	919	.067	.000	3.010	.000	.252	.000	2.	17.	0.	19.
13	A35	KY 15	17.3	14606	.068	.000	3.143	.000	.298	.000	25.	340.	0.	365.
14	004	US 60	27.8	5567	.139	.000	3.560	.000	.301	.000	9.	303.	0.	312.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/
RURAL MINOR ARTERIAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
14	254	KY 79	8.2	2954	.114	.000	3.068	.000	.300	.000	5.	114.	0.	118.
14	295	KY 259	6.0	2767	.111	.000	3.234	.000	.281	.000	4.	103.	0.	107.
15	514	KY 61	4.3	2225	.076	.000	2.535	.000	.202	.000	4.	31.	0.	35.
18	046	KY 80	17.1	3071	.115	.000	3.614	.000	.380	.000	5.	177.	0.	182.
19	271	US 27	1.2	7723	.108	.000	3.921	.000	.774	.000	12.	922.	0.	934.
19	301	KY 9	2.0	8707	.148	.000	4.555	.000	.325	.000	13.	697.	0.	710.
20	503	US 51	1.0	1879	.238	.000	4.152	.000	.261	.000	2.	177.	0.	179.
22	045	KY 9	3.0	4482	.305	.000	3.935	.000	.623	.000	5.	1225.	0.	1230.
22	308	KY 7	4.3	3643	.143	.000	2.876	.000	.273	.000	6.	149.	0.	155.
22	A54	KY 1	10.9	14243	.214	.004	3.680	5.077	.524	4.170	20.	2129.	100.	2250.
23	545	US 127	2.0	3326	.159	.000	3.156	.000	.272	.000	5.	165.	0.	170.
26	A51	US 421	17.7	17622	.047	.000	3.532	.000	.622	.000	30.	660.	0.	690.
26	A59	KY 80	7.4	11380	.071	.000	3.745	.000	.426	.000	19.	472.	0.	491.
27	251	US 127	7.8	8342	.135	.000	3.844	.000	.333	.000	13.	527.	0.	540.
28	A34	US 60	9.3	8835	.134	.002	3.508	5.333	.488	3.556	14.	736.	21.	771.
29	035	KY 90	20.0	3242	.122	.000	3.336	.000	.263	.000	5.	127.	0.	132.
30	764	US 60	.2	1944	.090	.000	4.149	.000	.313	.000	3.	83.	0.	86.
31	291	KY 259	9.8	6990	.059	.000	3.168	.000	.440	.000	12.	209.	0.	221.
32	508	KY 7	1.0	1196	.226	.000	2.681	.000	.295	.000	2.	78.	0.	80.
35	094	KY 11	12.7	5180	.128	.000	3.137	.000	.318	.000	8.	242.	0.	250.
36	005	US 23	9.3	18257	.132	.000	4.052	.000	.404	.000	28.	1441.	0.	1470.
36	022	KY 3	4.0	4948	.098	.000	3.260	.000	.303	.000	8.	175.	0.	183.
36	501	KY 80	1.1	6358	.186	.000	4.073	.000	.346	.000	9.	608.	0.	617.
36	755	US 23	19.7	14780	.124	.000	3.956	.000	.356	.000	23.	941.	0.	964.
36	778	KY 114	11.9	12213	.047	.000	3.077	.000	.507	.000	21.	328.	0.	349.
37	520	US 127	.1	15995	.084	.000	3.950	.000	.509	.000	26.	992.	0.	1019.
38	760	KY1099	.6	1044	.434	.000	3.712	.000	.395	.000	1.	243.	0.	244.
38	C09	KY1099	2.5	3528	.092	.000	3.257	.000	.349	.000	6.	135.	0.	141.
40	753	KY 34	.9	9426	.070	.000	3.477	.000	.444	.000	16.	369.	0.	385.
40	A07	KY 52	4.5	6306	.068	.000	2.899	.000	.490	.000	11.	223.	0.	234.
41	A05	KY 22	12.7	2653	.054	.000	3.291	.000	.922	.000	5.	159.	0.	164.
42	002	KY 58	12.0	1615	.092	.000	3.255	.000	.224	.000	3.	39.	0.	42.
42	280	KY 121	3.0	1606	.071	.000	3.372	.000	.333	.000	3.	47.	0.	49.
42	752	KY 121	16.0	3760	.189	.000	4.244	.000	.256	.000	6.	281.	0.	287.
43	560	KY9001	92.3	7170	.337	.000	4.248	.000	.276	.000	9.	1033.	0.	1041.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/
RURAL MINOR ARTERIAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
44	007	US 68	16.8	6584	.074	.000	2.904	.000	.383	.000	11.	197.	0.	208.
44	057	KY 61	9.2	5935	.089	.000	2.934	.000	.312	.000	10.	177.	0.	187.
44	286	KY 61	1.8	1284	.111	.000	3.149	.000	.361	.000	2.	60.	0.	62.
45	753	US 23	28.6	10387	.144	.000	3.988	.000	.307	.000	16.	666.	0.	682.
45	782	KY 8S	.2	6048	.208	.000	4.281	.000	.288	.000	9.	565.	0.	574.
46	A19	KY 69	13.8	8919	.081	.000	3.990	.000	.335	.000	15.	351.	0.	366.
47	153	KY1600	3.6	7698	.084	.000	2.618	.000	.270	.000	13.	166.	0.	179.
47	179	KY 313	4.1	5336	.083	.000	3.177	.000	.317	.000	9.	162.	0.	171.
47	211	KY9001	135.7	1580	.080	.000	3.240	.000	.213	.000	3.	31.	0.	34.
47	270	KY 61	2.4	12613	.112	.000	3.138	.000	.316	.000	20.	511.	0.	531.
48	008	KY 160	6.2	487	.106	.000	2.953	.000	.230	.000	1.	13.	0.	14.
48	256	US 421	3.5	1929	.126	.000	4.017	.000	.372	.000	3.	133.	0.	136.
48	286	US 421	12.6	5348	.111	.000	3.592	.000	.380	.000	9.	296.	0.	304.
49	046	US 27	9.1	3820	.071	.000	3.720	.000	.338	.000	6.	124.	0.	130.
49	529	US 62	2.0	5249	.066	.000	3.544	.000	.267	.000	9.	121.	0.	130.
51	148	US 60	17.5	3187	.060	.000	2.852	.000	.222	.000	5.	44.	0.	49.
51	156	KY9005	8.3	7855	.215	.000	4.130	.000	.260	.000	11.	663.	0.	674.
51	158	KY9004	71.3	12338	.262	.000	3.843	.000	.266	.000	16.	1205.	0.	1222.
51	556	US 41A	7.0	3790	.112	.000	2.379	.000	.187	.000	6.	69.	0.	75.
52	003	US 421	15.7	4126	.126	.000	3.352	.000	.253	.000	6.	161.	0.	168.
52	752	US 421	22.8	4894	.125	.000	3.693	.000	.355	.000	8.	293.	0.	301.
53	751	US 51	10.0	2057	.292	.000	4.491	.000	.248	.000	3.	245.	0.	247.
53	753	KY 123	19.7	132	.123	.000	2.517	.000	.191	.000	0.	3.	0.	3.
55	A17	US 421	11.6	4133	.064	.000	2.984	.000	.435	.000	7.	124.	0.	131.
57	501	US 68	5.9	9710	.067	.000	3.005	.000	.477	.000	16.	340.	0.	356.
58	001	US 23	16.3	7526	.232	.000	4.266	.000	.330	.000	10.	898.	0.	908.
58	285	KY 321	4.4	6704	.053	.000	2.882	.000	.437	.000	11.	166.	0.	177.
58	517	US 460	7.2	6053	.089	.000	3.805	.000	.353	.000	10.	265.	0.	275.
60	A26	KY 160	9.6	7577	.078	.000	3.390	.000	.399	.000	13.	291.	0.	304.
60	A29	KY 160	12.2	6956	.066	.000	3.549	.000	.360	.000	12.	213.	0.	225.
61	D40	US 25E	10.6	23292	.108	.001	4.154	5.500	.480	3.599	38.	1827.	14.	1879.
62	250	US 31E	12.4	2798	.131	.000	2.882	.000	.245	.000	4.	95.	0.	99.
62	258	KY 61	3.7	3091	.081	.000	3.131	.000	.243	.000	5.	70.	0.	75.
64	006	US 23	19.9	10200	.217	.000	4.658	.000	.343	.000	14.	1288.	0.	1302.
64	052	US 23	18.1	8771	.209	.000	4.525	.000	.354	.000	12.	1070.	0.	1082.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/
RURAL MINOR ARTERIAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
64	292	KY 645	1.3	3967	.148	.000	4.020	.000	.354	.000	6.	305.	0.	312.
65	257	KY 11	.6	2799	.094	.000	2.954	.000	.459	.000	5.	130.	0.	134.
65	A15	KY 11	4.2	9419	.073	.002	3.353	5.000	.344	3.459	16.	289.	6.	311.
67	250	US 23	.7	6994	.204	.000	3.726	.000	.389	.000	10.	756.	0.	766.
67	272	US 119	23.6	6778	.113	.000	3.611	.000	.452	.000	11.	453.	0.	464.
67	277	US 119	16.9	3052	.122	.000	3.306	.000	.353	.000	5.	159.	0.	163.
67	793	KY 15	9.1	9531	.184	.000	4.056	.000	.430	.000	14.	1118.	0.	1132.
69	778	US 127	9.6	10420	.065	.000	3.144	.000	.342	.000	18.	265.	0.	283.
69	A73	US 27	16.9	11833	.122	.000	3.237	.000	.349	.000	19.	597.	0.	616.
70	512	US 60	1.8	7683	.073	.000	3.568	.000	.543	.000	13.	397.	0.	410.
71	756	US 431	28.9	2198	.148	.000	3.555	.000	.292	.000	3.	123.	0.	127.
72	773	US 62	1.9	3825	.249	.000	4.077	.000	.279	.000	5.	396.	0.	401.
74	005	US 27	12.9	6919	.092	.000	3.179	.000	.230	.000	11.	171.	0.	182.
74	021	KY 90	8.0	680	.072	.000	2.966	.000	.198	.000	1.	10.	0.	11.
75	277	US 431	4.5	4589	.164	.000	3.235	.000	.260	.000	7.	231.	0.	238.
76	254	US 421	8.0	5850	.047	.000	2.744	.000	.533	.000	10.	146.	0.	156.
76	C60	KY 956	.2	5930	.082	.000	3.782	.000	.428	.000	10.	287.	0.	297.
76	C61	KY 956	1.1	5327	.057	.000	3.572	.000	.415	.000	9.	164.	0.	173.
78	004	US 68	19.0	2327	.068	.000	2.956	.000	.600	.000	4.	104.	0.	108.
78	565	US 68	4.0	6042	.154	.000	3.255	.000	.291	.000	9.	322.	0.	331.
79	018	US 62	11.2	5472	.205	.000	3.742	.000	.505	.000	8.	773.	0.	781.
79	504	US 641	10.6	8921	.067	.000	2.964	.000	.270	.000	15.	175.	0.	190.
79	757	US 68	10.0	13324	.082	.000	3.260	.000	.424	.000	22.	553.	0.	575.
80	003	KY 40	20.2	4904	.099	.000	3.415	.000	.464	.000	8.	281.	0.	289.
80	777	KY 645	3.0	4349	.171	.000	4.120	.000	.371	.000	7.	416.	0.	422.
81	009	KY 9	1.0	5787	.188	.000	4.203	.000	.275	.000	8.	460.	0.	468.
81	301	KY 11	3.7	5787	.103	.000	3.615	.000	.319	.000	9.	249.	0.	259.
81	517	US 68	3.3	3458	.143	.000	3.536	.000	.349	.000	5.	223.	0.	229.
82	254	KY 448	2.4	5761	.058	.000	2.788	.000	.277	.000	10.	94.	0.	104.
82	500	US 60	1.1	4665	.179	.000	3.419	.000	.402	.000	7.	419.	0.	426.
82	A33	KY 79	8.5	8421	.078	.000	2.931	.000	.320	.000	14.	226.	0.	240.
83	252	US 460	13.6	3222	.061	.000	3.003	.000	.276	.000	5.	60.	0.	65.
83	A05	US 460	9.0	5015	.052	.000	2.900	.000	.333	.000	9.	93.	0.	101.
85	050	KY9008	29.2	4380	.246	.000	4.677	.000	.237	.000	6.	435.	0.	441.
85	295	KY 90	6.7	2181	.186	.000	3.896	.000	.250	.000	3.	144.	0.	147.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/
RURAL MINOR ARTERIAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
87	280	US 460	17.2	10441	.045	.000	3.091	.000	.428	.000	18.	226.	0.	244.
87	753	KY 11	13.9	4302	.098	.000	3.347	.000	.304	.000	7.	158.	0.	165.
88	505	US 460	7.4	1635	.122	.000	2.494	.000	.212	.000	3.	38.	0.	41.
89	267	US 431	4.7	2491	.091	.000	2.760	.000	.207	.000	4.	48.	0.	52.
89	D22	US 431	18.0	8004	.099	.002	4.268	5.500	.366	3.599	13.	450.	14.	477.
90	543	US 31E	7.7	5402	.188	.000	2.592	.000	.263	.000	8.	252.	0.	260.
90	E07	US 31E	.7	4062	.115	.000	3.203	.000	.373	.000	7.	203.	0.	209.
92	504	US 231	11.8	11603	.258	.002	2.650	5.286	.295	3.543	14.	854.	48.	917.
93	253	KY 22	12.1	2263	.074	.000	2.834	.000	.266	.000	4.	46.	0.	50.
93	A38	KY 146	8.8	7789	.056	.000	2.787	.000	.410	.000	13.	181.	0.	194.
94	001	KY 22	13.3	1285	.050	.000	3.485	.000	.399	.000	2.	32.	0.	34.
96	757	US 27	14.9	6527	.106	.000	3.924	.000	.533	.000	11.	531.	0.	542.
97	251	KY 15	5.2	8965	.118	.000	3.052	.000	.277	.000	14.	326.	0.	340.
97	A50	KY 451	3.1	5980	.102	.000	2.836	.000	.282	.000	10.	178.	0.	188.
97	A83	KY 15	10.0	12173	.120	.003	3.810	5.400	.476	3.574	19.	967.	35.	1022.
97	A95	KY 15	13.4	29336	.026	.000	4.149	.000	.220	.000	52.	254.	0.	306.
98	049	US 119	23.3	15856	.085	.000	3.873	.000	.559	.000	26.	1062.	0.	1088.
98	122	US 119	17.9	9048	.154	.000	3.841	.000	.562	.000	14.	1099.	0.	1112.
98	325	US 460	10.1	7443	.144	.000	3.893	.000	.606	.000	12.	918.	0.	930.
98	506	US 460	1.9	9972	.112	.000	3.461	.000	.620	.000	16.	877.	0.	893.
98	572	US 23	.2	6130	.262	.000	3.809	.000	.372	.000	8.	829.	0.	837.
98	752	KY 194	16.9	1939	.115	.000	3.727	.000	.301	.000	3.	91.	0.	94.
98	762	KY1426	16.2	2431	.091	.000	3.149	.000	.320	.000	4.	82.	0.	86.
98	793	KY1426	9.8	7559	.082	.000	3.204	.000	.471	.000	13.	341.	0.	353.
98	812	US 119	9.0	7831	.165	.000	3.898	.000	.528	.000	12.	971.	0.	983.
99	C03	KY 15	3.6	9098	.042	.000	3.173	.000	.346	.000	16.	151.	0.	167.
100	055	KY 461	4.3	6405	.200	.000	3.530	.000	.421	.000	9.	695.	0.	704.
100	C13	KY 914	3.9	6422	.066	.000	3.132	.000	.288	.000	11.	139.	0.	150.
100	C17	KY 914	.8	4882	.154	.000	3.974	.000	.300	.000	8.	328.	0.	335.
100	D22	US 27	21.6	10167	.074	.000	3.157	.000	.322	.000	17.	279.	0.	296.
103	752	KY 32	.9	4762	.098	.000	3.640	.000	.346	.000	8.	214.	0.	222.
105	516	US 62	.4	7170	.096	.000	3.583	.000	.369	.000	12.	332.	0.	344.
108	753	KY 44	7.9	4824	.114	.000	2.901	.000	.344	.000	8.	200.	0.	208.
108	767	KY 44	3.6	2602	.100	.000	2.694	.000	.390	.000	4.	100.	0.	104.
108	783	KY 55	7.8	10590	.090	.000	2.894	.000	.229	.000	17.	231.	0.	248.

2013

AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/
RURAL MINOR ARTERIAL

						FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
109	061	US 68	8.2	8584	.104	.000	3.328	.000	.303	.000	14.	328.	0.	342.
109	782	KY 210	1.3	3857	.159	.000	3.841	.000	.306	.000	6.	263.	0.	269.
110	500	US 68	5.4	5626	.178	.000	3.767	.000	.387	.000	8.	531.	0.	539.
111	005	US 68	24.8	7211	.102	.000	3.897	.000	.295	.000	12.	308.	0.	319.
111	500	US 68	16.0	6307	.112	.000	3.561	.000	.299	.000	10.	275.	0.	285.
112	754	US 421	19.1	8289	.069	.000	3.376	.000	.367	.000	14.	258.	0.	272.
113	754	US 60	15.6	4991	.105	.015	3.610	5.250	.441	4.208	8.	301.	65.	374.
114	003	KY 101	8.3	4738	.081	.000	3.182	.000	.357	.000	8.	160.	0.	168.
115	027	KY 555	10.6	3537	.188	.000	3.899	.000	.293	.000	5.	277.	0.	282.
115	500	KY 55	1.5	8552	.170	.000	3.925	.000	.357	.000	13.	742.	0.	755.
115	501	US 150	12.0	2667	.179	.000	3.571	.000	.282	.000	4.	176.	0.	180.
115	507	US 150	8.1	9038	.148	.000	3.899	.000	.447	.000	14.	850.	0.	864.
116	506	KY 90	1.7	5906	.124	.000	3.718	.000	.270	.000	9.	269.	0.	278.
118	752	KY 90	7.9	1980	.104	.000	2.920	.000	.281	.000	3.	62.	0.	65.
118	781	US 25W	25.5	7745	.114	.000	2.554	.000	.273	.000	12.	224.	0.	237.
118	B23	US 25W	28.6	11903	.056	.000	3.258	.000	.525	.000	20.	416.	0.	436.
119	755	KY 11	1.5	1495	.140	.000	2.612	.000	.279	.000	2.	55.	0.	58.
120	C10	US 421	1.5	4696	.075	.000	3.174	.000	.560	.000	8.	228.	0.	236.

SUMMARY OF AVERAGE VALUES FOR
AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/
RURAL MINOR ARTERIAL

YEAR	3 YR AVG	13	11	07	05	04	01	00	99	98	97	96
UNCLASSIFIED ROADS (ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)												
NO OF CLASSIFICATION STA.	1150	368	546	236	246	238	135	30	46	70	57	79
AADT	6722	6557	6688	7057	7749	7081	7545	6242	7879	7603	8954	6439
PERCENT TRUCKS	13.193	12.294	13.241	14.482	14.228	13.752	16.023	18.686	12.956	11.839	10.003	13.718
AXLES PER TRUCK	3.504	3.479	3.473	3.617	3.505	3.591	3.577	3.637	3.603	3.532	3.504	3.522
EAL'S PER TRUCK AXLE	.291	.346	.260	.278	.270	.236	.283	.286	.284	.275	.271	.248
CLASSIFIED ROADS (MANUAL LOCATION WITH 3% OR MORE OF TRUCKS CLASSIFIED AS HEAVY/COAL)												
NO OF CLASSIFICATION STA.	0	0	0	0	0	1	5	10	13	25	25	25
AADT	0	0	0	0	0	13513	10440	8404	13147	10268	11012	8449
PERCENT TRUCKS	.000	.000	.000	.000	.000	5.715	12.956	15.789	17.028	17.769	16.283	13.686
PERCENT OF TRUCKS CLASSIFIED AS HEAVY/COAL	.000	.000	.000	.000	.000	18.139	35.328	28.408	37.977	37.125	30.049	29.687
AXLES PER TRUCK NORMAL	.000	.000	.000	.000	.000	3.229	3.614	3.514	3.451	3.384	3.615	3.229
AXLES PER TRUCK HEAVY/COAL	.000	.000	.000	.000	.000	5.179	5.112	4.978	5.010	5.188	5.174	4.764
EAL'S PER TRUCK AXLE NORMAL	.000	.000	.000	.000	.000	.218	.314	.316	.305	.290	.259	.250
EAL'S PER TRUCK AXLE HEAVY/COAL	.000	.000	.000	.000	.000	3.264	3.304	3.296	3.240	3.267	2.940	2.994

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

						FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
1	001	KY 76	12.0	1034	.059	.000	2.952	.000	.464	.000	2.	30.	0.	32.
2	788	KY 101	1.0	1891	.088	.000	2.924	.000	.409	.000	3.	73.	0.	76.
2	A03	KY3499	.0	2607	.060	.000	3.290	.000	.409	.000	4.	77.	0.	81.
2	A73	KY 100	13.0	4630	.078	.000	3.256	.000	.439	.000	8.	188.	0.	196.
4	005	KY 358	10.0	740	.095	.000	2.580	.000	.361	.000	1.	24.	0.	25.
4	011	KY 473	9.0	815	.093	.000	3.262	.000	.405	.000	1.	37.	0.	38.
6	269	KY 36	17.0	2099	.086	.000	2.601	.000	.408	.000	3.	70.	0.	74.
6	A15	US 60	6.0	3943	.086	.000	2.882	.000	.511	.000	6.	183.	0.	190.
7	002	KY 66	12.0	279	.118	.000	4.672	.000	.352	.000	0.	20.	0.	20.
7	004	KY 72	.0	1765	.072	.000	4.710	.000	.369	.000	3.	81.	0.	83.
7	005	KY 987	5.0	558	.030	.000	2.649	.000	.568	.000	1.	9.	0.	10.
7	022	KY 221	9.0	1024	.189	.000	4.145	.000	.496	.000	2.	146.	0.	147.
7	751	KY 190	7.0	798	.034	.000	2.665	.000	.384	.000	1.	10.	0.	11.
8	053	KY 20	13.0	2298	.074	.000	2.504	.000	.405	.000	4.	63.	0.	67.
8	255	KY 536	8.0	4315	.035	.000	2.949	.000	.617	.000	7.	101.	0.	108.
8	502	US 25	.0	5596	.177	.000	2.985	.000	.577	.000	8.	621.	0.	629.
9	015	KY1879	.0	772	.043	.000	2.410	.000	.393	.000	1.	11.	0.	13.
10	750	KY 3	7.0	3775	.062	.000	2.478	.000	.461	.000	6.	98.	0.	104.
11	500	KY 34	6.0	1798	.107	.000	2.227	.000	.370	.000	3.	58.	0.	61.
12	252	KY 10	19.0	1967	.074	.000	2.729	.000	.454	.000	3.	66.	0.	69.
12	512	KY 22	1.0	427	.077	.000	2.627	.000	.382	.000	1.	12.	0.	13.
13	001	KY 30	33.0	287	.098	.000	3.260	.000	.361	.000	0.	12.	0.	12.
13	013	KY 30	19.0	668	.072	.000	3.607	.000	.351	.000	1.	22.	0.	23.
13	256	KY 476	8.0	1542	.097	.000	4.903	.000	.322	.000	3.	86.	0.	89.
13	784	KY 205	2.0	754	.037	.000	2.969	.000	.355	.000	1.	11.	0.	12.
14	543	KY 105	.0	1035	.089	.000	2.969	.000	.422	.000	2.	42.	0.	44.
14	A01	KY 259	9.0	1993	.063	.000	2.795	.000	.411	.000	3.	53.	0.	56.
14	A05	KY 259	8.0	2265	.056	.000	2.730	.000	.404	.000	4.	51.	0.	55.
15	A53	KY 480	1.0	8217	.071	.000	2.557	.000	.473	.000	14.	259.	0.	273.
16	032	KY 70	19.0	1121	.086	.000	3.077	.000	.375	.000	2.	40.	0.	42.
16	A40	US 231	9.0	5997	.075	.000	2.680	.000	.501	.000	10.	221.	0.	231.
17	510	KY 139	12.0	1666	.080	.000	2.843	.000	.433	.000	3.	60.	0.	63.
17	511	KY 91	14.0	2375	.087	.000	3.727	.000	.490	.000	4.	137.	0.	141.
17	548	KY 293	9.0	1549	.064	.000	2.843	.000	.436	.000	3.	45.	0.	47.
18	646	KY 299	4.0	1538	.067	.000	2.774	.000	.446	.000	3.	47.	0.	49.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
19	009	KY 10	4.0	640	.052	.000	2.290	.000	.480	.000	1.	13.	0.	14.
19	801	KY 915	2.0	1009	.133	.000	2.244	.000	.420	.000	2.	46.	0.	48.
20	005	KY 307	3.0	1182	.092	.000	2.829	.000	.409	.000	2.	46.	0.	48.
20	023	KY1377	4.0	560	.061	.000	3.422	.000	.373	.000	1.	16.	0.	17.
20	046	KY 849	.0	202	.124	.000	2.701	.000	.383	.000	0.	9.	0.	10.
21	753	US 42	4.0	2380	.097	.000	3.313	.000	.567	.000	4.	158.	0.	162.
22	001	KY 7	12.0	506	.026	.000	2.398	.000	.364	.000	1.	4.	0.	5.
22	012	US 60	31.0	3453	.039	.000	2.692	.000	.457	.000	6.	61.	0.	67.
22	754	KY 2	13.0	1005	.208	.000	3.675	.000	.677	.000	1.	190.	0.	191.
23	550	KY 910	3.0	900	.103	.000	3.055	.000	.379	.000	1.	39.	0.	41.
23	A16	KY 70	.0	1182	.057	.000	2.629	.000	.429	.000	2.	28.	0.	30.
23	A64	KY 70	12.0	4002	.058	.000	2.970	.000	.471	.000	7.	119.	0.	126.
24	067	KY 189	7.0	531	.171	.000	2.463	.000	.359	.000	1.	29.	0.	30.
24	340	KY 813	.0	121	.471	.000	3.615	.000	.332	.000	0.	25.	0.	25.
24	D13	KY 109	29.0	1235	.269	.000	2.712	.000	.451	.000	2.	148.	0.	150.
25	032	US 60	11.0	763	.085	.000	2.774	.000	.562	.000	1.	37.	0.	38.
25	252	KY 15	1.0	1336	.073	.000	2.588	.000	.399	.000	2.	37.	0.	39.
26	020	KY 11	10.0	2136	.032	.000	2.955	.000	.488	.000	4.	36.	0.	40.
26	046	KY 66	.0	1781	.094	.000	4.631	.000	.357	.000	3.	101.	0.	104.
26	254	KY 66	10.0	643	.107	.000	2.409	.000	.369	.000	1.	22.	0.	23.
26	260	US 421	11.0	3945	.073	.000	2.874	.000	.438	.000	7.	132.	0.	139.
26	509	US 421	16.0	12603	.076	.000	3.531	.000	.492	.000	21.	606.	0.	627.
28	017	KY 365	.0	738	.075	.000	2.570	.000	.381	.000	1.	20.	0.	21.
30	317	US 231	.0	4633	.033	.000	3.064	.000	.422	.000	8.	73.	0.	81.
31	004	KY 728	9.0	214	.112	.000	2.874	.000	.385	.000	0.	10.	0.	10.
32	504	KY 32	9.0	1199	.054	.000	2.760	.000	.383	.000	2.	25.	0.	27.
33	018	KY1571	5.0	1287	.054	.000	2.386	.000	.367	.000	2.	22.	0.	24.
33	511	KY 89	.0	497	.068	.000	2.793	.000	.373	.000	1.	13.	0.	14.
33	763	KY 89	18.0	1632	.064	.000	2.464	.000	.370	.000	3.	35.	0.	37.
33	A18	KY 89	11.0	3587	.037	.000	2.627	.000	.375	.000	6.	47.	0.	53.
33	A55	KY 89	12.0	7561	.097	.000	2.945	.000	.384	.000	12.	302.	0.	315.
34	270	KY 859	.0	4397	.076	.000	2.692	.000	.519	.000	7.	171.	0.	178.
34	367	US 25	.0	3321	.052	.000	2.636	.000	.641	.000	6.	107.	0.	113.
34	567	KY1969	.0	860	.119	.000	2.499	.000	.364	.000	1.	34.	0.	35.
36	010	KY 194	.0	2036	.125	.000	4.222	.000	.384	.000	3.	150.	0.	153.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
36	019	KY3051	2.0	940	.046	.000	2.489	.000	.449	.000	2.	18.	0.	19.
36	321	KY 979	14.0	3290	.047	.000	2.854	.000	.422	.000	6.	69.	0.	74.
36	509	KY 680	3.0	3643	.075	.000	3.729	.000	.400	.000	6.	150.	0.	156.
36	752	KY 404	4.0	1989	.037	.000	2.814	.000	.404	.000	3.	30.	0.	34.
36	C05	KY1428	.0	5002	.093	.000	3.023	.000	.525	.000	8.	270.	0.	279.
38	A91	US 45	1.0	4622	.114	.000	3.256	.000	.362	.000	7.	227.	0.	234.
39	004	US 127	3.0	1936	.137	.000	3.130	.000	.456	.000	3.	138.	0.	141.
39	504	KY 467	.0	533	.152	.000	2.289	.000	.378	.000	1.	26.	0.	26.
40	756	KY 152	.0	1438	.054	.000	2.289	.000	.373	.000	2.	24.	0.	27.
41	501	KY 36	5.0	1005	.087	.000	2.676	.000	.451	.000	2.	38.	0.	40.
42	029	KY 131	4.0	1589	.047	.000	4.108	.000	.459	.000	3.	52.	0.	54.
42	307	KY 381	.0	668	.138	.000	2.829	.000	.367	.000	1.	35.	0.	36.
42	510	US 45	10.0	3946	.090	.000	3.626	.000	.411	.000	6.	194.	0.	200.
42	539	KY 129	6.0	799	.180	.000	3.542	.000	.591	.000	1.	110.	0.	111.
43	254	KY 54	16.0	3373	.070	.000	2.697	.000	.446	.000	6.	104.	0.	110.
43	265	KY 224	5.0	878	.112	.000	3.316	.000	.394	.000	1.	47.	0.	48.
43	502	KY 185	7.0	1747	.106	.000	2.696	.000	.373	.000	3.	68.	0.	71.
43	761	KY 631	.0	286	.112	.000	2.505	.000	.379	.000	0.	11.	0.	12.
43	764	KY 79	18.0	1441	.111	.000	2.907	.000	.432	.000	2.	73.	0.	76.
44	251	US 68	6.0	2369	.097	.000	3.115	.000	.394	.000	4.	103.	0.	107.
45	500	KY 2	.0	776	.153	.000	3.107	.000	.581	.000	1.	78.	0.	80.
46	509	KY 69	.0	1301	.133	.000	3.579	.000	.373	.000	2.	84.	0.	86.
47	342	KY 84	26.0	3626	.117	.000	3.422	.000	.415	.000	6.	220.	0.	226.
48	003	KY 160	13.0	1037	.074	.000	2.985	.000	.459	.000	2.	39.	0.	40.
48	251	KY 38	17.0	1189	.049	.000	2.512	.000	.378	.000	2.	20.	0.	22.
48	A06	KY 38	.0	6593	.055	.000	2.750	.000	.512	.000	11.	187.	0.	199.
49	500	KY 353	.0	1229	.055	.000	2.490	.000	.426	.000	2.	26.	0.	28.
50	036	US 31	13.0	2151	.063	.000	3.115	.000	.421	.000	4.	65.	0.	68.
50	067	KY 357	4.0	686	.086	.000	2.674	.000	.439	.000	1.	25.	0.	26.
50	A42	US 31	.0	6421	.081	.000	3.606	.000	.432	.000	11.	295.	0.	306.
51	817	KY 359	.0	1504	.116	.000	3.115	.000	.393	.000	2.	78.	0.	80.
52	001	KY 389	20.0	295	.129	.000	3.075	.000	.374	.000	0.	16.	0.	16.
52	006	KY 22	22.0	783	.172	.000	3.277	.000	.935	.000	1.	151.	0.	152.
52	262	KY 241	.0	1865	.058	.000	2.745	.000	.425	.000	3.	46.	0.	49.
52	522	KY 146	2.0	3318	.089	.000	3.030	.000	.487	.000	5.	159.	0.	165.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
52	763	KY 55	4.0	1013	.095	.000	3.073	.000	.693	.000	2.	75.	0.	76.
53	253	KY 307	11.0	894	.154	.000	3.762	.000	.464	.000	1.	88.	0.	89.
53	253	KY 58	17.0	1191	.154	.000	3.768	.000	.464	.000	2.	117.	0.	119.
54	002	KY 260	2.0	2042	.135	.000	2.553	.000	.403	.000	3.	104.	0.	107.
54	374	KY2171	2.0	2241	.129	.000	3.597	.000	.383	.000	4.	146.	0.	149.
56	369	KY 148	1.0	1204	.052	.000	2.483	.000	.436	.000	2.	25.	0.	27.
58	004	KY 201	.0	3128	.051	.000	3.011	.000	.400	.000	5.	70.	0.	76.
58	033	KY 581	1.0	1432	.044	.000	2.645	.000	.434	.000	2.	26.	0.	29.
58	254	KY1428	1.0	4632	.061	.000	2.848	.000	.500	.000	8.	147.	0.	155.
58	501	KY 40	7.0	9808	.063	.000	2.911	.000	.545	.000	17.	358.	0.	375.
58	A24	KY 40	11.0	5904	.064	.000	3.007	.000	.625	.000	10.	261.	0.	271.
59	275	KY 17	1.0	2855	.097	.000	2.663	.000	.464	.000	5.	124.	0.	129.
60	006	KY 550	24.0	1541	.180	.000	4.959	.000	.339	.000	2.	171.	0.	173.
60	009	KY1087	19.0	322	.031	.000	2.572	.000	.364	.000	1.	3.	0.	4.
60	012	KY 550	15.0	1014	.041	.000	3.485	.000	.408	.000	2.	22.	0.	24.
62	004	KY 84	9.0	844	.113	.000	2.957	.000	.374	.000	1.	38.	0.	40.
62	505	KY 224	.0	1634	.065	.000	3.360	.000	.447	.000	3.	59.	0.	61.
62	512	US 31	.0	1856	.089	.000	2.982	.000	.392	.000	3.	70.	0.	73.
63	013	KY 472	2.0	2547	.023	.000	3.259	.000	.480	.000	4.	34.	0.	38.
63	805	KY3094	1.0	1796	.061	.000	3.312	.000	.459	.000	3.	61.	0.	64.
64	005	KY 1	.0	347	.066	.000	2.189	.000	.369	.000	1.	7.	0.	7.
64	281	KY 644	1.0	1273	.052	.000	2.485	.000	.366	.000	2.	22.	0.	24.
65	256	KY 587	9.0	1623	.047	.000	2.851	.000	.362	.000	3.	29.	0.	32.
66	003	KY 699	3.0	1226	.069	.000	2.228	.000	.406	.000	2.	28.	0.	30.
66	258	KY 221	.0	197	.259	.000	4.583	.000	.315	.000	0.	27.	0.	27.
67	008	KY 317	1.0	2283	.041	.000	2.756	.000	.474	.000	4.	44.	0.	48.
67	252	KY2034	.0	6149	.027	.000	3.090	.000	.660	.000	11.	123.	0.	134.
67	270	KY 805	.0	3168	.034	.000	2.928	.000	.397	.000	6.	46.	0.	51.
67	C09	KY 15	2.0	5170	.082	.000	3.510	.000	.575	.000	9.	312.	0.	321.
68	257	KY 59	7.0	432	.074	.000	2.825	.000	.367	.000	1.	12.	0.	13.
68	A25	KY 8	13.0	2671	.040	.000	2.832	.000	.459	.000	5.	51.	0.	56.
69	003	KY 78	14.0	645	.053	.000	3.200	.000	.390	.000	1.	15.	0.	17.
72	A09	KY 93	14.0	3409	.049	.000	3.352	.000	.436	.000	6.	90.	0.	96.
73	845	KY 305	4.0	1115	.117	.000	2.794	.000	.383	.000	2.	51.	0.	53.
74	296	KY 92	21.0	1681	.116	.000	3.602	.000	.378	.000	3.	97.	0.	100.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

						FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
75	005	KY 250	6.0	732	.051	.000	3.401	.000	.339	.000	1.	16.	0.	17.
75	275	KY 85	11.0	1684	.303	.000	4.571	.000	.361	.000	2.	308.	0.	310.
79	802	KY1523	.0	1714	.174	.000	3.503	.000	.399	.000	3.	152.	0.	155.
79	C07	KY 95	4.0	6454	.117	.000	3.562	.000	.401	.000	10.	395.	0.	405.
80	253	KY1714	10.0	1663	.079	.000	3.623	.000	.374	.000	3.	65.	0.	68.
81	266	KY1237	2.0	337	.039	.000	2.469	.000	.391	.000	1.	5.	0.	5.
82	263	KY 333	3.0	960	.114	.000	2.586	.000	.405	.000	2.	42.	0.	43.
83	011	KY1274	10.0	852	.086	.000	3.067	.000	.442	.000	1.	36.	0.	38.
83	250	KY 746	3.0	968	.076	.000	2.778	.000	.381	.000	2.	29.	0.	30.
83	263	KY1569	2.0	174	.086	.000	2.237	.000	.384	.000	0.	5.	0.	5.
83	502	KY 77	.0	318	.053	.000	2.563	.000	.506	.000	1.	8.	0.	9.
84	570	KY 152	2.0	671	.051	.000	2.480	.000	.377	.000	1.	12.	0.	13.
86	266	KY 163	.0	1410	.129	.000	3.702	.000	.383	.000	2.	94.	0.	96.
86	A01	KY 163	10.0	5561	.117	.000	2.916	.000	.438	.000	9.	303.	0.	312.
87	509	US 60	.0	4733	.068	.000	2.855	.000	.451	.000	8.	151.	0.	159.
88	521	KY1010	.0	461	.104	.000	2.577	.000	.380	.000	1.	17.	0.	18.
88	A46	KY 191	13.0	4331	.074	.000	2.335	.000	.405	.000	7.	111.	0.	118.
89	002	US 62	19.0	2452	.098	.000	3.195	.000	.506	.000	4.	142.	0.	146.
89	258	KY 176	6.0	1965	.064	.000	2.963	.000	.428	.000	3.	58.	0.	61.
89	796	KY 70	10.0	2448	.088	.000	4.090	.000	.367	.000	4.	118.	0.	122.
89	A41	US 62	9.0	5336	.076	.000	3.126	.000	.713	.000	9.	330.	0.	339.
90	008	US 62	25.0	4055	.108	.000	2.900	.000	.514	.000	7.	237.	0.	244.
90	075	KY 555	.0	685	.130	.000	3.276	.000	.407	.000	1.	43.	0.	44.
90	261	US 62	18.0	3591	.091	.000	2.735	.000	.544	.000	6.	178.	0.	183.
90	277	KY 49	2.0	2492	.074	.000	2.774	.000	.420	.000	4.	78.	0.	82.
91	254	KY 36	8.0	1243	.080	.000	2.641	.000	.421	.000	2.	40.	0.	42.
91	750	KY 32	3.0	904	.063	.000	2.786	.000	.445	.000	2.	26.	0.	27.
91	A06	KY 32	8.0	7670	.073	.000	2.937	.000	.516	.000	13.	310.	0.	323.
92	500	US 231	16.0	4002	.054	.000	3.027	.000	.412	.000	7.	98.	0.	105.
92	511	US 62	2.0	1915	.073	.000	2.803	.000	.395	.000	3.	57.	0.	60.
94	007	KY 36	7.0	547	.115	.000	2.691	.000	.396	.000	1.	24.	0.	25.
94	752	KY 22	6.0	1339	.075	.000	2.601	.000	.607	.000	2.	58.	0.	60.
94	A04	KY 22	9.0	2669	.055	.000	2.628	.000	.483	.000	5.	69.	0.	73.
95	002	KY 30	12.0	2058	.045	.000	3.247	.000	.364	.000	4.	40.	0.	43.
95	A16	KY 11	12.0	2124	.075	.000	3.069	.000	.367	.000	4.	66.	0.	69.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

						FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
96	015	KY 159	11.0	950	.139	.000	3.048	.000	.723	.000	1.	106.	0.	108.
96	799	KY 17	7.0	1602	.118	.000	2.801	.000	.702	.000	3.	136.	0.	138.
97	004	KY 476	8.0	1697	.144	.000	4.214	.000	.357	.000	3.	134.	0.	137.
97	028	KY 80	5.0	2370	.060	.000	3.870	.000	.358	.000	4.	72.	0.	76.
97	029	KY 451	4.0	1606	.038	.000	2.878	.000	.358	.000	3.	23.	0.	26.
97	252	KY 7	.0	4321	.060	.000	4.017	.000	.357	.000	7.	136.	0.	143.
97	757	KY 28	1.0	788	.037	.000	2.322	.000	.374	.000	1.	9.	0.	11.
97	A01	KY 476	.0	4685	.037	.000	2.442	.000	.383	.000	8.	60.	0.	68.
97	A80	KY 451	.0	1775	.027	.000	2.310	.000	.357	.000	3.	14.	0.	18.
97	A87	KY 451	.0	3231	.063	.000	2.696	.000	.364	.000	6.	72.	0.	78.
98	087	KY1056	11.0	1938	.085	.000	3.672	.000	.372	.000	3.	82.	0.	85.
98	111	KY 194	.0	1425	.199	.000	4.786	.000	.352	.000	2.	175.	0.	177.
98	253	KY1499	2.0	1642	.324	.000	4.493	.000	.367	.000	2.	320.	0.	322.
98	591	KY1469	11.0	4491	.056	.000	3.652	.000	.397	.000	8.	133.	0.	140.
98	752	KY 194	17.0	2076	.105	.000	3.044	.000	.414	.000	3.	100.	0.	103.
98	793	KY1426	10.0	6523	.060	.000	3.009	.000	.455	.000	11.	196.	0.	207.
100	087	KY 39	2.0	4684	.043	.000	2.742	.000	.490	.000	8.	98.	0.	106.
100	092	KY2227	2.0	2072	.092	.000	2.983	.000	.397	.000	3.	82.	0.	85.
101	507	US 62	5.0	409	.186	.000	3.044	.000	.946	.000	1.	80.	0.	80.
102	275	US 25	5.0	1512	.071	.000	3.311	.000	.505	.000	2.	65.	0.	68.
103	010	KY 377	7.0	1476	.083	.000	2.948	.000	.519	.000	2.	68.	0.	71.
104	006	KY 558	.0	199	.101	.000	3.424	.000	.500	.000	0.	12.	0.	13.
104	256	KY 92	9.0	1247	.106	.000	3.090	.000	.413	.000	2.	62.	0.	64.
104	A26	KY 80	3.0	5911	.064	.000	3.048	.000	.535	.000	10.	225.	0.	235.
105	006	US 25	15.0	975	.146	.000	2.290	.000	.375	.000	2.	45.	0.	46.
105	010	US 25	6.0	6430	.054	.000	2.498	.000	.397	.000	11.	126.	0.	137.
107	267	KY 73	.0	398	.058	.000	2.947	.000	.384	.000	1.	10.	0.	10.
108	005	KY 55	.0	2137	.082	.000	2.776	.000	.453	.000	4.	81.	0.	84.
108	005	KY 44	11.0	6747	.083	.000	2.768	.000	.463	.000	11.	261.	0.	272.
108	019	KY 44	15.0	2110	.075	.000	2.581	.000	.429	.000	4.	64.	0.	67.
109	790	KY 323	6.0	880	.045	.000	2.488	.000	.369	.000	2.	13.	0.	15.
109	822	KY 527	1.0	1971	.046	.000	2.390	.000	.489	.000	3.	39.	0.	42.
110	021	KY 181	16.0	1359	.110	.000	2.957	.000	.412	.000	2.	66.	0.	68.
110	511	KY 104	.0	1053	.072	.000	3.227	.000	.442	.000	2.	40.	0.	41.
111	255	KY 164	18.0	659	.146	.000	2.869	.000	.655	.000	1.	66.	0.	67.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

						FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
113	A10	KY 365	.0	1055	.084	.000	2.390	.000	.380	.000	2.	30.	0.	31.
114	005	KY 101	3.0	1736	.116	.000	3.114	.000	.400	.000	3.	92.	0.	94.
114	009	US 68	15.0	3276	.173	.000	3.398	.000	.512	.000	5.	360.	0.	365.
114	306	KY1402	1.0	2701	.144	.000	2.775	.000	.427	.000	4.	168.	0.	172.
114	508	US 31	1.0	6614	.109	.000	3.287	.000	.538	.000	11.	465.	0.	476.
115	012	KY 53	3.0	349	.037	.000	2.872	.000	.394	.000	1.	5.	0.	6.
115	A17	KY 528	2.0	3130	.040	.000	2.596	.000	.496	.000	5.	59.	0.	64.
116	256	KY 92	16.0	1173	.066	.000	2.879	.000	.424	.000	2.	34.	0.	36.
117	005	KY 56	12.0	3398	.253	.000	4.054	.000	.363	.000	5.	462.	0.	467.
117	007	US 41	6.0	2095	.205	.000	3.687	.000	.872	.000	3.	505.	0.	508.
118	019	KY1193	5.0	349	.014	.000	3.712	.000	.885	.000	1.	6.	0.	7.
118	308	US 25	.0	3867	.057	.000	3.274	.000	.409	.000	7.	108.	0.	115.
119	033	KY 191	10.0	1085	.076	.000	2.557	.000	.419	.000	2.	32.	0.	34.
.20	006	US 62	7.0	3578	.053	.000	2.444	.000	.386	.000	6.	66.	0.	72.
8	289	US 42	9.2	8306	.079	.000	3.703	.000	.749	.000	14.	663.	0.	677.
8	750	KY 20	8.0	1649	.103	.000	3.003	.000	.689	.000	3.	128.	0.	131.
8	772	KY 8	.4	2794	.107	.000	3.150	.000	.739	.000	4.	252.	0.	257.
8	776	KY 18	7.7	2484	.128	.000	2.854	.000	.814	.000	4.	269.	0.	273.
17	296	US 62	11.6	1297	.129	.000	2.426	.000	.373	.000	2.	55.	0.	57.
19	778	KY 536	3.4	3346	.072	.000	2.600	.000	.577	.000	6.	131.	0.	136.
21	750	KY 55	7.7	1395	.105	.000	3.236	.000	.737	.000	2.	127.	0.	129.
21	752	KY 36	8.0	3658	.055	.000	3.347	.000	.841	.000	6.	207.	0.	213.
24	325	KY 115	6.0	1509	.255	.000	2.659	.000	.391	.000	2.	146.	0.	148.
24	529	KY 164	5.2	1025	.259	.000	2.334	.000	.385	.000	1.	87.	0.	88.
24	800	KY 91	4.8	2849	.259	.000	2.289	.000	.378	.000	4.	233.	0.	236.
29	012	KY 61	22.6	1131	.149	.000	4.063	.000	.471	.000	2.	117.	0.	119.
36	250	KY 680	13.2	7455	.074	.000	3.393	.000	.438	.000	12.	300.	0.	313.
36	256	KY 122	25.4	2564	.064	.000	3.165	.000	.402	.000	4.	76.	0.	80.
36	504	KY 550	2.2	1873	.054	.000	2.723	.000	.457	.000	3.	46.	0.	49.
36	514	KY 122	8.4	876	.027	.000	2.828	.000	.378	.000	1.	10.	0.	11.
36	517	KY 850	.2	1346	.051	.000	2.751	.000	.360	.000	2.	25.	0.	27.
36	523	KY 7	2.5	1937	.129	.000	3.861	.000	.344	.000	3.	121.	0.	124.
36	A15	KY1428	16.4	6210	.063	.000	2.983	.000	.488	.000	10.	207.	0.	217.
39	535	KY1039	.1	292	.054	.000	2.871	.000	.384	.000	0.	6.	0.	7.
39	751	US 42	6.7	3974	.076	.000	3.349	.000	.545	.000	7.	202.	0.	208.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
41	260	KY 330	2.1	2437	.057	.000	2.681	.000	.427	.000	4.	58.	0.	62.
41	A08	US 25	9.1	4107	.042	.000	2.821	.000	.491	.000	7.	86.	0.	93.
46	515	KY 144	1.0	956	.114	.000	2.629	.000	.402	.000	1.	42.	0.	44.
51	006	US 41	9.0	4722	.143	.000	2.631	.000	.391	.000	7.	253.	0.	260.
54	021	KY 138	.9	1390	.197	.000	2.565	.000	.359	.000	2.	92.	0.	94.
54	087	KY 260	1.6	3839	.098	.000	3.111	.000	.466	.000	6.	198.	0.	204.
54	254	US 41	6.6	4131	.142	.000	2.349	.000	.406	.000	6.	203.	0.	210.
54	266	US 41	.8	1177	.232	.000	3.147	.000	.373	.000	2.	117.	0.	119.
54	341	US 41	4.6	3585	.060	.000	3.022	.000	.376	.000	6.	89.	0.	95.
54	801	KY1069	2.5	1937	.089	.000	2.273	.000	.404	.000	3.	58.	0.	61.
54	D10	KY 109	3.2	4417	.077	.000	2.943	.000	.372	.000	7.	137.	0.	144.
58	015	KY 581	7.3	221	.062	.000	2.071	.000	.350	.000	0.	4.	0.	4.
58	291	KY1428	.3	6598	.070	.000	2.936	.000	.485	.000	11.	240.	0.	252.
58	502	KY 40	1.9	1549	.077	.000	2.886	.000	.360	.000	3.	45.	0.	48.
58	750	KY 172	12.2	3654	.068	.000	3.196	.000	.406	.000	6.	118.	0.	125.
58	A18	KY 321X	.1	5290	.066	.000	3.047	.000	.483	.000	9.	189.	0.	198.
58	A78	KY 40	12.3	6647	.058	.000	2.865	.000	.516	.000	11.	208.	0.	220.
59	011	KY 177	11.0	1750	.062	.000	2.418	.000	.435	.000	3.	41.	0.	44.
60	256	KY 7	4.9	1484	.183	.000	3.117	.000	.520	.000	2.	161.	0.	163.
60	257	KY1498	.1	587	.054	.000	2.795	.000	.459	.000	1.	15.	0.	16.
60	524	KY 899	.1	6707	.058	.000	3.259	.000	.480	.000	11.	221.	0.	233.
60	753	KY 550	.6	1309	.039	.000	2.728	.000	.357	.000	2.	18.	0.	21.
61	553	KY 11	6.3	1846	.055	.000	3.524	.000	.412	.000	3.	54.	0.	57.
61	782	KY 830	4.6	953	.042	.000	3.327	.000	.501	.000	2.	25.	0.	27.
61	B71	KY1232	1.5	2276	.019	.000	3.129	.000	.625	.000	4.	31.	0.	35.
63	798	US 25	17.2	3667	.109	.000	2.945	.000	.449	.000	6.	193.	0.	199.
64	253	KY1760	.2	620	.087	.000	3.152	.000	.355	.000	1.	22.	0.	23.
64	264	KY 32	23.5	1702	.075	.000	2.835	.000	.483	.000	3.	63.	0.	66.
64	504	KY 201	.2	720	.055	.000	3.079	.000	.370	.000	1.	17.	0.	18.
64	A05	KY2565	2.3	9782	.120	.000	3.299	.000	.403	.000	16.	570.	0.	585.
67	007	KY 343	.3	2509	.053	.000	2.960	.000	.443	.000	4.	65.	0.	69.
67	019	KY 805	7.2	1867	.077	.000	3.131	.000	.627	.000	3.	103.	0.	106.
67	253	KY 15X	.1	4768	.067	.008	3.251	4.333	.577	2.423	8.	216.	11.	236.
67	A16	KY 805	9.4	3974	.071	.000	3.129	.000	.601	.000	7.	194.	0.	201.
67	C20	KY 931	10.3	2721	.094	.000	2.666	.000	.465	.000	4.	115.	0.	120.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
75	250	KY 136	23.1	1213	.331	.000	2.662	.000	.396	.000	1.	155.	0.	156.
75	508	KY 85	1.0	1610	.245	.000	2.532	.000	.361	.000	2.	131.	0.	134.
75	774	KY 56	1.0	1949	.236	.000	2.787	.000	.415	.000	3.	195.	0.	198.
80	004	KY 292	13.3	2279	.275	.000	2.243	.000	.400	.000	3.	206.	0.	209.
80	755	KY 908	5.8	1734	.086	.000	2.670	.000	.396	.000	3.	58.	0.	61.
80	760	KY 3	15.5	1604	.053	.000	2.616	.000	.505	.000	3.	41.	0.	44.
80	766	KY1884	2.3	588	.079	.000	2.696	.000	.373	.000	1.	17.	0.	18.
84	Z01	CR1310	.1	38	.167	.000	2.286	.000	.363	.000	0.	2.	0.	2.
84	Z02	CR1310	3.1	236	.099	.000	3.350	.000	.442	.000	0.	13.	0.	13.
89	005	KY 81	1.5	2640	.106	.000	3.999	.000	.414	.000	4.	170.	0.	174.
92	037	KY 261	1.4	743	.164	.000	3.595	.000	.364	.000	1.	59.	0.	60.
92	252	US 62	14.7	2617	.106	.000	2.373	.000	.395	.000	4.	95.	0.	99.
94	285	KY 607	7.5	127	.088	.000	2.494	.000	.382	.000	0.	4.	0.	4.
94	286	KY 368	5.0	327	.034	.000	2.979	.000	.418	.000	1.	5.	0.	5.
94	759	KY 36	.1	296	.069	.000	2.503	.000	.389	.000	0.	7.	0.	8.
94	778	KY 227	16.0	694	.068	.000	2.531	.000	.362	.000	1.	16.	0.	17.
94	796	KY 227	25.0	2351	.076	.000	3.306	.000	1.038	.000	4.	226.	0.	229.
96	001	KY 10	.1	953	.067	.000	2.723	.000	.486	.000	2.	30.	0.	32.
100	078	KY 39	10.0	1868	.075	.000	3.141	.000	.425	.000	3.	68.	0.	71.
100	110	KY 70	8.0	1081	.098	.000	3.440	.000	.395	.000	2.	53.	0.	54.
100	368	KY1247	2.7	6604	.112	.000	3.702	.000	.458	.000	10.	458.	0.	468.
100	794	KY 635	5.8	1775	.072	.000	3.125	.000	.552	.000	3.	81.	0.	84.
113	C07	KY 360	8.0	1757	.140	.000	3.450	.000	.426	.000	3.	133.	0.	136.
117	004	US 41	9.9	6157	.221	.000	4.519	.000	.362	.000	9.	813.	0.	822.
117	005	KY 56	13.3	3598	.202	.000	4.154	.000	.357	.000	5.	393.	0.	398.
117	291	KY 132	24.4	1333	.179	.000	2.967	.000	.410	.000	2.	106.	0.	108.
117	A14	KY 120	9.0	609	.161	.000	2.939	.000	.769	.000	1.	81.	0.	82.
118	006	US 25W	14.3	7955	.065	.000	3.014	.000	.553	.000	13.	316.	0.	329.
118	293	KY 92	24.7	1390	.099	.000	3.532	.000	.672	.000	2.	119.	0.	121.
1	010	KY 80	14.4	4520	.065	.000	2.958	.000	.515	.000	8.	163.	0.	171.
1	021	KY 206	4.1	2271	.063	.000	2.680	.000	.479	.000	4.	67.	0.	71.
1	252	KY 55	5.1	1635	.109	.000	2.832	.000	.516	.000	3.	95.	0.	97.
2	250	KY1421	4.4	1671	.090	.000	2.702	.000	.410	.000	3.	61.	0.	64.
2	558	KY 100	6.0	1782	.118	.000	2.942	.000	.395	.000	3.	89.	0.	92.
2	A55	KY 100	14.7	11519	.165	.001	3.475	4.333	.529	2.423	17.	1270.	11.	1298.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
3	024	KY 44	12.5	3238	.125	.000	2.469	.000	.433	.000	5.	158.	0.	163.
3	534	KY 248	4.5	638	.269	.000	2.607	.000	.375	.000	1.	61.	0.	62.
4	036	KY 358	16.0	233	.045	.000	2.403	.000	.378	.000	0.	4.	0.	4.
5	762	KY 70	1.0	808	.087	.000	2.798	.000	.428	.000	1.	31.	0.	32.
5	807	US 68	5.5	3406	.066	.000	3.193	.000	.464	.000	6.	121.	0.	127.
6	010	KY 111	5.9	708	.147	.000	2.541	.000	.501	.000	1.	48.	0.	49.
6	253	US 60	9.0	1973	.063	.000	2.685	.000	.433	.000	3.	52.	0.	55.
6	256	KY 36	23.4	1488	.181	.000	2.438	.000	.477	.000	2.	114.	0.	116.
6	500	KY 36	11.0	2045	.082	.000	2.516	.000	.414	.000	3.	64.	0.	68.
6	A09	US 60	7.2	4986	.060	.000	2.847	.000	.545	.000	8.	170.	0.	178.
7	020	KY 221	2.3	3084	.242	.000	2.935	.000	.391	.000	4.	312.	0.	316.
7	268	KY 988	.6	401	.025	.000	2.516	.000	.388	.000	1.	4.	0.	4.
7	A74	KY 186	1.6	702	.179	.000	2.942	.000	.411	.000	1.	55.	0.	56.
8	284	KY 338	3.7	1838	.059	.000	3.688	.000	.503	.000	3.	72.	0.	75.
9	536	KY 57	.9	1464	.069	.000	2.814	.000	.504	.000	2.	52.	0.	55.
10	813	KY 5	.4	2455	.063	.000	2.998	.000	.573	.000	4.	97.	0.	101.
10	816	US 60	.8	10499	.120	.000	3.720	.000	.460	.000	17.	787.	0.	803.
10	C37	KY 538	.9	2909	.149	.000	2.266	.000	.415	.000	4.	148.	0.	153.
12	021	KY 19	13.4	1073	.038	.000	3.484	.000	.372	.000	2.	20.	0.	22.
12	500	KY 165	2.5	383	.043	.000	3.696	.000	.929	.000	1.	21.	0.	22.
12	502	KY 10	9.0	871	.042	.000	3.001	.000	.704	.000	1.	28.	0.	29.
12	755	KY1159	3.2	2129	.032	.000	3.421	.000	.556	.000	4.	48.	0.	52.
13	799	KY 30	14.6	6382	.110	.000	3.203	.000	.381	.000	10.	314.	0.	324.
13	A22	KY1812	3.1	2475	.055	.000	2.831	.000	.375	.000	4.	53.	0.	57.
14	007	KY 86	19.0	1531	.113	.000	3.060	.000	.407	.000	2.	79.	0.	82.
14	505	KY 261	7.8	2010	.143	.000	2.947	.000	.422	.000	3.	130.	0.	133.
14	797	KY 992	3.2	145	.112	.000	2.935	.000	.749	.000	0.	13.	0.	13.
15	C30	KY2674	2.8	1597	.053	.000	2.624	.000	.469	.000	3.	38.	0.	41.
16	042	KY 185	5.3	2051	.087	.000	3.133	.000	.454	.000	3.	93.	0.	96.
16	758	KY 70	16.1	2172	.138	.000	3.094	.000	.429	.000	3.	145.	0.	149.
16	A40	US 231	9.6	6578	.112	.010	3.675	4.286	.419	3.396	11.	411.	37.	459.
16	A41	KY 70	13.9	5349	.096	.000	3.156	.000	.450	.000	9.	267.	0.	276.
17	041	KY 70	14.0	789	.112	.000	2.991	.000	.455	.000	1.	44.	0.	45.
17	288	KY 91	5.0	1284	.102	.000	3.145	.000	.496	.000	2.	75.	0.	77.
17	523	US 62	2.9	4096	.066	.000	2.737	.000	.438	.000	7.	118.	0.	125.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
18	034	KY 94	19.7	903	.073	.000	2.930	.000	.378	.000	2.	27.	0.	28.
18	315	KY 94	11.5	7936	.065	.000	2.999	.000	.463	.000	13.	260.	0.	273.
18	B21	KY 94	6.6	4462	.063	.000	3.083	.000	.462	.000	7.	146.	0.	153.
19	012	KY 8	23.2	786	.030	.000	3.038	.000	.369	.000	1.	9.	0.	11.
19	276	KY 154	1.2	1120	.033	.000	3.181	.000	.377	.000	2.	17.	0.	19.
19	287	KY 824	.7	982	.037	.000	3.042	.000	.394	.000	2.	16.	0.	18.
20	009	US 62	2.3	2064	.155	.000	3.472	.000	.375	.000	3.	152.	0.	156.
20	025	KY1371	6.0	176	.208	.000	2.273	.000	.376	.000	0.	12.	0.	12.
20	030	KY 408	.9	158	.037	.000	2.229	.000	.397	.000	0.	2.	0.	2.
20	506	KY 80	.7	330	.077	.000	3.210	.000	.385	.000	1.	12.	0.	12.
21	002	US 42	14.5	4764	.166	.000	4.495	.000	.441	.000	7.	571.	0.	578.
21	282	KY 389	3.4	734	.042	.000	3.810	.000	.656	.000	1.	28.	0.	29.
22	034	KY 1	13.9	1754	.065	.000	2.852	.000	.428	.000	3.	50.	0.	53.
22	504	US 60	5.0	3245	.091	.000	3.141	.000	.516	.000	5.	173.	0.	179.
22	750	KY 2	3.8	1336	.234	.000	2.790	.000	.509	.000	2.	162.	0.	163.
22	A03	US 60	23.8	5947	.129	.002	3.743	4.000	.460	3.144	9.	484.	9.	502.
23	005	KY 70	22.0	2457	.086	.000	2.836	.000	.573	.000	4.	125.	0.	129.
23	500	KY 80	2.7	1013	.107	.000	2.463	.000	.420	.000	2.	41.	0.	43.
23	A07	KY2314	.9	3323	.050	.000	2.690	.000	.527	.000	6.	86.	0.	92.
23	A62	KY 49	1.7	2001	.075	.000	2.614	.000	.466	.000	3.	67.	0.	70.
23	A64	KY 70	13.0	4059	.071	.000	2.696	.000	.453	.000	7.	128.	0.	135.
24	007	KY 107	29.6	1578	.028	.000	2.916	.000	.406	.000	3.	19.	0.	22.
25	276	KY 89	5.3	2281	.063	.000	2.732	.000	.426	.000	4.	61.	0.	65.
25	509	KY 418	5.0	703	.089	.000	2.677	.000	.503	.000	1.	31.	0.	33.
26	261	KY 149	.6	1804	.026	.000	2.738	.000	.407	.000	3.	20.	0.	23.
26	291	US 421	.2	1231	.087	.000	4.235	.000	.461	.000	2.	77.	0.	79.
26	501	KY 11	7.7	2214	.094	.000	3.293	.000	.418	.000	4.	104.	0.	108.
26	797	KY 638	10.0	1535	.038	.000	3.149	.000	.457	.000	3.	31.	0.	34.
27	A03	KY1590	.6	4745	.104	.000	3.353	.000	.521	.000	8.	314.	0.	322.
28	557	KY 70	2.0	503	.288	.000	2.108	.000	.356	.000	1.	40.	0.	40.
28	A63	KY 120	2.0	1224	.163	.000	2.592	.000	.454	.000	2.	86.	0.	88.
30	008	KY 144	5.8	2145	.132	.000	3.676	.000	.391	.000	3.	150.	0.	153.
30	253	US 231	9.7	8476	.090	.000	2.675	.000	.439	.000	14.	328.	0.	342.
31	278	US 31W	.9	1920	.060	.000	2.885	.000	.451	.000	3.	55.	0.	58.
32	507	KY 173	.4	600	.079	.000	2.609	.000	.391	.000	1.	18.	0.	19.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
33	014	KY 213	2.7	328	.067	.000	2.466	.000	.383	.000	1.	8.	0.	8.
33	810	KY 89	13.8	6972	.088	.000	2.666	.000	.429	.000	11.	255.	0.	267.
34	G19	KY1968	2.9	1130	.048	.000	2.911	.000	.378	.000	2.	21.	0.	23.
35	251	KY 111	6.0	1445	.085	.000	2.541	.000	.425	.000	2.	49.	0.	51.
35	774	KY 165	1.0	2536	.096	.000	2.610	.000	.418	.000	4.	97.	0.	101.
35	A41	KY 11X	1.1	6466	.098	.000	2.784	.000	.511	.000	11.	328.	0.	339.
35	A46	KY 32	10.3	5855	.066	.000	2.910	.000	.559	.000	10.	229.	0.	239.
36	009	KY 302	2.0	5297	.105	.000	3.989	.000	.406	.000	9.	327.	0.	336.
36	250	KY 680	13.2	7250	.050	.000	2.998	.000	.556	.000	12.	221.	0.	233.
36	263	KY 680	5.1	4426	.135	.000	3.137	.000	.427	.000	7.	291.	0.	298.
36	265	KY 979	12.6	3050	.081	.000	2.501	.000	.453	.000	5.	102.	0.	107.
36	284	KY 122	19.2	3232	.057	.000	2.707	.000	.430	.000	6.	78.	0.	83.
36	321	KY 979	14.9	3050	.053	.000	2.795	.000	.488	.000	5.	81.	0.	86.
36	508	KY1091	.8	452	.049	.000	3.125	.000	.373	.000	1.	9.	0.	10.
36	A55	KY1428	13.6	7259	.046	.000	2.756	.000	.533	.000	13.	180.	0.	193.
38	308	KY 166	12.6	6298	.145	.000	3.187	.000	.425	.000	10.	451.	0.	461.
38	A88	KY 307	1.0	1595	.092	.000	3.162	.000	.411	.000	3.	70.	0.	72.
39	004	US 127	2.8	1818	.105	.000	4.229	.000	.668	.000	3.	197.	0.	200.
39	250	US 127	2.2	3643	.038	.000	3.744	.000	.713	.000	6.	135.	0.	141.
39	752	KY 35	7.2	1595	.052	.000	3.732	.000	.490	.000	3.	55.	0.	57.
39	768	KY1039	6.4	5813	.107	.000	4.482	.000	.560	.000	9.	568.	0.	577.
41	287	US 25	4.4	1076	.046	.000	3.076	.000	.492	.000	2.	28.	0.	29.
42	017	KY1276	2.0	1015	.097	.000	3.403	.000	.416	.000	2.	51.	0.	52.
42	050	KY 348	4.6	1104	.115	.000	2.355	.000	.461	.000	2.	50.	0.	51.
42	104	KY 131	11.0	2161	.080	.000	3.306	.000	.419	.000	4.	86.	0.	90.
42	274	KY 339	8.6	1436	.080	.000	3.043	.000	.405	.000	2.	51.	0.	54.
42	391	KY 97	14.0	1981	.095	.000	3.786	.000	.440	.000	3.	114.	0.	117.
42	402	KY 303	8.3	1335	.093	.000	3.314	.000	.413	.000	2.	62.	0.	64.
42	750	KY 80	5.4	2321	.086	.000	3.127	.000	.427	.000	4.	97.	0.	100.
43	009	US 62	28.8	2524	.054	.000	2.742	.000	.465	.000	4.	63.	0.	68.
43	251	KY 224	1.5	2786	.122	.000	3.035	.000	.392	.000	4.	148.	0.	152.
43	257	KY 187	9.7	3434	.087	.000	3.141	.000	.435	.000	6.	150.	0.	155.
43	540	KY 79	4.4	2877	.115	.000	3.173	.000	.416	.000	5.	160.	0.	165.
43	554	KY 79	6.8	1518	.142	.000	2.667	.000	.415	.000	2.	87.	0.	89.
43	Z01	KY 88	2.5	2612	.108	.000	2.452	.000	.394	.000	4.	99.	0.	103.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
44	016	KY 88	8.7	2030	.076	.000	3.086	.000	.363	.000	3.	63.	0.	66.
44	511	US 68	2.1	831	.137	.000	3.163	.000	.374	.000	1.	50.	0.	51.
45	008	KY 1	13.4	1919	.073	.000	2.944	.000	.406	.000	3.	61.	0.	65.
45	768	KY 7	12.3	881	.073	.000	2.636	.000	.388	.000	1.	24.	0.	25.
45	E50	KY 207	15.0	4559	.034	.000	2.880	.000	.579	.000	8.	94.	0.	102.
47	254	US 31W	11.7	3243	.127	.000	3.244	.000	.420	.000	5.	205.	0.	210.
47	360	US 31W	1.7	3235	.112	.000	3.261	.000	.505	.000	5.	219.	0.	224.
47	518	KY 84	11.8	1111	.077	.000	3.219	.000	.483	.000	2.	49.	0.	51.
48	016	KY 221	23.4	296	.274	.000	4.645	.000	.376	.000	0.	52.	0.	53.
48	250	KY 38	24.6	851	.081	.000	2.603	.000	.447	.000	1.	30.	0.	31.
48	A06	KY 38	.4	6464	.109	.002	3.125	4.000	.437	3.144	10.	352.	9.	371.
49	276	KY 32	13.1	1935	.025	.000	2.779	.000	.468	.000	3.	23.	0.	26.
50	260	US 31E	.7	4041	.118	.000	3.376	.000	.541	.000	6.	317.	0.	324.
50	296	KY 357	2.7	1104	.082	.000	2.566	.000	.379	.000	2.	32.	0.	34.
50	506	KY 728	3.7	1246	.156	.000	2.738	.000	.419	.000	2.	82.	0.	84.
50	513	KY 88	8.4	1577	.164	.000	2.854	.000	.394	.000	2.	106.	0.	109.
50	A20	KY 218	8.1	5345	.068	.000	3.152	.000	.525	.000	9.	220.	0.	229.
50	C40	US 31W	12.5	3887	.081	.000	3.678	.000	.394	.000	6.	168.	0.	174.
52	292	KY 22	10.9	1485	.130	.000	2.788	.000	.567	.000	2.	111.	0.	114.
52	538	KY 146	1.2	2848	.069	.000	2.636	.000	.413	.000	5.	77.	0.	82.
52	778	KY 153	6.7	6332	.101	.000	3.120	.000	.467	.000	10.	339.	0.	349.
53	252	KY 307	8.8	655	.091	.000	3.479	.000	.491	.000	1.	37.	0.	38.
53	772	KY 58	7.1	681	.047	.000	2.961	.000	.378	.000	1.	13.	0.	14.
54	516	KY 109	13.0	1063	.147	.000	2.800	.000	.398	.000	2.	63.	0.	65.
54	586	KY 70	16.0	2663	.142	.000	2.366	.000	.424	.000	4.	139.	0.	143.
54	D18	KY 109	1.3	1832	.077	.000	3.064	.000	.357	.000	3.	56.	0.	59.
55	013	KY 89	19.4	497	.034	.000	3.128	.000	.378	.000	1.	7.	0.	8.
57	008	KY1980	3.9	2671	.063	.000	2.758	.000	.457	.000	5.	77.	0.	81.
58	256	KY 825	13.6	762	.032	.000	2.211	.000	.375	.000	1.	8.	0.	9.
58	259	KY 40	8.9	14607	.072	.000	3.447	.000	.526	.000	25.	698.	0.	723.
58	271	KY 40	13.9	2926	.034	.000	2.600	.000	.508	.000	5.	48.	0.	53.
58	279	KY1750	3.8	1252	.056	.000	2.712	.000	.525	.000	2.	36.	0.	38.
58	776	KY 201	3.7	2012	.056	.000	2.810	.000	.513	.000	3.	59.	0.	62.
58	A68	KY1428	3.1	7628	.076	.000	3.035	.000	.581	.000	13.	376.	0.	388.
60	006	KY 550	26.3	1705	.117	.000	4.778	.000	.351	.000	3.	121.	0.	124.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
60	252	KY 582	4.2	2121	.091	.000	4.253	.000	.376	.000	3.	112.	0.	115.
60	259	KY 899	3.6	2039	.043	.000	2.802	.000	.410	.000	4.	36.	0.	40.
60	519	KY1088	8.2	766	.132	.000	2.532	.000	.439	.000	1.	41.	0.	42.
61	D71	KY 225	14.7	3721	.047	.000	2.533	.000	.478	.000	6.	76.	0.	83.
62	506	KY 357	6.3	830	.140	.000	2.567	.000	.362	.000	1.	40.	0.	41.
62	518	US 31E	.2	1893	.086	.000	2.938	.000	.390	.000	3.	68.	0.	71.
62	531	KY 84	1.0	1955	.123	.000	2.964	.000	.426	.000	3.	111.	0.	114.
62	A04	KY 210	5.0	9510	.065	.000	2.849	.000	.444	.000	16.	285.	0.	302.
63	053	US 25	8.5	11834	.109	.000	3.705	.000	.566	.000	19.	989.	0.	1008.
63	803	KY 490	2.0	5223	.065	.000	3.021	.000	.456	.000	9.	171.	0.	180.
64	012	KY 3	11.3	933	.088	.000	3.316	.000	.391	.000	2.	39.	0.	41.
64	015	KY2565	1.7	3818	.080	.000	3.528	.000	.397	.000	6.	156.	0.	162.
64	031	KY 3	23.2	1443	.066	.000	2.468	.000	.381	.000	2.	33.	0.	35.
64	764	KY 201	12.0	403	.118	.000	2.482	.000	.600	.000	1.	26.	0.	26.
65	515	KY 587	6.7	259	.059	.000	2.341	.000	.359	.000	0.	5.	0.	5.
65	A26	KY 52	12.3	2213	.023	.000	2.590	.000	.452	.000	4.	22.	0.	26.
66	505	KY 66	.2	575	.081	.000	3.291	.000	.539	.000	1.	30.	0.	31.
66	757	US 421	35.0	1211	.082	.000	3.683	.000	.348	.000	2.	46.	0.	48.
67	010	KY 7	27.1	845	.178	.000	3.099	.000	.413	.000	1.	70.	0.	72.
67	015	KY 803	.1	507	.167	.000	2.375	.000	.368	.000	1.	27.	0.	28.
67	291	KY1862	1.3	1286	.047	.000	2.517	.000	.450	.000	2.	25.	0.	27.
67	505	KY 160	4.3	551	.126	.000	2.994	.000	.624	.000	1.	48.	0.	49.
67	506	KY 160	9.6	520	.118	.000	2.820	.000	.684	.000	1.	44.	0.	44.
67	759	KY 7	12.9	4174	.050	.000	3.038	.000	.508	.000	7.	117.	0.	124.
67	A08	KY 805	8.8	3002	.101	.000	3.558	.000	.576	.000	5.	226.	0.	231.
67	A26	KY 805	13.3	1541	.243	.000	3.092	.000	.914	.000	2.	386.	0.	388.
67	C14	KY 15X	1.3	2555	.047	.000	2.658	.000	.518	.000	4.	60.	0.	65.
68	504	KY 377	7.0	486	.182	.000	3.649	.000	.714	.000	1.	86.	0.	86.
68	764	KY 57	4.1	2149	.127	.000	3.127	.000	.508	.000	3.	158.	0.	162.
68	A41	KY 8	12.2	1666	.144	.000	3.623	.000	.735	.000	3.	233.	0.	236.
69	015	KY 39	13.4	1273	.070	.000	2.743	.000	.451	.000	2.	39.	0.	41.
69	A60	KY 78	12.9	3164	.075	.000	3.023	.000	.469	.000	5.	122.	0.	128.
72	067	KY 293	.2	2575	.112	.000	3.061	.000	.383	.000	4.	123.	0.	127.
73	276	KY 348	5.0	1217	.046	.000	2.510	.000	.393	.000	2.	20.	0.	22.
73	293	KY 131	.3	2203	.193	.000	2.737	.000	.343	.000	3.	146.	0.	149.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
73	798	US 62	3.1	3037	.068	.000	3.313	.000	.410	.000	5.	103.	0.	108.
73	803	KY 305	1.0	432	.082	.000	2.537	.000	.389	.000	1.	13.	0.	14.
73	804	KY 726	2.0	1202	.056	.000	2.642	.000	.431	.000	2.	28.	0.	30.
73	840	KY 358	7.2	972	.067	.000	3.013	.000	.477	.000	2.	35.	0.	36.
74	313	KY 92	16.4	6024	.068	.000	2.954	.000	.555	.000	10.	246.	0.	256.
75	018	KY 136	19.3	1314	.053	.000	3.080	.000	.420	.000	2.	33.	0.	35.
75	259	KY 81	6.0	1508	.161	.000	2.890	.000	.373	.000	2.	95.	0.	98.
75	A21	KY 81	11.2	3700	.199	.000	2.784	.000	.401	.000	5.	300.	0.	305.
76	567	KY1983	2.4	318	.020	.000	2.559	.000	.394	.000	1.	3.	0.	3.
76	636	KY2872	2.5	6396	.082	.000	2.537	.000	.392	.000	11.	191.	0.	202.
77	002	KY1081	18.1	973	.057	.000	2.589	.000	.683	.000	2.	35.	0.	37.
77	015	KY 40	1.2	2737	.053	.000	2.540	.000	.372	.000	5.	50.	0.	55.
77	253	KY 7	8.1	847	.365	.000	4.670	.000	.342	.000	1.	181.	0.	182.
77	500	KY 30	2.7	400	.094	.000	3.197	.000	.438	.000	1.	20.	0.	21.
78	507	KY 49	21.3	2473	.081	.000	3.057	.000	.525	.000	4.	117.	0.	121.
78	518	KY 208	4.9	865	.150	.000	2.312	.000	.364	.000	1.	40.	0.	41.
78	755	KY 52	2.2	1181	.183	.000	2.452	.000	.451	.000	2.	87.	0.	88.
78	775	KY 429	5.0	605	.135	.000	3.179	.000	.650	.000	1.	62.	0.	63.
79	506	KY 58	3.6	1195	.133	.000	2.876	.000	.479	.000	2.	80.	0.	82.
79	569	KY 408	9.9	2949	.049	.000	2.886	.000	.485	.000	5.	74.	0.	79.
79	821	KY 95	.6	1533	.087	.000	2.286	.000	.370	.000	3.	41.	0.	44.
79	C05	KY1523	8.3	4742	.093	.000	3.452	.000	.420	.000	8.	235.	0.	242.
80	252	KY 292	11.8	1416	.070	.000	3.369	.000	.392	.000	2.	48.	0.	50.
80	261	KY1439	11.0	1262	.064	.000	3.154	.000	.674	.000	2.	64.	0.	66.
81	010	KY 10	9.6	1271	.119	.000	3.253	.000	.494	.000	2.	89.	0.	91.
81	266	KY1237	2.2	428	.066	.000	2.742	.000	.496	.000	1.	14.	0.	15.
81	522	US 62	7.0	615	.071	.000	2.754	.000	.456	.000	1.	20.	0.	21.
81	X02	KY 10	5.3	1831	.088	.000	3.221	.000	.499	.000	3.	94.	0.	97.
82	019	KY 228	18.5	1094	.105	.000	2.569	.000	.398	.000	2.	43.	0.	44.
82	255	KY 144	23.7	1598	.112	.000	2.623	.000	.399	.000	3.	68.	0.	71.
82	273	KY1638	2.6	7230	.118	.011	3.381	4.300	.361	3.102	12.	376.	49.	437.
82	318	KY1238	10.1	1620	.106	.000	2.822	.000	.425	.000	3.	74.	0.	77.
82	509	KY 144	11.9	1873	.092	.000	3.890	.000	.400	.000	3.	98.	0.	101.
82	554	KY 79	6.5	4874	.205	.000	2.629	.000	.479	.000	7.	460.	0.	467.
83	003	KY2023	.1	132	.129	.000	3.197	.000	.567	.000	0.	11.	0.	11.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
83	256	KY 946	.1	599	.093	.000	2.740	.000	.377	.000	1.	21.	0.	22.
83	750	KY 77	9.7	943	.075	.000	2.847	.000	.510	.000	2.	38.	0.	40.
85	043	KY 80	1.7	1322	.070	.000	3.132	.000	.431	.000	2.	45.	0.	48.
85	510	US 68	4.1	2336	.086	.000	2.968	.000	.417	.000	4.	90.	0.	94.
85	Y01	US 68	9.0	6290	.137	.000	3.669	.000	.506	.000	10.	584.	0.	594.
85	Y02	KY 163	11.0	2500	.097	.000	3.556	.000	.444	.000	4.	141.	0.	145.
86	262	KY 63	11.5	1533	.131	.000	2.676	.000	.369	.000	2.	73.	0.	75.
86	540	KY1366	3.4	1276	.062	.000	3.626	.000	.462	.000	2.	48.	0.	50.
86	A01	KY 163	10.3	5780	.169	.000	2.808	.000	.408	.000	9.	410.	0.	418.
86	A18	KY 100	14.3	5313	.262	.011	4.032	4.313	.350	2.846	7.	707.	72.	786.
87	017	KY 713	10.6	1344	.037	.000	2.507	.000	.383	.000	2.	17.	0.	20.
89	251	KY 70	16.5	1668	.062	.000	2.757	.000	.437	.000	3.	45.	0.	48.
89	504	KY 189	7.3	1984	.124	.000	2.805	.000	.430	.000	3.	109.	0.	112.
89	A14	US 62	12.0	9052	.055	.000	4.428	.000	.754	.000	15.	603.	0.	618.
89	B07	KY 189C	.1	2858	.162	.000	3.320	.000	.438	.000	4.	245.	0.	250.
90	002	US 62	33.8	1007	.199	.000	2.299	.000	.406	.000	1.	68.	0.	70.
90	006	KY 48	3.9	1835	.091	.000	2.906	.000	.398	.000	3.	70.	0.	73.
90	075	KY 555	.6	706	.107	.000	3.196	.000	.417	.000	1.	37.	0.	38.
90	A92	KY1430	2.1	3930	.109	.000	2.292	.000	.392	.000	6.	141.	0.	147.
91	A06	KY 32	8.5	7451	.075	.000	2.776	.000	.467	.000	13.	264.	0.	277.
91	A19	KY 36	3.4	5516	.087	.000	3.092	.000	.564	.000	9.	305.	0.	314.
92	002	KY 69	28.2	1525	.140	.000	3.532	.000	.395	.000	2.	107.	0.	110.
92	250	US 231	4.8	4305	.070	.000	2.609	.000	.420	.000	7.	121.	0.	128.
92	502	US 231	14.1	6343	.071	.000	3.662	.000	.482	.000	11.	287.	0.	298.
92	756	KY 54	3.0	2416	.072	.000	3.383	.000	.409	.000	4.	88.	0.	92.
92	A05	US 62	9.1	4401	.107	.000	2.531	.000	.439	.000	7.	192.	0.	199.
94	283	KY 227	10.8	680	.017	.000	3.047	.000	.367	.000	1.	4.	0.	6.
94	510	KY 355	2.2	710	.094	.000	3.565	.000	.825	.000	1.	70.	0.	71.
95	001	KY 28	2.4	1330	.072	.000	2.806	.000	.375	.000	2.	37.	0.	39.
95	752	KY 30	10.8	1504	.076	.000	3.304	.000	.433	.000	3.	61.	0.	63.
95	A04	KY 30	11.2	5130	.072	.000	2.967	.000	.361	.000	9.	145.	0.	153.
95	Y01	KY 30	10.8	1417	.127	.000	2.876	.000	.365	.000	2.	69.	0.	71.
96	016	KY 159	9.8	639	.060	.000	3.591	.000	.697	.000	1.	35.	0.	36.
97	002	KY 476	18.2	569	.115	.000	3.353	.000	.449	.000	1.	36.	0.	37.
97	255	KY 463	5.8	1271	.202	.000	4.127	.000	.428	.000	2.	164.	0.	166.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
97	506	KY 80	1.6	2746	.093	.000	4.273	.000	.367	.000	5.	146.	0.	151.
97	761	KY 28	17.9	4598	.096	.000	3.693	.000	.399	.000	8.	238.	0.	246.
97	A09	KY 15X	1.8	9443	.052	.000	3.274	.000	.587	.000	16.	345.	0.	362.
97	A45	KY1762	.1	2089	.023	.000	2.197	.000	.368	.000	4.	14.	0.	18.
98	003	KY 319	6.7	4706	.091	.000	3.593	.000	.540	.000	8.	302.	0.	310.
98	038	KY1056	7.3	1523	.100	.000	4.107	.000	.361	.000	2.	83.	0.	86.
98	112	KY 468	13.3	1029	.293	.000	3.261	.000	.642	.000	1.	231.	0.	232.
98	123	KY3220	.9	2934	.109	.000	4.268	.000	.570	.000	5.	284.	0.	289.
98	250	KY 194	69.7	865	.064	.000	2.870	.000	.473	.000	1.	28.	0.	30.
98	271	KY 197	13.4	2638	.117	.000	3.780	.000	.408	.000	4.	174.	0.	178.
98	511	KY 122	5.1	2263	.083	.000	4.089	.000	.441	.000	4.	124.	0.	128.
98	512	KY1469	11.1	3126	.063	.000	3.101	.000	.446	.000	5.	100.	0.	106.
98	513	KY 197	.8	1836	.146	.000	3.935	.000	.395	.000	3.	152.	0.	155.
98	542	KY3226	1.4	1467	.041	.000	2.429	.000	.506	.000	3.	27.	0.	29.
98	597	KY 122	10.6	2557	.067	.000	3.090	.000	.470	.000	4.	91.	0.	95.
98	616	KY 610	.1	1203	.165	.000	3.387	.000	.401	.000	2.	99.	0.	101.
98	628	KY3527	.2	613	.551	.000	4.384	.000	.438	.000	0.	238.	0.	238.
98	753	KY 194	8.9	1116	.099	.000	3.371	.000	.372	.000	2.	50.	0.	52.
98	809	KY3218	.2	1205	.484	.000	5.042	.000	.351	.000	1.	378.	0.	379.
98	C01	KY 80	2.5	5573	.159	.000	3.948	.000	.671	.000	9.	856.	0.	865.
98	E10	KY 292	6.2	4518	.044	.000	2.627	.000	.565	.000	8.	108.	0.	116.
99	A30	KY 213	7.6	2227	.064	.000	2.912	.000	.386	.000	4.	59.	0.	63.
99	C01	KY 11	21.0	3546	.062	.000	3.262	.000	.434	.000	6.	113.	0.	119.
99	C06	KY1057	.2	1965	.056	.000	2.551	.000	.384	.000	3.	40.	0.	43.
99	C18	KY 11	19.8	7412	.045	.000	2.783	.000	.398	.000	13.	134.	0.	147.
100	094	KY 635	8.0	3703	.064	.000	3.108	.000	.500	.000	6.	134.	0.	141.
100	124	KY 70	3.3	2165	.103	.000	3.031	.000	.432	.000	3.	107.	0.	111.
100	251	KY1247	2.5	470	.058	.000	2.930	.000	.795	.000	1.	24.	0.	25.
100	296	KY 192	2.9	2615	.072	.000	2.643	.000	.478	.000	4.	87.	0.	91.
100	758	KY 80	4.5	2758	.090	.000	2.652	.000	.411	.000	5.	99.	0.	103.
102	009	US 25	24.3	3853	.037	.000	2.783	.000	.430	.000	7.	63.	0.	70.
102	019	US 25	16.1	3872	.064	.000	3.044	.000	.471	.000	7.	130.	0.	137.
102	753	KY 70	5.0	2176	.066	.000	2.892	.000	.438	.000	4.	66.	0.	70.
103	026	KY 32	12.0	3562	.064	.000	2.631	.000	.462	.000	6.	102.	0.	108.
103	262	KY 173	2.1	1107	.140	.000	2.371	.000	.369	.000	2.	50.	0.	51.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
103	525	KY 801	4.2	1010	.091	.000	3.303	.000	.492	.000	2.	55.	0.	57.
103	527	KY1274	2.6	843	.051	.000	3.121	.000	.419	.000	1.	20.	0.	22.
104	022	KY 80	5.9	6742	.064	.000	2.982	.000	.517	.000	11.	244.	0.	255.
104	505	KY 55	5.6	825	.118	.000	3.055	.000	.447	.000	1.	48.	0.	50.
104	761	KY 379	14.6	1530	.081	.000	2.704	.000	.423	.000	3.	52.	0.	55.
104	C18	US 127X	1.8	2156	.067	.000	2.861	.000	.441	.000	4.	67.	0.	70.
105	017	US 25	21.0	360	.113	.000	3.408	.000	.361	.000	1.	18.	0.	19.
105	289	KY 922	.2	2407	.081	.000	2.721	.000	.425	.000	4.	83.	0.	87.
106	005	KY 12	1.5	1386	.074	.000	2.618	.000	.376	.000	2.	37.	0.	39.
106	036	KY 43	5.7	2049	.070	.000	2.539	.000	.375	.000	3.	50.	0.	53.
106	252	KY 395	8.9	2223	.117	.000	2.842	.000	.431	.000	4.	116.	0.	120.
107	555	KY 383	7.9	3241	.073	.000	3.331	.000	.533	.000	5.	154.	0.	159.
107	779	KY 73	11.3	2004	.059	.000	2.866	.000	.436	.000	3.	53.	0.	57.
109	023	KY 337	5.2	244	.077	.000	2.829	.000	.390	.000	0.	8.	0.	8.
109	A95	KY 70	3.0	4369	.088	.000	3.171	.000	.434	.000	7.	194.	0.	201.
110	503	KY 181	7.9	3110	.146	.000	3.131	.000	.416	.000	5.	216.	0.	221.
110	509	KY 346	.6	709	.199	.000	3.755	.000	.341	.000	1.	66.	0.	67.
110	534	KY 104	14.0	598	.099	.000	3.227	.000	.366	.000	1.	25.	0.	26.
110	A12	US 68X	3.3	1860	.092	.000	3.050	.000	.425	.000	3.	81.	0.	84.
111	507	KY 164	.5	972	.121	.000	2.943	.000	.439	.000	2.	56.	0.	58.
112	003	KY 754	4.1	1765	.157	.000	3.055	.000	.435	.000	3.	134.	0.	136.
112	253	US 42	1.0	4562	.086	.000	2.946	.000	.448	.000	8.	189.	0.	196.
113	005	KY 130	11.0	2301	.056	.000	3.843	.000	.427	.000	4.	78.	0.	82.
113	778	KY 360	3.0	632	.204	.000	3.541	.000	.430	.000	1.	72.	0.	73.
114	004	US 68	22.6	3306	.118	.000	3.238	.000	.420	.000	5.	195.	0.	200.
115	A04	KY 152	8.2	2004	.058	.000	2.968	.000	.533	.000	3.	67.	0.	70.
115	A14	KY 528	.5	1448	.065	.000	2.774	.000	.557	.000	2.	53.	0.	55.
116	261	KY 92	11.6	2782	.074	.000	2.657	.000	.431	.000	5.	86.	0.	91.
117	003	KY 56	11.5	1699	.147	.000	2.666	.000	.407	.000	3.	99.	0.	102.
117	558	KY 293	1.2	682	.088	.000	2.817	.000	.431	.000	1.	27.	0.	28.
117	579	KY 132	9.0	1504	.080	.000	2.726	.000	.426	.000	3.	51.	0.	54.
117	604	KY 109	7.5	3127	.097	.000	3.198	.000	.430	.000	5.	153.	0.	158.
118	045	KY 6	1.6	1301	.071	.000	3.270	.000	.364	.000	2.	40.	0.	42.
118	046	KY 26	9.8	2541	.064	.000	2.429	.000	.422	.000	4.	61.	0.	65.
118	274	KY1804	1.7	1301	.034	.000	3.104	.000	.431	.000	2.	22.	0.	25.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
						NORMAL	HEAVY	NORMAL	HEAVY	4-TIRE	NON-COAL	COAL	
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL		COAL	COAL	VEHICLES	TRUCKS	TRUCKS	TOTAL
119	001	KY 191	3.2	1945	.068	.000	2.684	.000	.431	.000	3.	57.	0.	60.
119	013	KY 746	4.8	287	.112	.000	2.318	.000	.382	.000	0.	10.	0.	11.
119	254	KY 205	4.1	1398	.178	.000	2.812	.000	.369	.000	2.	95.	0.	97.
119	A22	KY 15	9.8	2840	.064	.000	2.225	.000	.381	.000	5.	57.	0.	61.
120	052	KY1681	6.0	1018	.039	.000	2.365	.000	.373	.000	2.	13.	0.	15.
120	058	KY 341	.7	1643	.048	.000	2.483	.000	.412	.000	3.	29.	0.	32.

SUMMARY OF AVERAGE VALUES FOR
AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL

YEAR	3 YR AVG	13	11	07	05	04	01	00	99	98	97	96
UNCLASSIFIED ROADS (ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)												
NO OF CLASSIFICATION STA.	1931	636	944	351	365	430	189	37	69	98	54	64
AADT	2489	2418	2510	2557	2672	2506	2625	2040	3337	4285	4959	3078
PERCENT TRUCKS	10.047	9.634	10.306	10.097	10.213	10.209	11.364	9.750	8.596	9.685	8.848	7.483
AXLES PER TRUCK	3.077	3.027	3.079	3.165	3.129	3.095	3.011	2.953	3.023	3.038	3.061	2.877
EAL'S PER TRUCK AXLE	.423	.452	.348	.571	.550	.505	.234	.233	.194	.222	.226	.214
CLASSIFIED ROADS (MANUAL LOCATION WITH 3% OR MORE OF TRUCKS CLASSIFIED AS HEAVY/COAL)												
NO OF CLASSIFICATION STA.	0	0	0	0	0	0	5	5	11	14	17	23
AADT	0	0	0	0	0	0	3466	2785	5037	3512	6475	4453
PERCENT TRUCKS	.000	.000	.000	.000	.000	.000	7.382	6.125	11.662	17.000	7.541	9.820
PERCENT OF TRUCKS CLASSIFIED AS HEAVY/COAL	.000	.000	.000	.000	.000	.000	29.182	38.523	33.964	43.723	31.279	32.668
AXLES PER TRUCK NORMAL	.000	.000	.000	.000	.000	.000	3.427	3.352	2.934	3.548	2.984	2.888
AXLES PER TRUCK HEAVY/COAL	.000	.000	.000	.000	.000	.000	4.286	4.181	4.729	4.369	4.789	4.545
EAL'S PER TRUCK AXLE NORMAL	.000	.000	.000	.000	.000	.000	.148	.208	.179	.293	.210	.194
EAL'S PER TRUCK AXLE HEAVY/COAL	.000	.000	.000	.000	.000	.000	2.671	3.155	2.678	3.143	1.194	1.153

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS IV -- URBAN INTERSTATE

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
8	033	I 75	181.0	164732	.155	.000	3.997	.000	.348	.000	250.	12957.	0.	13207.
8	087	I 275	2.0	75525	.103	.000	3.962	.000	.322	.000	123.	3626.	0.	3749.
8	253	I 75	178.0	117141	.196	.000	4.472	.000	.360	.000	171.	13508.	0.	13679.
8	266	I 75	173.0	90793	.240	.000	4.467	.000	.375	.000	125.	13324.	0.	13449.
8	272	I 75	180.0	133930	.174	.000	4.152	.000	.355	.000	201.	12557.	0.	12758.
34	049	I 75	113.0	82052	.199	.000	4.340	.000	.365	.000	119.	9446.	0.	9565.
56	157	I 264	17.0	142599	.066	.000	3.883	.000	.342	.000	243.	4542.	0.	4785.
56	441	I 264	16.0	149157	.063	.000	3.775	.000	.428	.000	252.	5566.	0.	5818.
56	590	I 65	123.0	97625	.200	.000	4.388	.000	.354	.000	142.	11066.	0.	11208.
56	713	I 65	127.0	147525	.118	.000	4.218	.000	.473	.000	230.	12700.	0.	12930.
56	753	I 64	.0	69390	.092	.000	4.157	.000	.445	.000	114.	4298.	0.	4412.
56	783	I 65	130.0	163670	.128	.000	3.998	.000	.342	.000	259.	10444.	0.	10703.
56	B14	I 265	27.0	61986	.110	.000	3.842	.000	.319	.000	100.	3045.	0.	3145.
56	B39	I 265	29.0	56258	.103	.000	3.869	.000	.322	.000	92.	2622.	0.	2714.
56	D01	I 265	23.0	59133	.119	.000	3.643	.000	.306	.000	95.	2853.	0.	2948.
56	D09	I 265	14.0	68186	.108	.000	3.829	.000	.303	.000	111.	3103.	0.	3214.
56	G23	I 65	128.0	152341	.128	.000	4.073	.000	.337	.000	242.	9763.	0.	10005.
56	G81	I 65	125.0	128723	.137	.000	4.231	.000	.351	.000	202.	9585.	0.	9787.
59	078	I 275	78.0	88150	.061	.000	3.553	.000	.304	.000	150.	2122.	0.	2273.
59	081	I 275	79.0	89592	.062	.000	3.551	.000	.305	.000	153.	2213.	0.	2365.
59	792	I 75	184.0	192353	.134	.000	4.313	.000	.349	.000	302.	14202.	0.	14504.
73	D13	I 24	3.0	27620	.225	.000	4.574	.000	.363	.000	39.	3770.	0.	3809.
105	250	I 75	126.0	39755	.264	.000	4.432	.000	.352	.000	53.	5972.	0.	6025.
105	283	I 75	125.0	42615	.245	.000	4.478	.000	.356	.000	58.	6080.	0.	6139.
8	033	I 75	181.9	169729	.095	.000	4.140	.000	.361	.000	277.	8752.	0.	9028.
8	074	I 275	4.9	51234	.099	.000	4.631	.000	.357	.000	84.	3046.	0.	3130.
8	272	I 75	180.9	138195	.103	.000	3.973	.000	.341	.000	221.	7024.	0.	7245.
19	814	I 471	2.3	87666	.026	.000	4.211	.000	.324	.000	156.	1150.	0.	1306.
25	751	I 64	95.8	43367	.138	.000	4.570	.000	.351	.000	68.	3502.	0.	3570.
34	049	I 75	114.5	78699	.221	.000	4.205	.000	.343	.000	111.	9157.	0.	9268.
34	250	I 75	108.9	61171	.166	.000	4.216	.000	.347	.000	93.	5422.	0.	5515.
34	273	I 64	85.7	34555	.154	.000	4.572	.000	.351	.000	53.	3110.	0.	3163.
34	391	I 75	112.0	82631	.209	.000	4.302	.000	.352	.000	119.	9553.	0.	9672.
34	392	I 75	110.2	72467	.140	.000	4.294	.000	.347	.000	113.	5505.	0.	5618.
34	784	I 75	116.2	76524	.225	.000	4.329	.000	.346	.000	108.	9426.	0.	9534.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS IV -- URBAN INTERSTATE

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
47	144	I 65	93.7	49929	.328	.000	4.644	.000	.377	.000	61.	10463.	0.	10524.
56	A03	I 71	4.4	56628	.067	.000	4.565	.000	.350	.000	96.	2204.	0.	2301.
56	A05	I 71	10.3	56568	.178	.000	4.730	.000	.364	.000	85.	6332.	0.	6416.
56	A07	I 71	5.9	68992	.102	.000	4.633	.000	.356	.000	113.	4247.	0.	4360.
56	A26	I 64	11.6	73217	.081	.000	4.602	.000	.355	.000	123.	3535.	0.	3658.
56	A28	I 64	9.3	81065	.089	.000	3.642	.000	.301	.000	133.	2903.	0.	3036.
56	B33	I 64	15.9	93807	.075	.000	4.538	.000	.349	.000	158.	4082.	0.	4241.
56	G23	I 65	128.9	149690	.128	.000	4.104	.000	.363	.000	237.	10458.	0.	10695.
56	M36	I 65	135.8	88268	.129	.000	4.371	.000	.355	.000	140.	6458.	0.	6598.
56	N02	I 264	2.3	48975	.062	.000	4.430	.000	.338	.000	84.	1650.	0.	1734.
56	Q04	I 264	.3	48479	.066	.000	4.489	.000	.344	.000	83.	1805.	0.	1887.
56	T98	I 64	3.0	70131	.058	.000	4.537	.000	.348	.000	121.	2355.	0.	2476.
56	222	I 64	18.2	76654	.092	.000	4.581	.000	.352	.000	127.	4134.	0.	4261.
56	777	I 264	12.9	152577	.054	.000	4.372	.000	.336	.000	263.	4460.	0.	4723.
56	787	I 65	131.9	141349	.088	.000	4.118	.000	.413	.000	234.	7732.	0.	7966.
56	791	I 264	11.6	123644	.074	.000	3.503	.000	.283	.000	209.	3306.	0.	3515.
56	795	I 264	10.5	110951	.076	.000	3.545	.000	.291	.000	187.	3180.	0.	3367.
56	801	I 264	9.4	94458	.078	.000	3.376	.000	.285	.000	158.	2596.	0.	2754.
56	805	I 264	7.8	82147	.043	.000	4.355	.000	.333	.000	143.	1877.	0.	2021.
56	863	I 64	4.0	84581	.135	.000	3.860	.000	.399	.000	132.	6431.	0.	6563.
59	792	I 75	183.9	184229	.152	.000	4.287	.000	.352	.000	283.	15445.	0.	15728.
59	801	I 75	188.2	142976	.153	.000	4.513	.000	.430	.000	220.	15509.	0.	15729.
59	804	I 275	81.4	94381	.044	.000	4.416	.000	.339	.000	165.	2285.	0.	2450.
59	805	I 275	83.1	100976	.078	.000	3.637	.000	.297	.000	169.	3118.	0.	3287.
76	610	I 75	84.7	47778	.148	.000	4.621	.000	.363	.000	74.	4342.	0.	4416.
76	637	I 75	81.1	48103	.256	.000	4.607	.000	.363	.000	65.	7529.	0.	7594.
93	311	I 71	15.5	50718	.182	.000	4.734	.000	.365	.000	76.	5820.	0.	5895.
93	313	I 71	17.6	51281	.174	.000	4.614	.000	.356	.000	77.	5354.	0.	5431.
93	329	I 71	19.7	46294	.208	.000	4.743	.000	.366	.000	67.	6104.	0.	6171.
105	250	I 75	127.7	39181	.266	.000	4.753	.000	.366	.000	52.	6624.	0.	6676.
105	283	I 75	124.8	41731	.240	.000	4.703	.000	.361	.000	58.	6202.	0.	6260.
114	563	I 65	21.2	49589	.338	.000	4.646	.000	.369	.000	60.	10473.	0.	10532.
34	250	I 75	108.9	62393	.247	.000	4.523	.000	.364	.000	86.	9265.	0.	9351.
34	392	I 75	110.2	73506	.216	.000	4.521	.000	.364	.000	105.	9526.	0.	9631.
47	144	I 65	93.6	50230	.406	.000	4.663	.000	.381	.000	54.	13227.	0.	13281.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS IV -- URBAN INTERSTATE

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
56	805	I 264	8.0	80822	.045	.000	4.332	.000	.330	.000	141.	1890.	0.	2031.

SUMMARY OF AVERAGE VALUES FOR
AGGREGATE CLASS IV -- URBAN INTERSTATE

YEAR	3 YR AVG	13	11	07	05	04	01	00	99	98	97	96
UNCLASSIFIED ROADS (ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)												
NO OF CLASSIFICATION STA.	188	71	82	35	33	42	37	1	2	4	17	19
AADT	45030	45556	45498	42863	44802	44333	45087	39600	28000	34475	66585	65325
PERCENT TRUCKS	15.373	14.318	14.865	18.703	16.289	21.823	15.927	34.331	29.140	23.285	15.710	12.515
AXLES PER TRUCK	4.405	4.267	4.431	4.624	4.270	4.341	4.012	4.683	4.563	4.470	4.185	4.033
EAL'S PER TRUCK AXLE	.296	.351	.282	.218	.268	.272	.339	.245	.241	.231	.187	.181
CLASSIFIED ROADS (MANUAL LOCATION WITH 3% OR MORE OF TRUCKS CLASSIFIED AS HEAVY/COAL)												
NO OF CLASSIFICATION STA.	1	0	0	1	0	0	0	0	0	0	0	0
AADT	97564	0	0	97564	0	0	0	0	0	0	0	0
PERCENT TRUCKS	7.655	.000	.000	7.655	.000	.000	.000	.000	.000	.000	.000	.000
PERCENT OF TRUCKS CLASSIFIED AS HEAVY/COAL	4.329	.000	.000	4.329	.000	.000	.000	.000	.000	.000	.000	.000
AXLES PER TRUCK NORMAL	4.072	.000	.000	4.072	.000	.000	.000	.000	.000	.000	.000	.000
AXLES PER TRUCK HEAVY/COAL	4.808	.000	.000	4.808	.000	.000	.000	.000	.000	.000	.000	.000
EAL'S PER TRUCK AXLE NORMAL	.257	.000	.000	.257	.000	.000	.000	.000	.000	.000	.000	.000
EAL'S PER TRUCK AXLE HEAVY/COAL	1.726	.000	.000	1.726	.000	.000	.000	.000	.000	.000	.000	.000

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS V -- URBAN OTHER FREEWAY AND EXPRESSWAYS/
URBAN OTHER PRINCIPAL ARTERIAL

						FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
			MILE		TRUCK	WITH	NORMAL	HEAVY	NORMAL	HEAVY	4-TIRE	NON-COAL	COAL	
COU	STA	ROUTE	POINT	AADT	FRACT	COAL		COAL		COAL	VEHICLES	TRUCKS	TRUCKS	TOTAL
10	A05	US 23	19.0	27833	.049	.000	3.083	.000	.204	.000	48.	316.	0.	364.
10	A21	US 60	11.0	20637	.045	.000	2.817	.000	.201	.000	36.	192.	0.	228.
10	A25	US 23	2.0	13912	.194	.000	4.249	.000	.583	.000	20.	2437.	0.	2458.
11	B37	US 127	2.0	13392	.067	.000	3.661	.000	.210	.000	23.	252.	0.	275.
11	B38	US 127	3.0	12547	.105	.000	3.606	.000	.216	.000	20.	373.	0.	394.
30	A26	KY 54	1.0	15533	.090	.000	3.896	.000	.430	.000	26.	854.	0.	880.
30	C15	US 60	16.0	28587	.108	.000	4.047	.000	.313	.000	46.	1428.	0.	1474.
30	D07	US 60	13.0	25054	.111	.000	4.035	.000	.301	.000	40.	1226.	0.	1267.
34	A27	KY1974	11.0	27614	.038	.000	3.796	.000	.850	.000	48.	1221.	0.	1269.
45	004	US 23	8.0	16060	.114	.000	3.447	.000	.259	.000	25.	598.	0.	623.
47	008	US 31	21.0	24839	.078	.000	3.711	.000	.509	.000	42.	1343.	0.	1385.
47	145	US 31	33.0	19371	.087	.000	3.214	.000	.308	.000	32.	607.	0.	639.
47	B84	WK9001	136.0	23194	.178	.000	4.116	.000	.236	.000	34.	1458.	0.	1492.
51	B86	US 60	9.0	12950	.116	.000	3.473	.000	.351	.000	21.	669.	0.	690.
59	002	KY 17	24.0	8990	.069	.000	3.018	.000	.355	.000	15.	241.	0.	256.
59	D43	KY 17	22.0	17307	.073	.000	3.360	.000	.389	.000	29.	599.	0.	628.
63	A87	KY 192	19.0	24562	.075	.000	3.398	.000	.358	.000	41.	816.	0.	857.
63	C03	US 25	.0	22213	.117	.000	3.229	.000	.210	.000	35.	641.	0.	676.
73	298	US 68	.0	7546	.095	.000	3.332	.000	.269	.000	12.	233.	0.	246.
109	A14	US 68	5.0	13919	.297	.000	2.585	.000	.151	.000	18.	588.	0.	606.
19	817	US 27	13.5	29230	.016	.000	3.837	.000	.187	.000	53.	119.	0.	172.
19	A77	US 27	18.0	7799	.058	.000	2.947	.000	.286	.000	13.	139.	0.	152.
19	A91	US 27	15.7	37354	.039	.000	4.353	.000	.221	.000	66.	507.	0.	572.
24	B19	US 41	12.1	11532	.085	.001	3.341	5.000	.387	1.103	19.	460.	2.	481.
30	B41	KY2155	4.3	7292	.145	.000	3.153	.000	.250	.000	11.	303.	0.	315.
51	A03	US 41A	13.9	18581	.079	.000	3.807	.000	.494	.000	31.	1001.	0.	1031.
54	000	KY 813	10.3	2973	.185	.000	4.611	.000	.349	.000	4.	322.	0.	327.
54	001	KY 813	10.2	1041	.098	.000	3.802	.000	.443	.000	2.	63.	0.	65.
54	253	KY 813	10.8	2212	.063	.000	2.783	.000	.256	.000	4.	36.	0.	40.
59	B67	US 25	8.6	20404	.009	.000	3.563	.000	.167	.000	37.	38.	0.	75.
59	D53	KY 17	19.4	20175	.053	.000	3.411	.000	.413	.000	35.	552.	0.	586.
59	E12	KY 17N	23.7	4274	.163	.004	3.585	5.000	.416	1.103	6.	378.	6.	390.
59	G42	US 25	5.9	24181	.034	.000	4.479	.000	.228	.000	43.	305.	0.	347.
59	H28	US 25	13.3	5332	.097	.004	4.002	5.000	.457	1.103	9.	345.	4.	358.
71	B56	US 68	12.8	7168	.130	.000	3.580	.000	.288	.000	11.	350.	0.	361.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS V -- URBAN OTHER FREEWAY AND EXPRESSWAYS/
URBAN OTHER PRINCIPAL ARTERIAL

						FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
89	D24	KY1031	1.0	5606	.048	.000	2.912	.000	.273	.000	10.	77.	0.	87.
100	B27	KY 80	21.8	16089	.089	.000	4.491	.000	.227	.000	27.	535.	0.	562.
100	B37	KY 80	19.5	21551	.062	.000	4.490	.000	.226	.000	37.	496.	0.	533.
100	B38	KY 80	20.3	18808	.074	.000	4.532	.000	.229	.000	32.	527.	0.	559.
100	B61	KY 80	20.7	16340	.089	.000	4.491	.000	.227	.000	27.	543.	0.	570.
3	A57	US 127B	3.5	19694	.110	.000	4.061	.000	.413	.000	32.	1321.	0.	1353.
5	B30	KY 90	3.5	9362	.128	.000	3.318	.000	.255	.000	14.	368.	0.	383.
5	B42	US 31E	11.7	10401	.149	.000	3.098	.000	.250	.000	15.	439.	0.	454.
7	A06	US 25E	2.2	23918	.065	.000	4.329	.000	.221	.000	41.	540.	0.	581.
8	054	KY 237	11.4	12553	.024	.000	2.905	.000	.231	.000	22.	73.	0.	96.
8	B76	KY 18	14.8	50748	.057	.000	4.298	.000	.220	.000	87.	995.	0.	1082.
8	K95	KY 18	12.3	24591	.174	.006	4.641	5.000	.543	1.103	37.	3911.	52.	4000.
9	A22	US 27	6.2	16517	.084	.000	3.681	.000	.325	.000	27.	603.	0.	630.
10	044	US 23S	.2	12477	.103	.000	4.278	.000	.430	.000	20.	864.	0.	884.
10	A54	US 23	14.1	15813	.154	.000	3.752	.000	.379	.000	24.	1259.	0.	1283.
10	B34	US 60	11.9	10200	.126	.001	4.441	5.000	.547	1.103	16.	1138.	2.	1156.
10	C41	US 60	5.4	25552	.035	.000	4.164	.000	.209	.000	45.	286.	0.	331.
11	A83	KY 34	14.5	15357	.100	.000	2.949	.000	.184	.000	25.	305.	0.	330.
15	A08	KY 61	14.0	12704	.088	.000	3.446	.000	.390	.000	21.	546.	0.	567.
18	B33	US 641	5.0	8316	.100	.000	3.777	.000	.319	.000	14.	368.	0.	381.
18	B70	KY 121	14.2	10975	.030	.000	3.048	.000	.205	.000	19.	76.	0.	95.
19	E40	US 27	22.1	6845	.263	.000	3.965	.000	.385	.000	9.	1005.	0.	1014.
24	A97	US 68	9.8	12808	.089	.000	2.931	.000	.264	.000	21.	321.	0.	342.
24	B14	US 41A	14.8	17960	.085	.001	3.502	5.000	.460	1.103	30.	898.	2.	930.
24	B37	KY1682	2.8	3974	.146	.000	2.680	.000	.153	.000	6.	87.	0.	93.
24	C44	US 68B	6.0	14776	.084	.000	3.781	.000	.335	.000	24.	572.	0.	597.
24	F01	US 41A	2.5	22352	.051	.000	4.412	.000	.222	.000	39.	405.	0.	444.
30	B41	KY2155	3.6	7681	.074	.000	3.619	.000	.207	.000	13.	156.	0.	169.
30	B76	US 60	14.6	21323	.080	.000	4.171	.000	.210	.000	36.	543.	0.	579.
34	051	US 27	11.1	15335	.102	.000	3.246	.000	.269	.000	24.	499.	0.	523.
34	D88	KY 4	16.3	66038	.040	.000	2.873	.000	.224	.000	115.	613.	0.	728.
34	E45	US 25	16.7	28660	.043	.000	3.847	.000	.189	.000	50.	325.	0.	375.
34	E46	US 27	3.0	42733	.018	.000	4.090	.000	.203	.000	77.	235.	0.	312.
34	E58	KY 4	7.9	59935	.114	.003	3.939	5.000	.239	1.103	94.	2339.	42.	2476.
37	B60	US 127	7.1	19521	.019	.000	3.576	.000	.165	.000	35.	80.	0.	115.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS V -- URBAN OTHER FREEWAY AND EXPRESSWAYS/
URBAN OTHER PRINCIPAL ARTERIAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
42	A81	US 45	17.3	3769	.056	.000	3.466	.000	.370	.000	6.	99.	0.	106.
45	E44	US 23	5.0	15282	.144	.000	2.931	.000	.270	.000	23.	634.	0.	657.
47	123	US 31B	1.0	3371	.193	.000	3.722	.000	.239	.000	5.	211.	0.	216.
47	131	US 31B	1.0	1573	.058	.000	2.613	.000	.155	.000	3.	13.	0.	16.
47	231	KY9001	135.7	9869	.118	.000	2.610	.000	.171	.000	15.	189.	0.	204.
47	241	US 31B	.3	1314	.107	.000	3.362	.000	.204	.000	2.	35.	0.	37.
47	B16	KY 61	5.0	14724	.102	.000	3.559	.000	.343	.000	24.	670.	0.	694.
47	B79	US 31B	1.8	24342	.099	.000	3.237	.000	.268	.000	39.	761.	0.	800.
47	B83	US 31W	19.4	39683	.020	.000	4.109	.000	.205	.000	71.	249.	0.	320.
47	D21	US 31W	28.9	23833	.081	.003	3.009	5.000	.359	1.103	40.	758.	12.	810.
47	D71	US 31W	24.2	35141	.032	.000	4.308	.000	.218	.000	62.	386.	0.	448.
51	754	KY 425	4.0	8939	.155	.000	4.140	.000	.328	.000	14.	687.	0.	700.
51	A02	KY 351	.3	12021	.086	.003	3.667	5.000	.446	1.103	20.	614.	6.	640.
54	063	KY9004	45.2	12665	.203	.000	4.601	.000	.234	.000	18.	1010.	0.	1028.
54	A61	KY9004	43.5	22699	.130	.000	4.416	.000	.224	.000	36.	1064.	0.	1100.
54	A91	KY9004	44.4	19273	.232	.000	3.631	.000	.264	.000	27.	1564.	0.	1591.
56	257	US 31E	1.2	25902	.064	.000	3.631	.000	.331	.000	44.	733.	0.	777.
56	501	KY 61	5.2	27842	.026	.000	3.723	.000	.181	.000	50.	176.	0.	225.
56	557	KY1020	3.3	19415	.114	.000	3.714	.000	.360	.000	31.	1085.	0.	1116.
56	593	KY 61	3.0	22716	.015	.000	3.500	.000	.163	.000	41.	73.	0.	113.
56	607	KY 61	5.3	22366	.211	.000	3.936	.000	.203	.000	32.	1372.	0.	1404.
56	785	KY 61	8.5	23597	.016	.000	3.536	.000	.165	.000	42.	79.	0.	121.
56	857	US 31E	16.2	10827	.066	.000	3.089	.000	.364	.000	18.	292.	0.	310.
56	955	US 60A	4.9	17204	.081	.001	3.634	5.000	.519	1.103	29.	956.	2.	987.
56	987	US 31W	21.1	6520	.137	.000	3.335	.000	.349	.000	10.	380.	0.	390.
56	C71	KY1865	.9	18476	.075	.000	3.231	.000	.381	.000	31.	619.	0.	650.
56	G84	KY1934	1.6	14845	.122	.000	3.588	.000	.244	.000	24.	577.	0.	600.
56	L54	US 31W	20.0	4988	.087	.000	3.861	.000	.357	.000	8.	217.	0.	225.
59	001	US 25	13.6	14983	.089	.003	2.853	5.000	.241	1.103	25.	332.	8.	365.
59	B67	US 25	8.6	21632	.014	.000	3.520	.000	.159	.000	39.	63.	0.	102.
59	C16	US 25	7.3	27850	.038	.000	4.133	.000	.208	.000	49.	336.	0.	385.
59	C71	KY1120	.2	10067	.083	.001	4.278	5.000	.478	1.103	17.	626.	2.	645.
59	E05	KY 17	23.4	4332	.143	.062	5.128	5.000	.504	1.103	6.	550.	76.	633.
59	E08	KY 8	7.0	7840	.204	.021	4.670	5.000	.572	1.103	11.	1531.	66.	1608.
59	G42	US 25	5.9	25144	.066	.000	4.960	.000	.250	.000	43.	751.	0.	793.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS V -- URBAN OTHER FREEWAY AND EXPRESSWAYS/
URBAN OTHER PRINCIPAL ARTERIAL

						FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
			MILE		TRUCK	WITH	NORMAL	HEAVY	NORMAL	HEAVY	4-TIRE	NON-COAL	COAL	
COU	STA	ROUTE	POINT	AADT	FRACT	COAL		COAL		COAL	VEHICLES	TRUCKS	TRUCKS	TOTAL
59	L36	KY 17	12.7	15620	.044	.000	4.284	.000	.554	.000	27.	597.	0.	624.
63	A65	KY 80	10.0	19724	.141	.000	4.347	.000	.411	.000	31.	1808.	0.	1839.
71	A90	US 79	11.5	8943	.156	.000	3.620	.000	.264	.000	13.	485.	0.	498.
71	B11	US 431	13.4	7683	.127	.000	3.743	.000	.280	.000	12.	372.	0.	384.
71	B55	US 68	11.0	6591	.189	.000	3.786	.000	.273	.000	9.	470.	0.	480.
73	A45	US 60	11.8	20284	.016	.000	3.889	.000	.187	.000	36.	86.	0.	122.
73	A52	US 60X	.6	12590	.019	.000	3.662	.000	.173	.000	23.	56.	0.	78.
73	B14	US 60	10.3	25288	.086	.000	4.641	.000	.236	.000	42.	870.	0.	912.
73	C01	US 45	10.6	13824	.048	.000	3.639	.000	.372	.000	24.	331.	0.	355.
73	C13	US 62	10.0	11364	.043	.000	3.896	.000	.187	.000	20.	131.	0.	151.
73	C20	US 45	6.2	21696	.049	.000	4.334	.000	.219	.000	38.	367.	0.	405.
73	C49	US 45	7.7	27754	.040	.000	4.264	.000	.216	.000	49.	373.	0.	422.
73	C82	US 60	10.1	21127	.092	.000	4.660	.000	.236	.000	35.	785.	0.	820.
76	A46	US 25	15.3	20489	.063	.003	2.949	5.000	.262	1.103	35.	363.	8.	406.
76	A65	KY 876	9.2	28709	.059	.000	3.303	.000	.370	.000	49.	750.	0.	799.
76	C44	KY 21	8.9	14459	.053	.000	3.138	.000	.421	.000	25.	371.	0.	396.
78	A31	KY 55	1.0	7112	.102	.000	3.417	.000	.231	.000	12.	209.	0.	221.
81	A95	US 62	18.0	5156	.162	.000	3.661	.000	.267	.000	8.	298.	0.	306.
90	A21	US 62	15.8	9800	.074	.000	3.142	.000	.283	.000	17.	236.	0.	252.
90	A99	US 31E	16.0	15740	.109	.000	3.355	.000	.243	.000	25.	512.	0.	538.
98	A73	US 23	24.1	26876	.080	.000	4.423	.000	.224	.000	45.	778.	0.	823.
100	A05	US 27	15.6	29787	.038	.000	4.020	.000	.200	.000	52.	331.	0.	384.
100	B27	KY 80	21.8	16071	.102	.000	4.436	.000	.226	.000	26.	599.	0.	625.
100	B31	US 27	17.1	18058	.039	.000	4.120	.000	.207	.000	32.	217.	0.	248.
100	B61	KY 80B	20.7	16514	.078	.000	4.324	.000	.218	.000	28.	443.	0.	470.
100	B90	US 27	11.3	20726	.063	.000	3.059	.000	.195	.000	35.	284.	0.	320.
106	A65	KY 55	7.3	17809	.094	.000	3.511	.000	.296	.000	29.	636.	0.	666.
106	A67	KY 53	6.6	16392	.127	.007	3.633	5.000	.377	1.103	26.	1033.	30.	1089.
114	A14	US 68	9.1	24752	.027	.000	3.747	.000	.181	.000	44.	165.	0.	209.

SUMMARY OF AVERAGE VALUES FOR
AGGREGATE CLASS V -- URBAN OTHER FREEWAY AND EXPRESSWAYS/
URBAN OTHER PRINCIPAL ARTERIAL

YEAR	3 YR AVG	13	11	07	05	04	01	00	99	98	97	96
UNCLASSIFIED ROADS (ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)												
NO OF CLASSIFICATION STA.	528	133	291	104	100	158	50	22	27	41	44	54
AADT	18333	17346	18568	18940	19072	19655	26231	25643	15462	20421	26925	21135
PERCENT TRUCKS	9.399	9.093	9.461	9.615	10.485	8.880	9.511	9.077	8.265	9.957	5.879	7.370
AXLES PER TRUCK	3.736	3.704	3.757	3.718	3.735	3.519	3.427	3.186	3.228	3.386	3.083	3.144
EAL'S PER TRUCK AXLE	.305	.293	.307	.312	.321	.304	.258	.273	.257	.243	.213	.204
CLASSIFIED ROADS (MANUAL LOCATION WITH 3% OR MORE OF TRUCKS CLASSIFIED AS HEAVY/COAL)												
NO OF CLASSIFICATION STA.	3	1	1	1	0	2	1	4	6	7	10	9
AADT	10129	4332	4332	21725	0	16301	27600	29000	23966	22371	23716	26200
PERCENT TRUCKS	14.577	14.337	14.337	15.056	.000	13.117	3.787	11.487	12.142	7.806	9.999	9.095
PERCENT OF TRUCKS CLASSIFIED AS HEAVY/COAL	5.193	6.198	6.198	3.183	.000	5.409	16.745	12.844	11.826	8.567	11.037	29.425
AXLES PER TRUCK NORMAL	4.830	5.128	5.195	4.167	.000	4.284	3.123	3.875	3.836	3.724	3.685	3.393
AXLES PER TRUCK HEAVY/COAL	4.487	5.000	4.712	3.750	.000	4.484	4.057	4.106	3.860	3.950	4.696	4.573
EAL'S PER TRUCK AXLE NORMAL	.372	.504	.319	.293	.000	.304	.221	.219	.234	.199	.203	.220
EAL'S PER TRUCK AXLE HEAVY/COAL	2.683	1.103	3.834	3.112	.000	5.068	3.028	2.919	2.948	2.650	1.143	.941

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS

2013

AGGREGATE CLASS VI -- URBAN MINOR ARTERIAL/
URBAN COLLECTOR/URBAN LOCAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
3	004	US 127	7.0	7171	.045	.000	2.747	.000	.758	.000	12.	243.	0.	256.
3	A49	KY 44	13.0	8265	.088	.000	3.529	.000	.876	.000	14.	817.	0.	831.
5	B36	KY 63	13.0	2360	.109	.000	2.800	.000	.679	.000	4.	178.	0.	182.
7	A72	KY3502	.0	3180	.015	.000	3.186	.000	.718	.000	6.	39.	0.	45.
8	054	KY 237	11.0	11263	.043	.000	2.359	.000	.765	.000	20.	317.	0.	337.
8	M12	KY1017	3.0	5784	.062	.000	2.803	.000	.852	.000	10.	315.	0.	324.
9	A50	KY1940	.0	648	.136	.000	2.213	.000	.617	.000	1.	44.	0.	45.
9	A53	US 68	3.0	5714	.123	.000	3.179	.000	.770	.000	9.	627.	0.	636.
10	005	US 60	12.0	7442	.089	.000	3.289	.000	.799	.000	12.	634.	0.	647.
10	A20	KY 168	6.0	6741	.046	.000	3.058	.000	.800	.000	12.	279.	0.	290.
11	A17	US 127	5.0	11949	.097	.000	3.096	.000	.785	.000	20.	1024.	0.	1044.
15	801	KY 61	18.0	12547	.089	.000	3.119	.000	.756	.000	21.	962.	0.	983.
15	A30	KY 480	1.0	8411	.107	.000	3.056	.000	.828	.000	14.	831.	0.	845.
15	A43	KY 44	14.0	12450	.079	.000	2.622	.000	.768	.000	21.	723.	0.	744.
15	A44	KY 44	10.0	8289	.072	.000	2.579	.000	.755	.000	14.	424.	0.	438.
15	A51	KY1494	6.0	7335	.040	.000	2.680	.000	.781	.000	13.	226.	0.	239.
17	B01	US 62	7.0	4100	.117	.000	3.120	.000	.725	.000	7.	397.	0.	404.
18	A10	KY 821	.0	4898	.086	.000	3.600	.000	.779	.000	8.	432.	0.	440.
19	A28	KY 9	22.0	9092	.077	.000	3.306	.000	.774	.000	15.	655.	0.	669.
19	A72	KY1892	.0	6421	.039	.000	4.302	.000	.896	.000	11.	352.	0.	363.
24	303	US 41	5.0	6694	.163	.000	4.107	.000	1.198	.000	10.	1959.	0.	1969.
24	A57	KY 380	1.0	8037	.077	.000	2.726	.000	.725	.000	14.	448.	0.	462.
24	B57	KY1007	1.0	5667	.064	.000	2.654	.000	.775	.000	10.	270.	0.	280.
24	C02	KY 109	10.0	958	.268	.000	2.661	.000	.848	.000	1.	212.	0.	213.
25	A49	KY 89	15.0	4572	.056	.000	2.849	.000	.858	.000	8.	228.	0.	235.
25	A93	KY1923	13.0	3395	.068	.000	2.479	.000	.727	.000	6.	151.	0.	157.
30	289	KY 54	5.0	12379	.068	.000	3.229	.000	.880	.000	21.	873.	0.	894.
30	B97	KY3143	1.0	10948	.048	.000	3.185	.000	.761	.000	19.	469.	0.	488.
34	C45	KY 57	.0	8665	.103	.000	3.465	.000	.720	.000	14.	809.	0.	823.
37	A20	KY 420	2.0	3717	.039	.000	2.780	.000	.798	.000	7.	117.	0.	124.
37	C19	KY1005	7.0	1840	.058	.000	2.410	.000	.746	.000	3.	70.	0.	73.
42	132	KY1830	.0	1040	.100	.000	2.601	.000	.923	.000	2.	91.	0.	93.
42	B44	KY 464	.0	1646	.023	.000	2.502	.000	.566	.000	3.	20.	0.	23.
43	033	KY 259	14.0	2684	.087	.000	2.703	.000	.841	.000	4.	194.	0.	199.
43	A13	KY 920	.0	1723	.129	.000	3.939	.000	1.302	.000	3.	416.	0.	418.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS

2013

AGGREGATE CLASS VI -- URBAN MINOR ARTERIAL/
URBAN COLLECTOR/URBAN LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
		MILE		TRUCK	WITH	NORMAL	HEAVY	NORMAL	HEAVY	4-TIRE	NON-COAL	COAL	
COU	STA ROUTE	POINT	AADT	FRACT	COAL		COAL		COAL	VEHICLES	TRUCKS	TRUCKS	TOTAL
47	095 KY 447	.0	4276	.049	.000	2.380	.000	.714	.000	7.	129.	0.	136.
47	B16 KY 61	5.0	14092	.102	.000	3.528	.000	1.087	.000	23.	2013.	0.	2036.
47	B48 US 31	15.0	14929	.127	.000	2.507	.000	.738	.000	24.	1279.	0.	1302.
47	B95 US 62	21.0	7795	.087	.000	2.812	.000	.881	.000	13.	617.	0.	630.
47	B96 KY 251	1.0	5444	.028	.000	2.569	.000	.694	.000	10.	98.	0.	108.
47	B98 KY3005	8.0	20693	.083	.000	2.666	.000	.751	.000	34.	1248.	0.	1282.
47	D48 KY 144	4.0	5838	.052	.000	2.697	.000	.754	.000	10.	224.	0.	234.
47	D51 KY1646	1.0	6198	.066	.000	2.857	.000	.730	.000	10.	310.	0.	320.
47	D56 KY1500	6.0	1664	.033	.000	2.436	.000	.749	.000	3.	37.	0.	40.
49	A29 US 62	9.0	2717	.117	.000	3.113	.000	.777	.000	4.	280.	0.	284.
49	A58 US 27	5.0	12511	.292	.000	2.524	.000	.860	.000	16.	2892.	0.	2908.
51	A42 KY 136	20.0	6544	.098	.000	3.170	.000	.929	.000	11.	691.	0.	702.
51	B71 KY1539	.0	1900	.053	.000	2.681	.000	.875	.000	3.	86.	0.	89.
51	C41 KY2098	.0	1231	.114	.000	2.241	.000	.660	.000	2.	76.	0.	78.
54	A47 KY1581	.0	3264	.078	.000	2.266	.000	.667	.000	5.	141.	0.	146.
54	A55 KY 70	20.0	10087	.121	.000	3.096	.000	.870	.000	16.	1201.	0.	1217.
54	B42 US 41	.0	10904	.074	.000	4.093	.000	1.060	.000	18.	1279.	0.	1297.
56	085 KY1694	.0	7335	.065	.000	2.623	.000	.719	.000	13.	329.	0.	342.
56	189 KY 146	3.0	8917	.043	.000	2.756	.000	.786	.000	16.	300.	0.	316.
56	195 KY2050	1.0	11914	.044	.000	3.055	.000	.716	.000	21.	422.	0.	443.
56	309 KY1065	10.0	10175	.033	.000	2.879	.000	.732	.000	18.	256.	0.	274.
56	393 KY2845	.0	7587	.050	.000	2.841	.000	.813	.000	13.	318.	0.	331.
56	556 KY1065	1.0	12774	.050	.000	2.773	.000	.763	.000	22.	493.	0.	515.
56	563 KY1020	.0	4015	.072	.000	2.626	.000	.833	.000	7.	230.	0.	237.
56	584 KY1450	2.0	11179	.065	.000	2.969	.000	.850	.000	19.	675.	0.	693.
56	704 KY2055	3.0	3172	.055	.000	2.412	.000	.797	.000	5.	123.	0.	129.
56	820 KY2056	1.0	6021	.167	.000	3.604	.000	1.154	.000	9.	1527.	0.	1537.
56	A54 KY1819	11.0	9530	.084	.000	3.015	.000	.879	.000	16.	770.	0.	786.
56	M19 US 60	1.0	13768	.073	.000	3.243	.000	.932	.000	23.	1105.	0.	1128.
56	S16 KY 61	11.0	4266	.098	.000	3.057	.000	.907	.000	7.	421.	0.	428.
57	A01 KY 169	11.0	6183	.058	.000	2.213	.000	.664	.000	11.	193.	0.	204.
57	A24 US 27	1.0	15756	.088	.000	3.105	.000	.764	.000	26.	1201.	0.	1227.
59	D65 KY 236	1.0	9044	.048	.000	2.829	.000	.708	.000	16.	315.	0.	331.
59	H45 KY2373	.0	9833	.055	.000	3.551	.000	.988	.000	17.	698.	0.	715.
59	L25 KY 536	3.0	7639	.061	.000	2.481	.000	.734	.000	13.	312.	0.	325.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS

2013

AGGREGATE CLASS VI -- URBAN MINOR ARTERIAL/
URBAN COLLECTOR/URBAN LOCAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
59	L45	KY1486	.0	1076	.045	.000	2.500	.000	.793	.000	2.	35.	0.	37.
63	A06	KY 80	11.0	7592	.036	.000	3.099	.000	.764	.000	13.	235.	0.	248.
63	A40	KY1769	.0	1308	.072	.000	2.510	.000	.855	.000	2.	74.	0.	76.
71	A61	US 68	3.0	8289	.134	.000	3.324	.000	.922	.000	13.	1242.	0.	1255.
73	A01	KY 284	.0	3658	.036	.000	2.803	.000	.706	.000	6.	95.	0.	101.
73	C53	KY1310	.0	3078	.026	.000	2.721	.000	.730	.000	5.	59.	0.	64.
76	A28	US 25	1.0	14496	.090	.000	3.495	.000	.794	.000	24.	1319.	0.	1343.
76	C47	KY 595	1.0	2560	.050	.000	2.656	.000	.738	.000	4.	91.	0.	95.
78	A23	KY 55	.0	6492	.049	.000	3.208	.000	1.043	.000	11.	386.	0.	397.
78	A67	US 68	12.0	6891	.107	.000	3.240	.000	.693	.000	11.	602.	0.	614.
81	A67	US 62	16.0	11312	.060	.000	3.099	.000	.875	.000	19.	667.	0.	686.
81	C13	US 68	.0	4954	.104	.000	3.373	.000	1.079	.000	8.	681.	0.	689.
84	A24	US 127	4.0	11699	.088	.000	3.056	.000	.723	.000	19.	831.	0.	851.
87	B05	KY 713	6.0	2751	.060	.000	2.453	.000	.821	.000	5.	122.	0.	127.
87	B36	KY 686	4.0	8044	.072	.000	2.508	.000	.707	.000	14.	373.	0.	387.
93	267	KY 362	2.0	1409	.057	.000	2.395	.000	.605	.000	2.	43.	0.	45.
93	287	KY 53	4.0	9116	.054	.000	2.802	.000	.775	.000	16.	391.	0.	406.
93	508	US 42	.0	15324	.081	.000	3.283	.000	.821	.000	26.	1215.	0.	1241.
93	B19	KY 329	.0	7791	.067	.000	2.860	.000	.749	.000	13.	408.	0.	421.
98	619	KY1460	2.0	3952	.033	.000	2.423	.000	.669	.000	7.	77.	0.	84.
98	A84	KY3496	.0	6751	.249	.000	3.118	.000	.456	.000	9.	871.	0.	880.
100	A11	KY1247	6.0	11284	.072	.000	3.328	.000	.806	.000	19.	799.	0.	818.
100	B53	KY 769	9.0	3941	.037	.000	3.050	.000	.882	.000	7.	143.	0.	150.
100	B75	KY1642	5.0	3592	.052	.000	2.626	.000	.804	.000	6.	145.	0.	151.
100	B79	KY3057	.0	1115	.032	.000	2.655	.000	.783	.000	2.	27.	0.	29.
105	292	KY 32	.0	8207	.043	.000	2.512	.000	.717	.000	14.	230.	0.	244.
106	A57	KY 53	8.0	6237	.050	.000	2.711	.000	.774	.000	11.	237.	0.	247.
107	A51	KY1008	4.0	4165	.130	.000	3.839	.000	1.101	.000	6.	838.	0.	845.
107	A95	KY3498	.0	2203	.038	.000	3.185	.000	.948	.000	4.	93.	0.	96.
114	A13	US 68	.0	16007	.076	.000	3.396	.000	.768	.000	27.	1160.	0.	1187.
116	A54	KY 90	.0	6514	.050	.000	3.089	.000	.774	.000	11.	283.	0.	294.
118	A54	US 25	32.0	7860	.040	.000	3.059	.000	.858	.000	14.	305.	0.	318.
120	A42	KY1659	.0	1383	.034	.000	2.445	.000	.647	.000	2.	27.	0.	30.
7	A79	KY2396	.4	2064	.027	.000	2.230	.000	.786	.000	4.	36.	0.	39.
8	040	KY 338	26.6	5299	.073	.000	2.455	.000	.681	.000	9.	236.	0.	245.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS VI -- URBAN MINOR ARTERIAL/
URBAN COLLECTOR/URBAN LOCAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
8	302	US 42	10.0	12146	.063	.000	2.593	.000	.706	.000	21.	510.	0.	530.
8	C66	KY 212	.4	22333	.009	.000	4.254	.000	1.407	.000	40.	448.	0.	488.
8	G89	KY 237	5.9	15284	.037	.000	2.961	.000	.737	.000	27.	445.	0.	472.
8	K03	KY3076	.2	4886	.095	.000	2.786	.000	.854	.000	8.	404.	0.	412.
8	K22	KY1829	1.7	13339	.075	.003	3.651	5.000	.940	2.275	22.	1255.	12.	1289.
8	K25	US 25	9.1	13273	.074	.000	3.200	.000	.868	.000	22.	996.	0.	1018.
17	A35	KY 91	11.2	3755	.184	.000	2.917	.000	.839	.000	6.	618.	0.	623.
19	776	KY 10	.9	1684	.035	.000	2.422	.000	.792	.000	3.	42.	0.	45.
19	A53	KY 8	5.0	1459	.058	.000	2.716	.000	.714	.000	2.	59.	0.	62.
19	B20	KY 445	.1	3569	.043	.000	2.767	.000	.657	.000	6.	102.	0.	108.
24	A84	KY 507	.8	2769	.058	.000	2.628	.000	.586	.000	5.	89.	0.	93.
24	B29	KY 109	11.8	2589	.143	.000	2.288	.000	.689	.000	4.	212.	0.	216.
24	C05	KY1979	.1	2517	.068	.000	2.518	.000	.747	.000	4.	117.	0.	122.
51	B21	KY 285	2.1	3726	.037	.000	2.634	.000	.733	.000	6.	98.	0.	104.
51	B60	US 60	11.0	16188	.071	.002	3.329	5.000	.766	2.275	27.	1067.	12.	1107.
51	B69	KY 351	3.0	5520	.029	.000	2.933	.000	.733	.000	10.	126.	0.	135.
51	C17	KY 812	7.0	4608	.039	.000	2.818	.000	.752	.000	8.	138.	0.	146.
51	C32	KY 136	18.0	2831	.084	.000	3.675	.000	.864	.000	5.	275.	0.	279.
54	A52	KY 336	4.1	2846	.050	.000	3.064	.000	.801	.000	5.	127.	0.	132.
59	B57	KY1303	5.6	36504	.085	.000	4.449	.000	1.484	.000	61.	7518.	0.	7579.
59	B72	KY 371	2.1	19986	.008	.000	3.907	.000	1.312	.000	36.	301.	0.	338.
59	D56	KY 371	.7	8866	.032	.000	3.043	.000	.822	.000	16.	262.	0.	277.
59	H00	KY1072	2.7	6336	.044	.000	3.153	.000	.888	.000	11.	288.	0.	299.
59	L29	KY 16	3.4	6019	.075	.000	2.700	.000	.745	.000	10.	333.	0.	343.
59	L40	KY1829	4.2	5012	.053	.000	2.472	.000	.813	.000	9.	195.	0.	204.
63	074	US 25	9.4	17847	.104	.002	3.174	5.000	.859	2.275	29.	1843.	12.	1884.
89	D44	US 62	16.8	9312	.117	.000	4.325	.000	1.222	.000	15.	2099.	0.	2114.
100	A24	KY1247	8.0	7596	.074	.000	3.286	.000	.870	.000	13.	590.	0.	602.
100	A64	KY2292	.8	6605	.034	.000	2.982	.000	.800	.000	12.	199.	0.	210.
100	B04	KY2227	.2	5617	.084	.000	2.771	.000	.821	.000	9.	394.	0.	403.
100	B50	KY 80X	4.1	7837	.074	.003	3.416	5.000	.765	2.275	13.	551.	8.	573.
100	B81	KY1247	5.4	6350	.068	.000	3.239	.000	.832	.000	10.	424.	0.	434.
116	A22	KY 92	8.1	5971	.091	.000	2.896	.000	.778	.000	10.	450.	0.	460.
116	A70	KY3106	.3	3263	.050	.000	3.034	.000	.812	.000	6.	146.	0.	151.
118	A36	KY 312	1.8	5137	.079	.000	2.473	.000	.694	.000	9.	253.	0.	262.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS VI -- URBAN MINOR ARTERIAL/
URBAN COLLECTOR/URBAN LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
			MILE		TRUCK	WITH	NORMAL	HEAVY	NORMAL	HEAVY	4-TIRE	NON-COAL	COAL	
COU	STA	ROUTE	POINT	AADT	FRACT	COAL		COAL		COAL	VEHICLES	TRUCKS	TRUCKS	TOTAL
118	A65	KY1259	.3	2236	.048	.000	2.240	.000	.646	.000	4.	56.	0.	60.
118	A66	KY 727	2.5	3591	.035	.000	2.742	.000	.839	.000	6.	107.	0.	114.
118	D04	KY 856	1.2	262	.007	.000	2.059	.000	1.254	.000	0.	2.	0.	2.
3	A38	US 127	6.0	5055	.144	.000	3.985	.000	1.324	.000	8.	1406.	0.	1414.
3	A49	KY 44	13.4	10840	.082	.000	3.258	.000	.823	.000	18.	864.	0.	882.
3	A65	KY3359	.5	4101	.049	.000	2.638	.000	.641	.000	7.	124.	0.	131.
5	A81	US 31E	.4	6970	.051	.002	3.289	5.000	.815	2.275	12.	347.	4.	363.
5	A99	US 31E	2.0	8811	.100	.000	2.766	.000	.713	.000	14.	632.	0.	646.
5	B43	KY1307	8.5	956	.079	.000	3.309	.000	.671	.000	2.	63.	0.	65.
5	D46	KY3160	2.0	3532	.086	.000	3.040	.000	.684	.000	6.	232.	0.	238.
7	A48	KY 74	15.8	12552	.038	.000	3.377	.000	.755	.000	22.	447.	0.	469.
7	A92	KY2401	.7	2339	.016	.000	3.128	.000	.878	.000	4.	39.	0.	43.
7	B63	KY 441	1.1	2134	.046	.000	2.944	.000	.601	.000	4.	63.	0.	67.
8	292	KY3060	1.0	5261	.032	.000	3.720	.000	.821	.000	9.	188.	0.	198.
8	315	KY 16	2.8	4129	.076	.000	3.448	.000	.755	.000	7.	299.	0.	306.
8	C65	KY 236	2.2	7745	.101	.000	3.539	.000	.802	.000	13.	814.	0.	827.
8	C70	KY 212	1.0	10962	.158	.000	3.186	.000	1.021	.000	17.	2058.	0.	2075.
8	H53	KY1829	.5	15680	.123	.000	3.475	.000	1.012	.000	25.	2485.	0.	2510.
8	K16	US 25	10.3	15994	.036	.000	3.412	.000	1.014	.000	28.	736.	0.	764.
8	K42	KY 18	16.5	15382	.012	.000	3.971	.000	1.298	.000	28.	359.	0.	387.
8	L04	KY 20	17.4	7738	.062	.000	3.863	.000	.925	.000	13.	625.	0.	638.
8	M21	KY3076	.9	15074	.107	.003	4.546	5.000	.990	2.275	24.	2638.	17.	2679.
10	B55	KY 168	.2	2037	.034	.000	2.296	.000	.564	.000	4.	33.	0.	37.
10	C17	KY 766	1.7	3164	.056	.000	2.935	.000	.914	.000	5.	174.	0.	180.
10	C48	KY3294	.5	2650	.040	.000	2.508	.000	.600	.000	5.	58.	0.	63.
11	A24	US 127	6.5	6121	.048	.000	2.505	.000	.752	.000	11.	202.	0.	213.
11	A27	KY 33	1.0	6305	.042	.000	2.564	.000	.705	.000	11.	174.	0.	185.
15	769	KY1526	13.3	6651	.079	.000	2.791	.000	.725	.000	11.	388.	0.	399.
15	803	KY1450	2.2	10766	.082	.000	2.987	.000	.900	.000	18.	864.	0.	882.
15	A51	KY1494	7.4	6765	.052	.000	2.606	.000	.733	.000	12.	243.	0.	255.
15	C37	US 31X	.4	2616	.063	.000	2.720	.000	.717	.000	4.	117.	0.	122.
17	A02	KY 293	7.0	2890	.071	.000	2.861	.000	.836	.000	5.	180.	0.	185.
17	A92	KY 293	4.5	2937	.054	.000	2.910	.000	.814	.000	5.	137.	0.	143.
18	A43	KY2594	.3	4033	.112	.000	3.107	.000	.929	.000	6.	477.	0.	484.
19	770	KY 709	.1	7381	.061	.000	3.705	.000	.904	.000	13.	550.	0.	562.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS VI -- URBAN MINOR ARTERIAL/
URBAN COLLECTOR/URBAN LOCAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
19	A22	KY1120	3.5	5791	.030	.000	3.989	.000	.843	.000	10.	214.	0.	224.
19	A32	KY1632	.5	6018	.053	.000	3.368	.000	.765	.000	10.	298.	0.	308.
24	A01	KY 272	9.0	6037	.055	.000	3.012	.000	.863	.000	10.	317.	0.	327.
24	A50	US 41	14.0	8131	.151	.000	2.433	.000	.734	.000	12.	800.	0.	813.
24	C02	KY 109	10.5	979	.097	.000	3.312	.000	1.115	.000	2.	127.	0.	128.
24	C41	KY2544	.2	1926	.069	.000	2.888	.000	.843	.000	3.	118.	0.	121.
24	F02	KY 911	.9	8025	.025	.000	3.373	.000	.839	.000	14.	210.	0.	224.
24	F07	KY 115	1.0	6703	.027	.000	2.999	.000	.751	.000	12.	149.	0.	161.
24	F09	KY 400	.4	5819	.023	.000	3.182	.000	.688	.000	10.	109.	0.	119.
30	065	KY1456	4.7	4516	.047	.000	2.971	.000	.847	.000	8.	195.	0.	202.
30	A27	KY 298	8.3	6990	.043	.000	2.844	.000	.634	.000	12.	199.	0.	212.
30	A32	KY2698	2.8	13182	.107	.004	3.375	5.000	.772	2.275	21.	1334.	25.	1380.
30	B77	KY 331	.5	4895	.191	.000	3.787	.000	1.027	.000	7.	1331.	0.	1338.
30	C32	KY2707	1.0	3338	.092	.000	2.241	.000	.642	.000	6.	161.	0.	167.
30	C81	KY3067	.6	1142	.066	.000	2.536	.000	.766	.000	2.	54.	0.	56.
30	C86	KY 81	11.4	13994	.127	.000	2.845	.000	.845	.000	22.	1566.	0.	1589.
34	043	KY1973	17.0	2318	.089	.000	2.391	.000	.649	.000	4.	118.	0.	122.
34	053	KY1970	1.7	1149	.055	.000	2.429	.000	.637	.000	2.	36.	0.	37.
34	283	KY1973	10.0	1420	.058	.000	2.570	.000	.670	.000	2.	52.	0.	54.
34	E53	KY1681	4.0	3254	.070	.000	2.636	.000	.677	.000	5.	148.	0.	154.
34	E60	KY1977	6.0	6180	.055	.000	2.586	.000	.726	.000	11.	234.	0.	244.
34	G07	KY1978	1.8	1784	.020	.000	2.267	.000	.626	.000	3.	19.	0.	22.
42	B05	US 45	16.0	2042	.134	.000	3.193	.000	1.091	.000	3.	350.	0.	353.
42	B72	US 45B	.5	5385	.127	.000	3.591	.000	1.000	.000	8.	895.	0.	904.
43	276	KY1214	12.4	1601	.095	.000	2.522	.000	.750	.000	3.	105.	0.	108.
43	A13	KY 920	1.0	1848	.145	.000	2.776	.000	.531	.000	3.	144.	0.	147.
43	A18	US 62	21.1	12202	.119	.002	3.924	5.000	.956	2.275	20.	1979.	12.	2011.
43	A45	KY 259	12.4	14416	.184	.000	3.590	.000	.858	.000	21.	2987.	0.	3008.
43	A51	KY3155	.3	9492	.055	.000	3.418	.000	1.111	.000	16.	728.	0.	744.
45	E04	KY 750	1.0	4679	.042	.000	2.795	.000	.798	.000	8.	161.	0.	169.
45	E37	KY1725	.7	4185	.032	.000	2.774	.000	.718	.000	7.	99.	0.	106.
45	E46	KY 244	2.0	2169	.049	.000	2.562	.000	.815	.000	4.	82.	0.	85.
47	095	KY 447	1.4	3863	.042	.000	2.576	.000	.796	.000	7.	120.	0.	127.
47	121	US 62	17.0	1620	.069	.000	2.610	.000	.758	.000	3.	80.	0.	83.
47	133	US 62	17.7	2741	.224	.000	3.848	.000	1.259	.000	4.	1086.	0.	1089.

2013

AGGREGATE CLASS VI -- URBAN MINOR ARTERIAL/
URBAN COLLECTOR/URBAN LOCAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
47	212	US 31B	.1	7838	.135	.000	3.359	.000	1.151	.000	12.	1490.	0.	1503.
47	A09	US 31W	17.7	22225	.046	.000	4.121	.000	.958	.000	39.	1487.	0.	1525.
47	B22	KY3005	.1	9589	.130	.000	3.769	.000	1.042	.000	15.	1783.	0.	1798.
47	B55	KY1357	16.5	8773	.032	.000	2.880	.000	.826	.000	15.	246.	0.	261.
47	B57	KY 251	3.5	5323	.044	.000	2.963	.000	.618	.000	9.	156.	0.	165.
47	C06	KY3005	4.8	20159	.054	.000	3.892	.000	1.010	.000	35.	1567.	0.	1602.
47	G06	KY 1500	2.6	2873	.056	.000	3.040	.000	.755	.000	5.	134.	0.	139.
47	G08	KY 144	.8	6846	.061	.000	2.871	.000	.789	.000	12.	346.	0.	358.
47	G17	KY 391	2.7	574	.046	.000	2.028	.000	.907	.000	1.	18.	0.	19.
49	A81	KY 356	14.4	3837	.074	.000	3.803	.000	.830	.000	6.	327.	0.	333.
51	771	KY1299	9.2	413	.037	.000	2.427	.000	.752	.000	1.	10.	0.	11.
51	B27	US 41	12.4	7326	.086	.000	3.025	.000	.861	.000	12.	595.	0.	607.
51	C23	KY2084	.3	4566	.063	.000	3.153	.000	.902	.000	8.	296.	0.	304.
54	A38	KY1074	.3	1120	.031	.000	3.534	.000	.529	.000	2.	25.	0.	26.
54	A50	KY 262	3.4	1986	.059	.000	2.496	.000	.689	.000	3.	74.	0.	77.
54	A54	KY 481	1.8	3259	.041	.000	2.736	.000	.688	.000	6.	91.	0.	97.
54	A62	KY 254	1.1	1961	.027	.000	2.608	.000	.606	.000	3.	31.	0.	34.
54	B61	KY3052	.4	3273	.022	.000	2.917	.000	.741	.000	6.	58.	0.	64.
56	156	KY1932	4.4	16374	.058	.000	3.660	.000	.951	.000	28.	1216.	0.	1244.
56	234	KY2840	.1	6346	.028	.000	2.779	.000	.779	.000	11.	140.	0.	151.
56	269	KY 864	2.9	7021	.042	.000	2.833	.000	.778	.000	12.	236.	0.	248.
56	278	KY2053	1.8	2447	.045	.000	2.760	.000	.741	.000	4.	81.	0.	86.
56	426	KY2251	.1	6169	.056	.000	3.226	.000	.848	.000	11.	343.	0.	354.
56	479	KY3084	.6	11725	.109	.000	3.026	.000	.764	.000	19.	1075.	0.	1094.
56	552	KY1865	.5	14265	.068	.000	3.064	.000	.820	.000	24.	887.	0.	911.
56	642	KY 907	6.6	15843	.070	.003	3.266	5.000	.783	2.275	27.	1037.	17.	1080.
56	653	KY1931	2.2	15345	.044	.000	3.606	.000	.905	.000	27.	796.	0.	823.
56	752	KY1819	9.7	9436	.052	.000	3.396	.000	.959	.000	16.	578.	0.	594.
56	849	KY2051	4.4	4344	.146	.000	3.101	.000	1.000	.000	7.	719.	0.	726.
56	899	US 60A	7.9	19073	.123	.010	4.799	5.000	.991	2.275	30.	4013.	95.	4139.
56	A21	KY 22	1.2	12467	.035	.000	3.534	.000	.896	.000	22.	509.	0.	530.
56	G05	KY1727	3.2	11694	.063	.000	3.037	.000	.802	.000	20.	657.	0.	676.
56	G27	KY1931	7.4	14271	.066	.000	2.960	.000	.797	.000	24.	817.	0.	841.
56	K17	KY 61	10.9	8908	.057	.000	2.807	.000	.850	.000	15.	441.	0.	456.
56	L29	KY3082	.7	5187	.045	.000	2.685	.000	.961	.000	9.	221.	0.	230.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS VI -- URBAN MINOR ARTERIAL/
URBAN COLLECTOR/URBAN LOCAL

					FRACT OF TRK	AXLES/TRUCK		EAL'S/AXLE		2-DIRECTION EAL'S IN 1000'S			
COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	WITH COAL	NORMAL	HEAVY COAL	NORMAL	HEAVY COAL	4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
57	A19	KY2332	.1	6418	.079	.000	2.812	.000	.707	.000	11.	367.	0.	378.
59	B50	KY1501	.3	8800	.019	.000	3.903	.000	.811	.000	16.	192.	0.	208.
59	B57	KY1303	5.6	38860	.022	.000	4.305	.000	1.433	.000	69.	1921.	0.	1990.
59	B59	KY 8	.3	3744	.050	.000	3.311	.000	.731	.000	6.	165.	0.	172.
59	C61	KY 16	9.0	9697	.034	.000	3.610	.000	.779	.000	17.	336.	0.	353.
59	D62	KY 177	18.8	2661	.062	.000	3.668	.000	1.124	.000	4.	248.	0.	253.
59	H30	KY1072	1.1	18759	.057	.007	3.893	5.000	.949	2.275	32.	1420.	29.	1482.
59	L47	KY2044	.3	1443	.021	.000	3.126	.000	.742	.000	3.	25.	0.	28.
59	M15	KY3070	.4	8964	.065	.000	3.648	.000	.845	.000	15.	653.	0.	669.
61	A19	KY1629	.6	841	.032	.000	2.121	.000	.804	.000	1.	17.	0.	18.
61	A21	KY 312	.5	12772	.058	.000	3.537	.000	.821	.000	22.	781.	0.	803.
63	A31	KY 229	11.8	4304	.110	.000	2.797	.000	.745	.000	7.	359.	0.	366.
63	A35	US 25	10.8	11372	.050	.000	3.604	.000	.768	.000	20.	570.	0.	589.
63	A43	KY1006	6.2	8707	.032	.000	3.012	.000	.759	.000	15.	232.	0.	247.
63	A77	US 25	13.9	12287	.066	.000	3.862	.000	.928	.000	21.	1064.	0.	1085.
63	C23	US 25W	.8	7680	.046	.000	2.970	.000	.755	.000	13.	287.	0.	301.
71	A02	KY 178	8.5	3655	.038	.000	2.469	.000	.677	.000	6.	84.	0.	91.
71	A75	KY 79	.1	2170	.053	.000	2.946	.000	.783	.000	4.	97.	0.	101.
71	B10	US 68X	.4	4656	.139	.000	2.578	.000	.756	.000	7.	461.	0.	468.
71	B30	US 431X	1.0	8839	.059	.000	3.049	.000	.818	.000	15.	472.	0.	487.
73	086	KY 450	7.2	1987	.067	.000	2.589	.000	.727	.000	3.	92.	0.	95.
73	A58	US 45X	.6	6692	.065	.000	3.796	.000	.908	.000	11.	549.	0.	560.
73	B24	US 62	10.0	10293	.036	.000	4.151	.000	1.337	.000	18.	748.	0.	766.
73	C88	KY 998	3.1	4559	.095	.000	2.942	.000	.757	.000	7.	352.	0.	360.
76	781	KY1156	1.0	1622	.069	.000	2.623	.000	.683	.000	3.	73.	0.	75.
76	A82	KY 169	1.1	5465	.042	.000	2.553	.000	.774	.000	9.	164.	0.	173.
76	B10	US 25X	3.0	11983	.037	.000	2.921	.000	.862	.000	21.	412.	0.	433.
76	C64	KY1983	.3	2463	.169	.000	2.236	.000	.714	.000	4.	242.	0.	246.
76	D11	KY1983	1.4	688	.089	.000	2.670	.000	.686	.000	1.	41.	0.	43.
78	A78	KY2154	.6	3674	.076	.000	3.979	.000	1.248	.000	6.	504.	0.	510.
78	A82	KY2154	2.4	5617	.154	.000	3.722	.000	1.084	.000	9.	1268.	0.	1277.
78	A91	KY2154	5.9	663	.136	.000	3.807	.000	1.267	.000	1.	158.	0.	159.
81	022	KY1236	1.5	1553	.031	.000	2.705	.000	.644	.000	3.	32.	0.	34.
81	A32	KY 8	12.2	3861	.141	.000	3.719	.000	.678	.000	6.	502.	0.	508.
90	260	KY 605	10.1	4288	.048	.000	2.398	.000	.704	.000	7.	127.	0.	134.

EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATION STATIONS
2013
AGGREGATE CLASS VI -- URBAN MINOR ARTERIAL/
URBAN COLLECTOR/URBAN LOCAL

COU	STA	ROUTE	MILE POINT	AADT	TRUCK FRACT	FRACT OF TRK WITH COAL	AXLES/TRUCK NORMAL	AXLES/TRUCK HEAVY COAL	EAL'S/AXLE NORMAL	EAL'S/AXLE HEAVY COAL	2-DIRECTION EAL'S IN 1000'S 4-TIRE VEHICLES	NON-COAL TRUCKS	COAL TRUCKS	TOTAL
90	A61	US 150	.1	12988	.138	.000	3.616	.000	.742	.000	20.	1755.	0.	1775.
90	A78	US 62	14.2	8082	.119	.000	3.281	.000	.699	.000	13.	808.	0.	821.
93	B01	KY 146	.3	12820	.057	.000	3.529	.000	.907	.000	22.	856.	0.	878.
93	B18	KY 146	5.1	7857	.057	.000	2.918	.000	.789	.000	13.	376.	0.	389.
98	A75	KY1384	5.5	2366	.049	.000	2.609	.000	.642	.000	4.	72.	0.	76.
100	A22	KY 39	.1	2752	.038	.000	2.979	.000	.743	.000	5.	85.	0.	90.
100	A67	KY1577	3.8	7845	.028	.000	3.269	.000	.876	.000	14.	233.	0.	247.
100	B03	KY1575	.2	4975	.093	.000	2.947	.000	.853	.000	8.	425.	0.	433.
100	B04	KY2227	.2	5617	.091	.000	2.888	.000	.817	.000	9.	440.	0.	449.
100	B79	KY3057	.6	1153	.033	.000	2.946	.000	.849	.000	2.	36.	0.	38.
100	C08	KY1247	4.4	4133	.093	.000	3.273	.000	.948	.000	7.	436.	0.	443.
103	A50	US 60	9.3	14218	.082	.000	2.620	.000	.774	.000	24.	865.	0.	888.
105	012	KY 620	1.8	7704	.063	.000	2.764	.000	.772	.000	13.	376.	0.	389.
105	292	KY 32	1.0	8241	.068	.000	2.944	.000	.674	.000	14.	403.	0.	417.
107	A66	KY 100	9.5	3956	.095	.000	3.149	.000	.668	.000	6.	289.	0.	296.
107	A77	KY 73	9.0	2756	.063	.000	2.921	.000	.735	.000	5.	136.	0.	140.
107	B16	KY 100	10.3	3061	.057	.000	3.046	.000	.916	.000	5.	176.	0.	181.
109	A13	KY3183	3.5	6250	.162	.000	3.732	.000	.669	.000	9.	922.	0.	932.
109	A50	KY 289	.4	7088	.078	.000	2.600	.000	.774	.000	12.	406.	0.	418.
109	A91	US 68	4.2	10209	.070	.000	3.316	.000	.852	.000	17.	738.	0.	756.
114	A36	KY 234	12.4	20433	.025	.000	3.844	.000	1.252	.000	36.	885.	0.	921.
114	A44	US 231X	3.5	12945	.062	.000	3.164	.000	.782	.000	22.	721.	0.	743.
114	A47	KY2665	6.3	4597	.055	.000	3.188	.000	.697	.000	8.	205.	0.	213.
114	B43	US 31W	11.1	19792	.041	.000	3.513	.000	.937	.000	35.	977.	0.	1011.
114	B48	US 31W	18.5	20566	.082	.000	4.004	.000	1.061	.000	34.	2608.	0.	2642.
114	B68	KY 880	2.1	13892	.080	.001	3.979	5.000	.912	2.275	23.	1481.	4.	1508.
114	C23	US 31W	8.7	20111	.143	.002	3.698	5.000	.877	2.275	31.	3395.	29.	3455.
114	C30	KY2158	1.0	4762	.070	.000	2.694	.000	.782	.000	8.	258.	0.	266.
114	C31	US 31W	12.2	18691	.010	.000	3.707	.000	1.183	.000	34.	291.	0.	325.
114	C89	US 68	9.8	19477	.053	.000	2.661	.000	.871	.000	34.	868.	0.	902.
116	A66	KY 90X	3.3	11229	.111	.003	2.766	5.000	.779	2.275	18.	979.	17.	1013.
118	B37	KY1259	3.7	1947	.037	.000	2.462	.000	.715	.000	3.	47.	0.	50.
118	D37	KY2386	.7	6772	.034	.000	3.236	.000	.735	.000	12.	201.	0.	212.
120	A06	KY 33	12.0	10718	.082	.000	2.948	.000	.863	.000	18.	818.	0.	836.

SUMMARY OF AVERAGE VALUES FOR
AGGREGATE CLASS VI -- URBAN MINOR ARTERIAL/
URBAN COLLECTOR/URBAN LOCAL

YEAR	3 YR AVG	13	11	07	05	04	01	00	99	98	97	96
UNCLASSIFIED ROADS (ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)												
NO OF CLASSIFICATION STA.	980	314	489	177	159	222	32	17	17	32	36	39
AADT	7370	7259	7363	7587	6745	7713	10619	11235	8809	10994	13803	11221
PERCENT TRUCKS	7.862	7.137	8.002	8.764	8.838	8.750	7.631	7.463	7.294	11.278	5.230	4.847
AXLES PER TRUCK	3.088	3.043	3.107	3.117	3.028	3.004	3.016	3.125	2.933	3.095	2.731	2.741
EAL'S PER TRUCK AXLE	.821	.825	.819	.821	.799	.771	.800	.821	.211	.145	.160	.150
CLASSIFIED ROADS (MANUAL LOCATION WITH 3% OR MORE OF TRUCKS CLASSIFIED AS HEAVY/COAL)												
NO OF CLASSIFICATION STA.	4	0	3	1	0	0	0	0	0	2	1	2
AADT	11912	0	9890	17978	0	0	0	0	0	22100	16100	8575
PERCENT TRUCKS	18.914	.000	13.592	34.880	.000	.000	.000	.000	.000	13.201	3.053	3.124
PERCENT OF TRUCKS CLASSIFIED AS HEAVY/COAL	4.280	.000	4.232	4.425	.000	.000	.000	.000	.000	15.310	13.688	53.480
AXLES PER TRUCK NORMAL	4.174	.000	4.451	3.341	.000	.000	.000	.000	.000	3.857	2.875	2.980
AXLES PER TRUCK HEAVY/COAL	5.000	.000	5.000	5.000	.000	.000	.000	.000	.000	5.041	4.955	3.647
EAL'S PER TRUCK AXLE NORMAL	.850	.000	.850	.849	.000	.000	.000	.000	.000	.120	.134	.163
EAL'S PER TRUCK AXLE HEAVY/COAL	2.275	.000	2.275	2.275	.000	.000	.000	.000	.000	1.610	1.083	.350

6.0 AGGREGATE CLASS REGRESSION ESTIMATES

AGGREGATE CLASS I -- RURAL INTERSTATE
AVERAGE VALUES (SMOOTHED)

YEAR	ANNUAL CHANGE (%)	13	11	07	05	04	01	00	99	98	97
UNCLASSIFIED ROADS (ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)											
NO. OF CLASSIFICATION STA.		67	68	33	42	36	24	10	9	17	20
AADT	2.099	34843	34111	33380	32649	31917	31186	30455	29723	28992	28260
PERCENT TRUCKS	? -2.502 ?	27.451	28.137	28.824	29.511	30.198	30.885	31.572	32.259	32.946	33.632
AXLES PER TRUCK	? -.019 ?	4.516	4.517	4.518	4.518	4.519	4.520	4.521	4.522	4.523	4.524
EAL'S PER TRUCK AXLE	3.033	.297	.288	.279	.270	.261	.252	.243	.234	.225	.216
CLASSIFIED ROADS (MANUAL LOCATION WITH 3% OR MORE OF TRUCKS CLASSIFIED AS HEAVY/COAL)											
NO. OF CLASSIFICATION STA.		0	0	0	0	0	1	0	0	0	0
AADT	.000	**********	**********	**********	**********	**********	21300	**********	**********	**********	**********
PERCENT TRUCKS	.000	**********	**********	**********	**********	**********	23.543	**********	**********	**********	**********
PERCENT OF TRUCKS CLASSIFIED AS HEAVY/COAL	.000	**********	**********	**********	**********	**********	3.878	**********	**********	**********	**********
AXLES PER TRUCK NORMAL	.000	**********	**********	**********	**********	**********	4.252	**********	**********	**********	**********
AXLES PER TRUCK HEAVY/COAL	.000	**********	**********	**********	**********	**********	4.637	**********	**********	**********	**********
EAL'S PER TRUCK AXLE NORMAL	.000	**********	**********	**********	**********	**********	.244	**********	**********	**********	**********
EAL'S PER TRUCK AXLE HEAVY/COAL	.000	**********	**********	**********	**********	**********	1.870	**********	**********	**********	**********

AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/
RURAL MINOR ARTERIAL
AVERAGE VALUES (SMOOTHED)

YEAR	ANNUAL CHANGE (%)	13	11	07	05	04	01	00	99	98	97
UNCLASSIFIED ROADS (ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)											
NO. OF CLASSIFICATION STA.		368	546	236	246	238	135	30	46	70	57
AADT	? -2.585 ?	6571	6741	6911	7081	7251	7421	7590	7760	7930	8100
PERCENT TRUCKS	.946	14.362	14.226	14.090	13.954	13.818	13.682	13.547	13.411	13.275	13.139
AXLES PER TRUCK	? -.163 ?	3.526	3.532	3.537	3.543	3.549	3.555	3.560	3.566	3.572	3.578
EAL'S PER TRUCK AXLE	.927	.291	.288	.286	.283	.280	.278	.275	.272	.269	.267
CLASSIFIED ROADS (MANUAL LOCATION WITH 3% OR MORE OF TRUCKS CLASSIFIED AS HEAVY/COAL)											
NO. OF CLASSIFICATION STA.		0	0	0	0	1	5	10	13	25	25
AADT	1.938	**********		12194	11958	11721	11485	11249	11012	10776	10540
PERCENT TRUCKS	?-35.939 ?	**********		5.447	7.405	9.363	11.320	13.278	15.235	17.193	19.151
PERCENT OF TRUCKS CLASSIFIED AS HEAVY/COAL	? -9.860 ?	**********		21.591	23.720	25.849	27.978	30.107	32.235	34.364	36.493
AXLES PER TRUCK NORMAL	? -1.014 ?	**********		3.317	3.350	3.384	3.417	3.451	3.485	3.518	3.552
AXLES PER TRUCK HEAVY/COAL	? -.132 ?	**********		5.077	5.083	5.090	5.097	5.103	5.110	5.117	5.124
EAL'S PER TRUCK AXLE NORMAL	? -1.301 ?	**********		.268	.271	.275	.278	.282	.285	.289	.292
EAL'S PER TRUCK AXLE HEAVY/COAL	1.481	**********		3.448	3.397	3.346	3.295	3.244	3.193	3.142	3.091

AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/
RURAL MINOR COLLECTOR/RURAL LOCAL
AVERAGE VALUES (SMOOTHED)

YEAR	ANNUAL CHANGE (%)	13	11	07	05	04	01	00	99	98	97
UNCLASSIFIED ROADS (ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)											
NO. OF CLASSIFICATION STA.		636	944	351	365	430	189	37	69	98	54
AADT	?-11.504 ?	1971	2197	2424	2651	2877	3104	3331	3558	3784	4011
PERCENT TRUCKS	1.117	10.393	10.277	10.161	10.044	9.928	9.812	9.696	9.580	9.464	9.348
AXLES PER TRUCK	.255	3.094	3.086	3.078	3.070	3.062	3.054	3.046	3.038	3.030	3.023
EAL'S PER TRUCK AXLE	? 7.050 ?	.518	.481	.445	.408	.372	.335	.299	.262	.226	.189
CLASSIFIED ROADS (MANUAL LOCATION WITH 3% OR MORE OF TRUCKS CLASSIFIED AS HEAVY/COAL)											
NO. OF CLASSIFICATION STA.		0	0	0	0	0	5	5	11	14	17
AADT	?-43.350 ?	**********		1556	2231	2905	3580	4255	4930	5604	
PERCENT TRUCKS	?-20.482 ?	**********		5.465	6.584	7.703	8.823	9.942	11.061	12.181	
PERCENT OF TRUCKS CLASSIFIED AS HEAVY/COAL	? -2.975 ?	**********		31.577	32.516	33.455	34.395	35.334	36.274	37.213	
AXLES PER TRUCK NORMAL	1.957	**********		3.525	3.456	3.387	3.318	3.249	3.180	3.111	
AXLES PER TRUCK HEAVY/COAL	? -2.990 ?	**********		3.993	4.113	4.232	4.351	4.471	4.590	4.710	
EAL'S PER TRUCK AXLE NORMAL	?-16.855 ?	**********		.124	.145	.166	.187	.208	.229	.249	
EAL'S PER TRUCK AXLE HEAVY/COAL	? 7.900 ?	**********		3.755	3.458	3.161	2.865	2.568	2.272	1.975	

AGGREGATE CLASS IV -- URBAN INTERSTATE
AVERAGE VALUES (SMOOTHED)

YEAR	ANNUAL CHANGE (%)	13	11	07	05	04	01	00	99	98	97
UNCLASSIFIED ROADS (ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)											
NO. OF CLASSIFICATION STA.		71	82	35	33	42	37	1	2	4	17
AADT	? -.323 ?	43055	43194	43333	43471	43610	43749	43888	44027	44166	44305
PERCENT TRUCKS	? -6.613 ?	15.751	16.793	17.835	18.877	19.918	20.960	22.002	23.043	24.085	25.127
AXLES PER TRUCK	? -.019 ?	4.381	4.382	4.382	4.383	4.384	4.385	4.386	4.387	4.388	4.388
EAL'S PER TRUCK AXLE	3.359	.310	.300	.289	.279	.269	.258	.248	.237	.227	.216
CLASSIFIED ROADS (MANUAL LOCATION WITH 3% OR MORE OF TRUCKS CLASSIFIED AS HEAVY/COAL)											
NO. OF CLASSIFICATION STA.		0	0	1	0	0	0	0	0	0	0
AADT	.000	**********		97564**********							
PERCENT TRUCKS	.000	**********		7.655**********							
PERCENT OF TRUCKS CLASSIFIED AS HEAVY/COAL	.000	**********		4.329**********							
AXLES PER TRUCK NORMAL	.000	**********		4.072**********							
AXLES PER TRUCK HEAVY/COAL	.000	**********		4.808**********							
EAL'S PER TRUCK AXLE NORMAL	.000	**********		.257**********							
EAL'S PER TRUCK AXLE HEAVY/COAL	.000	**********		1.726**********							

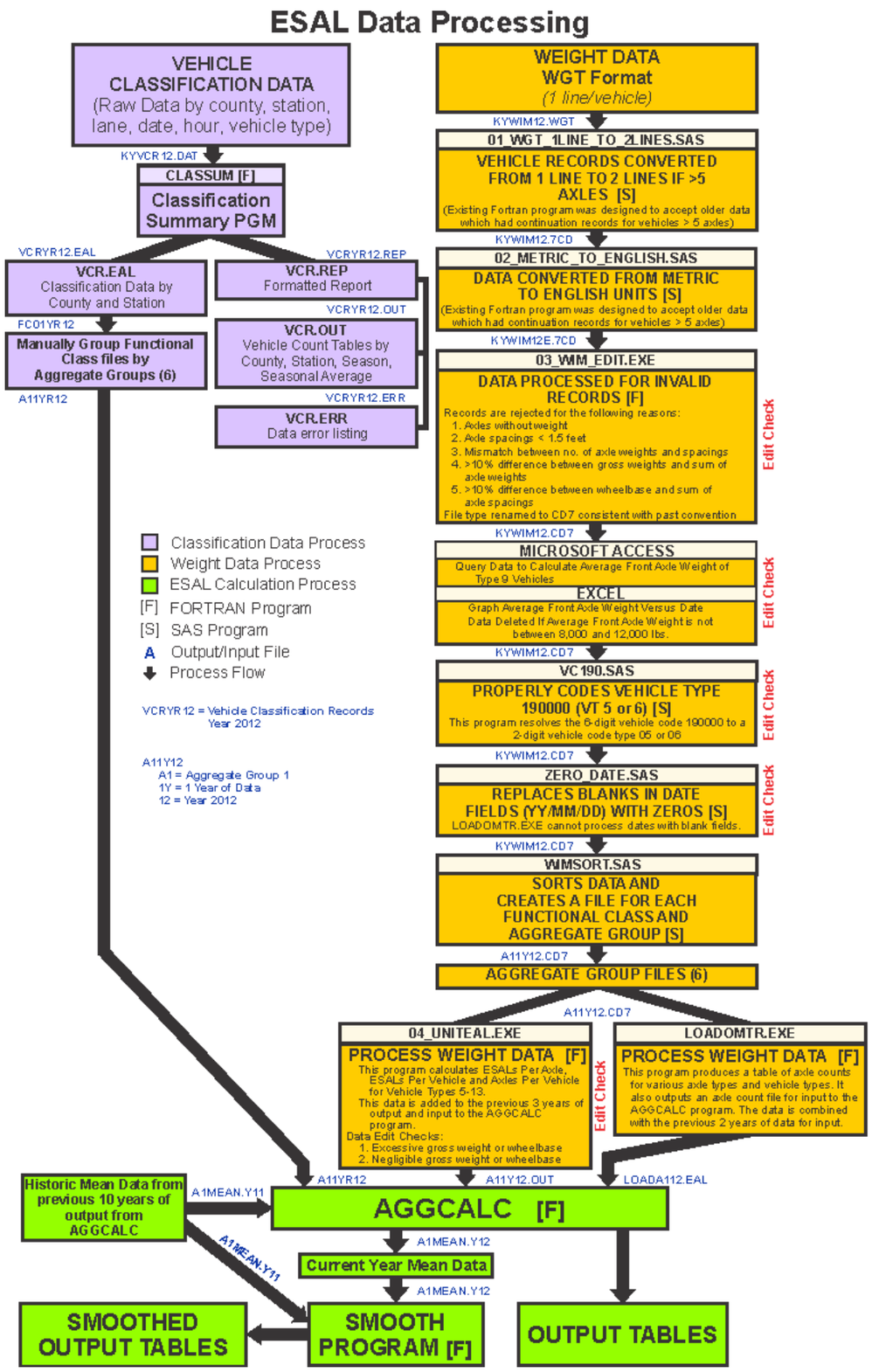
AGGREGATE CLASS V -- URBAN OTHER FREEWAY AND EXPRESSWAYS/
URBAN OTHER PRINCIPAL ARTERIAL
AVERAGE VALUES (SMOOTHED)

YEAR	ANNUAL CHANGE (%)	13	11	07	05	04	01	00	99	98	97
UNCLASSIFIED ROADS (ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)											
NO. OF CLASSIFICATION STA.		133	291	104	100	158	50	22	27	41	44
AADT	? -3.664 ?	17879	18534	19189	19844	20499	21154	21809	22464	23119	23774
PERCENT TRUCKS	2.170	9.999	9.782	9.565	9.348	9.131	8.914	8.697	8.480	8.263	8.046
AXLES PER TRUCK	1.968	3.812	3.737	3.662	3.587	3.512	3.437	3.362	3.287	3.212	3.137
EAL'S PER TRUCK AXLE	3.068	.323	.313	.303	.293	.283	.273	.263	.253	.243	.234
CLASSIFIED ROADS (MANUAL LOCATION WITH 3% OR MORE OF TRUCKS CLASSIFIED AS HEAVY/COAL)											
NO. OF CLASSIFICATION STA.		1	1	1	0	2	1	4	6	7	10
AADT	?-24.529 ?	8980	11183	13386	15589	17791	19994	22197	24400	26603	28806
PERCENT TRUCKS	4.666	14.497	13.821	13.145	12.468	11.792	11.115	10.439	9.763	9.086	8.410
PERCENT OF TRUCKS CLASSIFIED AS HEAVY/COAL	?-14.462 ?	5.440	6.227	7.014	7.801	8.587	9.374	10.161	10.948	11.734	12.521
AXLES PER TRUCK NORMAL	3.424	4.895	4.727	4.560	4.392	4.225	4.057	3.890	3.722	3.554	3.387
AXLES PER TRUCK HEAVY/COAL	1.171	4.538	4.485	4.432	4.379	4.326	4.273	4.220	4.167	4.113	4.060
EAL'S PER TRUCK AXLE NORMAL	? 6.458 ?	.397	.371	.346	.320	.294	.269	.243	.218	.192	.166
EAL'S PER TRUCK AXLE HEAVY/COAL	2.091	3.177	3.111	3.044	2.978	2.912	2.845	2.779	2.712	2.646	2.579

AGGREGATE CLASS VI -- URBAN MINOR ARTERIAL/
URBAN COLLECTOR/URBAN LOCAL
AVERAGE VALUES (SMOOTHED)

YEAR	ANNUAL CHANGE (%)	13	11	07	05	04	01	00	99	98	97
UNCLASSIFIED ROADS (ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)											
NO. OF CLASSIFICATION STA.		314	489	177	159	222	32	17	17	32	36
AADT	?-10.273 ?	6300	6947	7595	8242	8889	9536	10184	10831	11478	12125
PERCENT TRUCKS	.503	8.225	8.183	8.142	8.101	8.059	8.018	7.977	7.935	7.894	7.853
AXLES PER TRUCK	.683	3.116	3.094	3.073	3.052	3.031	3.009	2.988	2.967	2.945	2.924
EAL'S PER TRUCK AXLE	? 8.364 ?	.990	.907	.824	.741	.659	.576	.493	.410	.327	.245
CLASSIFIED ROADS (MANUAL LOCATION WITH 3% OR MORE OF TRUCKS CLASSIFIED AS HEAVY/COAL)											
NO. OF CLASSIFICATION STA.		0	3	1	0	0	0	0	0	2	1
AADT	? -5.815 ?	12796	13540	14284	15028	15772	16517	17261	18005	18749	19493
PERCENT TRUCKS	? 7.970 ?	26.901	24.757	22.613	20.469	18.325	16.182	14.038	11.894	9.750	7.606
PERCENT OF TRUCKS CLASSIFIED AS HEAVY/COAL	?-59.580 ?	2.366	3.775	5.185	6.595	8.004	9.414	10.823	12.233	13.642	15.052
AXLES PER TRUCK NORMAL	2.316	4.107	4.011	3.916	3.821	3.726	3.631	3.536	3.441	3.346	3.251
AXLES PER TRUCK HEAVY/COAL	.023	5.005	5.004	5.002	5.001	5.000	4.999	4.998	4.997	4.996	4.994
EAL'S PER TRUCK AXLE NORMAL	? 10.170 ?	.993	.892	.791	.690	.589	.488	.387	.286	.185	.084
EAL'S PER TRUCK AXLE HEAVY/COAL	? 5.439 ?	2.487	2.352	2.217	2.081	1.946	1.811	1.675	1.540	1.405	1.270

APPENDIX A


ESAL Data Processing
VEHICLE CLASSIFICATION DATA
(Raw Data by county, station, lane, date, hour, vehicle type)
KYVCR12.DAT
CLASSUM [F]
Classification Summary PGM
VCRYR12.EAL
VCR.EAL
Classification Data by County and Station
VCRYR12.REP
VCR.REP
Formatted Report
VCRYR12.OUT
VCR.OUT
Vehicle Count Tables by County, Station, Season, Seasonal Average
VCRYR12.ERR
VCR.ERR
Data error listing
FC01YR12
Manually Group Functional Class files by Aggregate Groups (6)
A11YR12
WEIGHT DATA
WGT Format
(1 line/vehicle)
KYWIM12.WGT
01_WGT_1LINE_TO_2LINES.SAS
VEHICLE RECORDS CONVERTED FROM 1 LINE TO 2 LINES IF >5 AXLES [S]
(Existing Fortran program was designed to accept older data which had continuation records for vehicles > 5 axles)
KYWIM12.7CD
02_METRIC_TO_ENGLISH.SAS
DATA CONVERTED FROM METRIC TO ENGLISH UNITS [S]
(Existing Fortran program was designed to accept older data which had continuation records for vehicles > 5 axles)
KYWIM12E.7CD
03_WIM_EDIT.EXE
DATA PROCESSED FOR INVALID RECORDS [F]
Records are rejected for the following reasons:
1. Axles without weight
2. Axle spacings < 1.5 feet
3. Mismatch between no. of axle weights and spacings
4. >10% difference between gross weights and sum of axle weights
5. >10% difference between wheelbase and sum of axle spacings
File type renamed to CD7 consistent with past convention
Edit Check
KYWIM12.CD7
MICROSOFT ACCESS
Query Data to Calculate Average Front Axle Weight of Type 9 Vehicles
EXCEL
Graph Average Front Axle Weight Versus Date
Data Deleted If Average Front Axle Weight is not between 8,000 and 12,000 lbs.
Edit Check
KYWIM12.CD7
VC190.SAS
PROPERLY CODES VEHICLE TYPE 190000 (VT 5 or 6) [S]
This program resolves the 6-digit vehicle code 190000 to a 2-digit vehicle code type 05 or 06
Edit Check
KYWIM12.CD7
ZERO_DATE.SAS
REPLACES BLANKS IN DATE FIELDS (YY/MM/DD) WITH ZEROS [S]
LOADOMTR.EXE cannot process dates with blank fields.
Edit Check
KYWIM12.CD7
WIMSORT.SAS
SORTS DATA AND CREATES A FILE FOR EACH FUNCTIONAL CLASS AND AGGREGATE GROUP [S]
A11Y12.CD7
AGGREGATE GROUP FILES (6)
A11Y12.CD7
04_UNITEAL.EXE
PROCESS WEIGHT DATA [F]
This program calculates ESALs Per Axle, ESALs Per Vehicle and Axles Per Vehicle for Vehicle Types 5-13.
This data is added to the previous 3 years of output and input to the AGGCALC program.
Data Edit Checks:
1. Excessive gross weight or wheelbase
2. Negligible gross weight or wheelbase
Edit Check
LOADOMTR.EXE
PROCESS WEIGHT DATA [F]
This program produces a table of axle counts for various axle types and vehicle types. It also outputs an axle count file for input to the AGGCALC program. The data is combined with the previous 2 years of data for input.
Classification Data Process
Weight Data Process
ESAL Calculation Process
[F] FORTRAN Program
[S] SAS Program
A Output/Input File
Process Flow
VCRYR12 = Vehicle Classification Records Year 2012
A11Y12
A1 = Aggregate Group 1
1Y = 1 Year of Data
12 = Year 2012
A11YR12
A11Y12.OUT
LOADA112.EAL
Historic Mean Data from previous 10 years of output from AGGCALC
A1MEAN.Y11
AGGCALC [F]
A1MEAN.Y12
Current Year Mean Data
A1MEAN.Y12
A1MEAN.Y11
SMOOTHED OUTPUT TABLES
SMOOTH PROGRAM [F]
OUTPUT TABLES

APPENDIX B

ESAL PROCESSING PROGRAM CODE

B.1 VEHICLE CLASSIFICATION DATA PROCESSING

CLASSIFICATION SUMMARY PROGRAM

```
**************************************************************************************
C  *  PROGRAM NAME : CLASSIFICATION SUMMARY                                          *
C  *                                                                                 *
C  *  PROGRAM FUNCTION: THIS PROGRAM USES VEHICLE CLASSIFICATION                     *
C  *      DATA TO COMPUTE AN ANNUAL AVERAGE DAILY TRAFFIC FLOW                       *
C  *      ESTIMATE FOR EACH TRAFFIC COUNT  LOCATION                                  *
C  *  THE OUTPUT CONSISTS OF HOURLY ESTIMATIONS FOR EVERY                            *
C          SEASON AND LOCATION                                                      *
C  *   EDIT CHECKS ARE PERFOMED                                                      *
C  ***********************************************************************************
C   THE CLASSUM PROGRAM CONSISTS OF SIX SUBPROGRAMS:                                *
C          VCREDIT, PREP, VCR1, VCR2, VCR3, AND VCR4, COORDINATED                   *
C          BY A BATCH FILE NAMED CLASSUM.BAT                                        *
**************************************************************************************
```

CLASSUM BAT

```
VCREDIT
PREP
VCR1
VCR2
VCR3
VCR4
```

VCREDIT.FOR

```
IMPLICIT INTEGER (A-Z)
      INTEGER LINE(77),DAT(4,24,8),VC(4,24,8,16),VH(16)
      CHARACTER*14 IINF,FILENM
      DATA DAT,VC/768* 0,12288* 0/
      DATA N0, N1, N2, N3, N4, N5, N6, N7, N8, N9/
     *     '0','1','2','3','4','5','6','7','8','9'/
      DATA  A,  B,  C,  D,  E,  F,  G,  H,  I,  J,  K,  L,  M /
     *     'A','B','C','D','E','F','G','H','I','J','K','L','M'/
      DATA  N,  O,  P,  Q,  R,  S,  T,  U,  V,  W,  X,  Y,  Z /
     *     'N','O','P','Q','R','S','T','U','V','W','X','Y','Z'/
C     CALL REREAD
      OPEN(111,FILE='FILE.TMP',STATUS='OLD', MODE='READ')
      READ(111, 89)FILENM
   89  FORMAT(A12)
C     WRITE(*,'(A)')' THE INPUT DATA FILE NAME ='
C     READ(*,'(A)')IINF
      IIN=15
      OPEN(IIN,FILE=FILENM, STATUS='OLD', MODE='READ')
      OPEN(12,FILE='FILE12')
  100 READ(15,1500,END=9999) LINE,CRDNO
      IF (LINE(7).EQ.N0.OR.LINE(7).EQ.N1.OR.LINE(7).EQ.N2.OR.
     *    LINE(7).EQ.N3.OR.LINE(7).EQ.N4.OR.LINE(7).EQ.N5.OR.
     *    LINE(7).EQ.N6.OR.LINE(7).EQ.N7.OR.LINE(7).EQ.N8.OR.
     *    LINE(7).EQ.N9) GOTO 900
      IF (LINE(7).EQ.A.OR.LINE(7).EQ.B.OR.LINE(7).EQ.C.OR.
     *    LINE(7).EQ.D.OR.LINE(7).EQ.E.OR.LINE(7).EQ.F) GOTO 200
      IF (LINE(7).EQ.G.OR.LINE(7).EQ.H.OR.LINE(7).EQ.J.OR.
     *    LINE(7).EQ.K.OR.LINE(7).EQ.L.OR.LINE(7).EQ.M) GOTO 220
      IF (LINE(7).EQ.N.OR.LINE(7).EQ.P.OR.LINE(7).EQ.Q.OR.
     *    LINE(7).EQ.R.OR.LINE(7).EQ.S.OR.LINE(7).EQ.T) GOTO 240
      IF (LINE(7).EQ.U.OR.LINE(7).EQ.V.OR.LINE(7).EQ.W.OR.
     *    LINE(7).EQ.X.OR.LINE(7).EQ.Y.OR.LINE(7).EQ.Z) GOTO 260
  200 LINE(7) = N1
      GOTO 300
  220 LINE(7) = N5
      GOTO 300
  240 LINE(7) = N3
      GOTO 300
  260 LINE(7) = N7
  300 IF (CRDNO.GE.98) GOTO 800
      OPEN(10,FILE='FILE10')
      WRITE(10,1500) LINE,CRDNO
      BACKSPACE 10
      READ(10,1510) JJ,DTE,(VH(MM),MM=1,16)
 1510 FORMAT(6X,I1,2X,I6,I3,1X,I5,1X,I4,1X,I3,1X,I2,1X,I4,2I3,1X,
     *       I3,I4,I3,1X,3I3,1X,I3,I5)
```

```
      KK = 1
      IF (DTE.GE. 30000) KK = 2
      IF (DTE.GE. 60000) KK = 3
      IF (DTE.GE. 90000) KK = 4
      IF (DTE.GE.120000) KK = 1
      II = MOD(DTE,100)
      IF (II.EQ.0) II = 24
      DO 20 IN=1,16
      VC(KK,II,JJ,IN)=VH(IN)+VC(KK,II,JJ,IN)
   20 CONTINUE
      PRNT=1
      DAT(KK,II,JJ)=DTE
      GOTO 100
  800 IF (PRNT.EQ.0) GOTO 900
      PRNT = 0
      LCTR = 0
      DO 850 KK = 1, 4
        DO 850 II = 1, 24
        IF (DAT(KK,II,1).NE.0.OR.DAT(KK,II,3).NE.0.OR.
     *       DAT(KK,II,5).NE.0.OR.DAT(KK,II,7).NE.0) LCTR = LCTR + 1
          DO 850 JJ = 1,7,2
            IF (DAT(KK,II,JJ).EQ.0) GOTO 850
              WRITE(12,1520) (LINE(LL),LL=1,6),JJ,LINE(8),LINE(9),
     *           DAT(KK,II,JJ),(VC(KK,II,JJ,LL),LL=1,16),LCTR
  1520 FORMAT(6A1,I1,2A1,I6,I3,1X,I5,1X,I4,1X,I3,1X,I2,1X,I4,2I3,1X,
     *     I3,I4,I3,1X,3I3,1X,I3,I5,I2,1X)
            DAT(KK,II,JJ) = 0
          DO 840 IN=1,16
              VC(KK,II,JJ,IN)=0
   840 CONTINUE
   850 CONTINUE
   900 WRITE(12,1500) LINE,CRDNO
      GOTO 100
  1500 FORMAT(77A1,I2,1X)
 C9999 RETURN
  9999 CLOSE(10,STATUS='DELETE')
      STOP
      END
```

PREP.FOR

```
$DEBUG
      COMMON/FILES/IIN,IOUT,FILE099,FILE12
      REAL*8 RTA,RTB,RTC,RT
      REAL*8 AMP,XL,BMP,CMP,AX,ESL
      INTEGER IW,IFC,IADT,ICO,JCO
      OPEN(10,FILE='FILE10', STATUS='OLD', MODE='READ')
      OPEN(100,FILE='FILE100')
  20   READ (10,1000,END=90) ICO,RTA,RTB,AMP,BMP,XL,IW
      WRITE(100,1050)ICO,RTA,RTB,AMP,BMP,XL,IW
      GOTO 20
      CLOSE(100)
C*****************************************************************
C  *  FORMAT STATEMENTS.                                         *
C  *****************************************************************
1000   FORMAT (I3,1X,A2,A6,A9,2A8,21X,I10)
1050   FORMAT(I3,A2,A6,3A7,16X,I9)
  90   STOP
      END
```

VCR1.FOR

```
$DEBUG
C                          CLASSIFICATION SUMMARY
C  ***************************************************************
C ********************    SUB PROGRAM : VCR1   ********************
C ***************************************************************
      COMMON/FILES/IIN,IOUT,FILE099,FILE12
      REAL*8 ROUTE(500),TROUT
      REAL*8 T1,T3,T4,T5,T6,T7,T8,T9,T10
      REAL*4 DENLOG
      INTEGER T2,CN(500),FED(500),DIRS(4,24),OWTW(500)
      INTEGER FC(500)
      REAL*8 SN(500)
      REAL M(4,24,15),MP(500),MULT
      INTEGER S,D,DP1,DM1,N,T,F,DIR,VC(14),OVCA(23),OVCB(15)
      REAL HF(2,4,24,24,14),SF(500,4,15),SE(4)
      INTEGER RM(500),R,HR,YR(500),NL,AADT(500),DOR(4,24),MOR(4,24)
      INTEGER CHRU,CHRI,YF83,YF88,HRS(4),NUMHRS(500,4)
      INTEGER BLANK,MONTH,DAY,HRCT,MONCT
      INTEGER DISC(76),ERRCT,PRNT(15),COLHAL,OLDATE
      INTEGER ICNT,IOLD,ISTATE,LUMP(4,24)
      CHARACTER*14 IOUTF,IINF
      CHARACTER*11 FN1,FN2
      CHARACTER*2 YR1
C
C  ***************************************************************
C  *   FOLLOWING STATEMENTS   INITIALIZE  THE VARIABLES  FOR THIS   *
C  *   PROGRAM.                                    *
C  ***************************************************************
      DATA M/1440* 0./
      DATA SF/30000* 0./
      DATA SE/4* 1./
      DATA HRS/4* 0/
      DATA FED/500* 0/
      DATA OWTW/500* 1/
      DATA DOR/96* 0/
      DATA MOR/96* 0/
      DATA DIRS/96* 0/
      DATA LUMP/96* 0/
      DATA CHRI/'I'/
      DATA CHRU/'U'/
      DATA BLANK/' '/
      DATA BLANK3/'   '/
      NL = 0
      HRCT = 0
      MONCT = 0
      ERRCT = 0
```

```
      YF83 = 0
      YF88 = 0
      F = 0
      OLDATE = 0
      ISTATE = 0
      ICNT   = 0
      IOLD   = 0
      OPEN(16, FILE = 'FILE016', MODE ='WRITE')
      CLOSE(16,STATUS='DELETE')
      DO 5 R = 1,2
        DO 5 S = 1,4
          DO 5 N = 1,24
            DO 5 D = 1,24
              DO 5 T = 1,14
  5             HF(R,S,N,D,T) = 0.
C
C  ****************************************************************
C *  READS THE DATA FROM THE  VEHICLE CLASSIFICATION DISK *
C  ****************************************************************
C
 111  FORMAT('')
     DO 6 N=1,40
       WRITE(*,111)
  6  CONTINUE

      WRITE(*,*)"                ****** VCR1 is Running ******"
     DO 7 N=1,10
       WRITE(*,111)
  7  CONTINUE


      IOUTF=FILE099
      IINF=FILE12
      IOUT = 99
      OPEN(99, ACCESS='SEQUENTIAL', STATUS = 'SCRATCH',
    * FORM='FORMATTED')
      OPEN(12,FILE='FILE12', STATUS='OLD', MODE='READ')
      READ(12,4999)YR1
 4999  FORMAT(T8,A2)
      FN1='VCRYR'//YR1//'.REP'
      FN2='VCRYR'//YR1//'.ERR'
      OPEN(8, FILE = FN1, MODE ='WRITE')
      OPEN(16, FILE = 'FILE016', MODE ='WRITE')
      OPEN(7, FILE = FN2, MODE ='WRITE')
      WRITE (8,5400)
      REWIND 12
      IIN = 12
      GOTO 20
```

```
 10   BACKSPACE 12
 20   READ (12,5000,END=90) T1,T2,TROUT,T3,T4,T5,
     *                  T6,T7,T8,T9,T10,ICSN
      IF (T2.GT.83) YF83 = 1
      IF (T2.GE.89) YF88 = 1
C  ****************************************************************
C  THE FOLLOWING ASSIGNMENT STATEMENT IS INCLUDED DUE TO THE FACT THAT
C  THE VEHICLE CLASSIFICATION RECORDS RECEIVED FROM FRANKFORT IS STILL
C  IN 14 VEHICLE FORMAT.  YF88(YEAR FOLLOWING 1988) WAS DESIGNED TO
C  CAUSE THE PROGRAM TO READ THE DATA DIFFERENTLY WHEN THE DATA WENT
C  TO 13 VEHICLE TYPES.
C  ****************************************************************
C
      YF88 = 0
      IF (ICSN-98) 25, 32, 35
 25   IF (F.NE.2) GOTO 26
      F = 1
      GOTO 59
 26   WRITE(IOUT,5000) T1,T2,TROUT,T3,T4,T5,T6,T7,
     *                  T8,T9,T10,ICSN
        IF (YF88.EQ.1) GOTO 27
        IF (YF83.EQ.1.AND.YF88.EQ.0) GOTO 24
C
C  ****************************************************************
C  *   READ THE DATA  USING THE  PROPER  FORMAT, AND  IF  NECESSARY *
C  *   CONVERT THE VEHICLE CODES 1 - 22 (OR 1 - 14)  TO THE VEHICLE *
C  *   CODES 1-13                                                   *
C  ****************************************************************
        OPEN(99,FILE=IOUTF)
        BACKSPACE 99
        READ (IOUT,2000) DIR,MONTH,DAY,HR,(OVCA(I),I=1,23)
        VC( 1)=OVCA(22)
        VC( 2)=OVCA( 1) + OVCA( 2) + OVCA( 3)+ OVCA( 4)
        VC( 3)=OVCA( 5) + OVCA( 6)
        VC( 4)=OVCA(20) + OVCA(21)
        VC( 5)=OVCA( 7)
        VC( 6)=OVCA( 8)
        VC( 7)=OVCA( 9)
        VC( 8)=OVCA(10) + OVCA(11) + OVCA(18)
        VC( 9)=OVCA(12) + OVCA(19)
        VC(10)=OVCA(13) + OVCA(14) + OVCA(15)
        VC(11)=OVCA(16)
        VC(12)=OVCA(17)
        VC(13)=0
        VC(14)=OVCA(23)
        GOTO 28
 24     BACKSPACE 99
        READ (99,2005) DIR,MONTH,DAY,HR,(OVCB(I),I=1,15)
```

```
      WRITE (IOUT,2005) DIR,MONTH,DAY,HR,(OVCB(I),I=1,15)
      VC( 1)=OVCB( 1)
      VC( 2)=OVCB( 2)
      VC( 3)=OVCB( 3)
      VC( 4)=OVCB( 4) + OVCB( 5)
      VC( 5)=OVCB( 6)
      VC( 6)=OVCB( 7)
      VC( 7)=OVCB( 8)
      VC( 8)=OVCB( 9)
      VC( 9)=OVCB(10)
      VC(10)=OVCB(11)
      VC(11)=OVCB(12)
      VC(12)=OVCB(13)
      VC(13)=OVCB(14)
      VC(14)=OVCB(15)
      GOTO 28
 27    BACKSPACE 99
      READ(99,2010) DIR,MONTH,DAY,HR,(VC(I),I=1,14)
 28    IF (HR.EQ.0) HR = 24
      IF (HR.LE.0.OR.HR.GT.24) GOTO 20
      IF (MONTH.LT.1.OR.MONTH.GT.12) GOTO 20
      DO 29 T = 1,14
        IF (VC(T).LT.0) VC(T) = 0
 29    CONTINUE
C
C  ****************************************************************
C  *   DETERMINE  THE  SEASON  OF   THE  COUNT.                   *
C  *   COMPUTE  THE NUMBER  OF  TRUCKS  COUNTED.                  *
C  *   REJECTS THE DATA IF THE NUMBER OF COAL TRUCKS EXCEEDS THE    *
C  *   NUMBER OF TRUCKS.                                          *
C  ****************************************************************
C
C       IF (IABS((MONTH*31+DAY)-OLDATE).GT.15) OLDATE = MONTH*31+DAY
      ISEAS = MOD(INT(FLOAT(MONTH)/3.),4) + 1
      NUMTRK = VC(6) + VC(7) + VC(8) + VC(9) + VC(10)
      IF (VC(14).GT.NUMTRK) GOTO 20
C
C  ****************************************************************
C  *   DETERMINATION OF THE NUMBER OF DIRECTIONS COUNTED, AND AVERAGE*
C  *   THE DATA IF MORE THAN ONE COUNT IS TAKEN FOR A PARTICULAR HOUR*
C  ****************************************************************
C
      DENOM = 1.
      DIRS(ISEAS,HR) = DIRS(ISEAS,HR) * 10 + DIR
      IF (DIRS(ISEAS,HR).GE.100) DENOM = .6666666
      M(ISEAS,HR,14) = 0
      DO 30 I=1,13
        M(ISEAS,HR,I)=(M(ISEAS,HR,I) + VC(I)) * DENOM
```

```
          M(ISEAS,HR,15)=M(ISEAS,HR,15)+M(ISEAS,HR,I)
 30     CONTINUE
        M(ISEAS,HR,14)=(M(ISEAS,HR,14) + VC(14)) * DENOM
        F = 1
        SE(ISEAS) = SE(ISEAS) * (.9 + .2111 * SGN(M(ISEAS,HR,14)))
        IF (DIRS(ISEAS,HR).LT.10) GOTO 10
        IF (MOD(DIRS(ISEAS,HR),11).NE.0) OWTW(NL+1) = 2
      GOTO 20
C
C ****************************************************************
C *   FOLLOWING STATEMENTS ARE  EXECUTED  WHEN  ALL  DATA  FOR A  *
C *   SPECIFIC  LOCATION HAS BEEN  READ.                          *
C ****************************************************************
C
 32   IF (F.EQ.2) GOTO 20
      IF (F.EQ.0) GOTO 59
      F = 2
      WRITE(99,5000) T1,T2,TROUT,T3,T4,T5,T6,T7,
     *                T8,T9,T10,ICSN
      BACKSPACE 99
      READ (99,5100) (DISC(I),I=1,76)
      WRITE(16,5200) (DISC(I),I=1,6),(DISC(I),I=10,76)
      GOTO 20
C
C ****************************************************************
C *   PRINTS A VERBAL  LOCATION DESCRIPTION AT THE BEGINNING OF   *
C *   THE PRINTOUT.                                               *
C *   DECIDE THE ROAD TYPE.                                       *
C ****************************************************************
C
 35   IF (F.EQ.0) GOTO 20
        IF (F.EQ.1) GOTO 59
        NL = NL + 1
        WRITE( 8,5300) (DISC(I),I=1,6),T2,TROUT,(DISC(I),I=10,76)
        WRITE(99,5000) T1,T2,TROUT
        BACKSPACE 99
        READ (99,5100) (DISC(I),I=1,17)
        R = 2
        DO 40 I = 10,14
          IF (DISC(I).EQ.BLANK) GOTO 40
          IF (DISC(I).EQ.CHRU.OR.DISC(I).EQ.CHRI) R = 1
          GOTO 45
 40     CONTINUE
C
C ****************************************************************
C *   FOLLOWING STATEMENTS  DECIDE IF THE LOCATION  IS A COAL     *
C *   ROAD OR NOT.                                                *
C ****************************************************************
```

```
C
 45     RM(NL) = R
        NUMCOL = 0
        NUMTRK = 0
        DO 50 S = 1,4
          DO 50 N = 1,24
            NUMCOL = NUMCOL + M(S,N,14)
            DO 50 T = 4,13
              NUMTRK = NUMTRK + M(S,N,T)
 50     CONTINUE
        NUMCOL = NUMCOL * 100
        COLHAL = 0
        IF (NUMCOL.GE.NUMTRK) COLHAL = 1
C
C  ****************************************************************
C  *   COMPUTE HOURLY CONVERSION FACTORS FOR EXISTING DATA        *
C  ****************************************************************
C
        DO 58 S = 1,4
          IF (SE(S).LE.1.) SF(NL,S,15) = -10
          IF (SE(S).LE.1.) GOTO 58
          DO 57 D = 1,23
            IF (M(S,D,15).EQ.0.) GOTO 57
            DP1 = D + 1
            DO 55 N=DP1,24
              IF (M(S,N,15).EQ.0.) GOTO 55
                DO 52 T=1,13
                  DENOM = M(S,N,T) + M(S,D,T) + 2
                  DENLOG = ALOG10(DENOM)
                  HF(R,S,N,D,T) = (HF(R,S,N,D,T)*HF(R,S,D,N,T)+
     *              (((M(S,N,T)-M(S,D,T))*DENLOG)/DENOM))/
     *               (HF(R,S,D,N,T)+DENLOG)
                  HF(R,S,D,N,T) = HF(R,S,D,N,T) + DENLOG
 52               CONTINUE
                IF (COLHAL.EQ.0) GOTO 55
                  DENOM = M(S,N,14) + M(S,D,14) + 2
                  DENLOG = ALOG10(DENOM)
                  HF(R,S,N,D,14)=(HF(R,S,N,D,14)*HF(R,S,D,N,14)+
     *              (((M(S,N,14)-M(S,D,14))*DENLOG)/DENOM))/
     *               (HF(R,S,D,N,14)+DENLOG)
                  HF(R,S,D,N,14) = HF(R,S,D,N,14) + DENLOG
 55           CONTINUE
 57         CONTINUE
 58   CONTINUE
C
C  ****************************************************************
C  *   STORE AND RESET THE DATA SET.                              *
C  ****************************************************************
```

```
C
 59   DO 70 S = 1,4
          DO 60 D = 1,24
            DIRS(S,D) = 0
            DO 60 T = 1,15
              M(S,D,T) = 0.
 60         CONTINUE
 70   SE(S) = 1
      OLDATE = 0
      F = 0
      GOTO 20
C
C  ****************************************************************
C  *   CALCULATION OF RATIOS FOR HOURS IN WHICH DIRECT RELATIONSHIP *
C  *   ARE NOT AVAILABLE (I.E. IF HOUR 1 TO 3 IS NOT KNOWN  BUT  1  *
C  *   TO 2 AND 2 TO 3 ARE,  1 TO 3 MAY BE CALCULATED              *
C  ****************************************************************
C
 90   NL = 0
      DO 98 R = 1,2
        DO 98 S = 1,4
          DO 98 D = 1,23
            DP1 = D + 1
            DO 98 N = DP1,24
              DO 98 T = 1,14
                IF (HF(R,S,D,N,T).NE.0.) GOTO 98
                DENOM = SGN(HF(R,1,D,N,T)) + SGN(HF(R,2,D,N,T))
     *                + SGN(HF(R,3,D,N,T)) + SGN(HF(R,4,D,N,T))
                IF (DENOM.EQ.0.) GOTO 95
                HF(R,S,N,D,T)=(HF(R,1,N,D,T) + HF(R,2,N,D,T)
     *                 + HF(R,3,N,D,T) + HF(R,4,N,D,T))/DENOM
                HF(R,S,D,N,T)=(HF(R,1,D,N,T) + HF(R,2,D,N,T)
     *                 + HF(R,3,D,N,T) + HF(R,4,D,N,T))/DENOM
                GOTO 98
 95               IR = 3 - R
                HF(R,S,N,D,T) = HF(IR,S,N,D,T)
                HF(R,S,D,N,T) = HF(IR,S,D,N,T)
 98   CONTINUE
100   CONTINUE
      WRITE(8,5500)
C  ****************************************************************
C  *   FOLLOWING STATEMENTS READ THE  DATA OFF THE  DISK  A  SECOND *
C  *   TIME, DUE  TO THE FACT THAT THERE IS  TOO MUCH RAW DATA TO   *
C  *   BE STORED.                                                   *
C  ****************************************************************
      REWIND 12
110   READ (12,5000,END=190) T1,T2,TROUT,T3,T4,T5,
     *                 T6,T7,T8,T9,T10,ICSN
```

```
      IF (ICSN-98) 120,134,135
  120 IF (F.NE.2) GOTO 123
      F = 1
      GOTO 163
  123 WRITE(99,5000) T1,T2,TROUT,T3,T4,T5,T6,T7,
     *                T8,T9,T10,ICSN
      IF (YF88.EQ.1) GOTO 125
      IF (YF83.EQ.1.AND.YF88.EQ.0) GOTO 124
C
C ****************************************************************
C *  FOLLOWING READ THE DATA USING  THE  PROPER  FORMAT, AND IF  *
C *  NECESSARY CONVERT THE VEHICLE CODES 1 - 22 TO CODES 1 - 14  *
C ****************************************************************
      BACKSPACE 99
      READ (99,2000) DIR,MONTH,DAY,HR,(OVCA(I),I=1,23)
      VC( 1)=OVCA(22)
      VC( 2)=OVCA( 1) + OVCA( 2) + OVCA( 3)+ OVCA( 4)
      VC( 3)=OVCA( 5) + OVCA( 6)
      VC( 4)=OVCA(20) + OVCA(21)
      VC( 5)=OVCA( 7)
      VC( 6)=OVCA( 8)
      VC( 7)=OVCA( 9)
      VC( 8)=OVCA(10) + OVCA(11) + OVCA(18)
      VC( 9)=OVCA(12) + OVCA(19)
      VC(10)=OVCA(13) + OVCA(14) + OVCA(15)
      VC(11)=OVCA(16)
      VC(12)=OVCA(17)
      VC(13)=0
      VC(14)=OVCA(23)
      GOTO 126
  124   BACKSPACE 99
      READ (99,2005) DIR,MONTH,DAY,HR,(OVCB(I),I=1,15)
      VC( 1)=OVCB( 1)
      VC( 2)=OVCB( 2)
      VC( 3)=OVCB( 3)
      VC( 4)=OVCB( 4) + OVCB( 5)
      VC( 5)=OVCB( 6)
      VC( 6)=OVCB( 7)
      VC( 7)=OVCB( 8)
      VC( 8)=OVCB( 9)
      VC( 9)=OVCB(10)
      VC(10)=OVCB(11)
      VC(11)=OVCB(12)
      VC(12)=OVCB(13)
      VC(13)=OVCB(14)
      VC(14)=OVCB(15)
      GOTO 126
  125   BACKSPACE 99
```

```
      READ(99,2010) DIR,MONTH,DAY,HR,(VC(I),I=1,14)
 126    IF (HR.EQ.0) HR = 24
      IF (HR.GE.0.AND.HR.LE.24) GOTO 128
        HRCT = HRCT + 1
        GOTO 110
 128    IF (MONTH.GT.0.AND.MONTH.LE.12) GOTO 129
        MONCT = MONCT + 1
        GOTO 110
 129    DO 130 T = 1,14
        IF (VC(T).LT.0) VC(T) = 0
 130    CONTINUE
C
C  ***************************************************************
C  *   FOLLOWING STORE THE SEASON, MONTH, DAY, HOUR, AND DIRECTION  *
C  *   TRAFFIC FLOW FOR THE LOCATION.                              *
C  ***************************************************************
C
      ISEAS = MOD(INT(FLOAT(MONTH)/3.),4) + 1
      DOR(ISEAS,HR) = DAY
      MOR(ISEAS,HR) = MONTH
      DENOM = 1
      DIRS(ISEAS,HR) = DIRS(ISEAS,HR) * 10 + DIR
      IF (DIRS(ISEAS,HR).GE.100) DENOM = .666666
      M(ISEAS,HR,15) = 0
      DO 131 I=1,13
        M(ISEAS,HR,I)=(M(ISEAS,HR,I) + VC(I))*DENOM
        M(ISEAS,HR,15) = M(ISEAS,HR,15) + M(ISEAS,HR,I)
 131    CONTINUE
      M(ISEAS,HR,14)=(M(ISEAS,HR,14) + VC(14))*DENOM
      F = 1
      IF (SF(NL+1,ISEAS,15).GT.-10) GOTO 110
        ICNT = 1
        LUMP(ISEAS,HR) = 1
    GOTO 110
C
C  ***************************************************************
C  *   FOLLOWING STATEMENTS ARE  EXECUTED  WHEN  ALL DATA  FOR  A  *
C  *   SPECIFIC LOCATION HAS BEEN READ.  AND CLOSE THE DATA SET    *
C  ***************************************************************
C
 134  IF (F.EQ.0) GOTO 110
    F = 2
    GOTO 110
 135  IF (F.EQ.0) GOTO 110
      IF (F.EQ.1) GOTO 163
      NL = NL + 1
      ROUTE(NL) = TROUT
      WRITE(99,5000) T1,T2,ROUTE(NL),T3,T4,T5,
```

```
     *                T6,T7,T8,T9,T10,ICSN
          BACKSPACE 99
          READ(99,3000) CN(NL),SN(NL),YR(NL),
     *          FED(NL),FC(NL),AADT(NL),MP(NL)
C
C ***************************************************************
C *  FOLLOWING STATEMENTS  USE THE HOURLY FACTORS COMPUTED  *
C *  PREVIOUSLY TO SUPPLY INFORMATION FOR THOSE HOURS FOR WHICH  *
C *  THERE IS NO COLLECTED DATA.                          *
C ***************************************************************
C
          IF (ICNT.EQ.0) GOTO 143
            DO 141 S = 1,4
              ICNT = 0
              DO 136 T = 1,14
                SF(NL,S,T) = 0.
  136         CONTINUE
              DO 138 N = 1,24
                IF (LUMP(S,N).EQ.0) GOTO 138
                ICNT = ICNT + 1
                DO 137 T = 1,14
                  SF(NL,S,T) = SF(NL,S,T) + M(S,N,T)
  137           CONTINUE
  138         CONTINUE
              IF (ICNT.EQ.0) GOTO 141
                R = 0
                SF(NL,S,15) = -1
                DO 140 N = 1,24
                  IF (LUMP(S,N).EQ.0) GOTO 140
                  LUMP(S,N) = 0
                  R = R + 1
                  M(S,N,15) = 0
                  DO 139 T = 1,13
                  M(S,N,T)=INT(FLOAT(R)*SF(NL,S,T)/FLOAT(ICNT))-
     *                 INT(FLOAT(R-1)*SF(NL,S,T)/FLOAT(ICNT))
                  M(S,N,15) = M(S,N,15) + M(S,N,T)
  139               CONTINUE
                 M(S,N,14)=INT(FLOAT(R)*SF(NL,S,14)/FLOAT(ICNT))-
     *                 INT(FLOAT(R-1)*SF(NL,S,14)/FLOAT(ICNT))
  140         CONTINUE
            ICNT = 0
  141       CONTINUE
  143       R = RM(NL)
          F = 0
          DO 158 S = 1,4
            IF (SF(NL,S,15).EQ.-10.) GOTO 158
            DO 155 D = 1,24
              IF (DIRS(S,D).GT.0) HRS(S) = HRS(S) + 1
```

```
      MULT = 1.
      IF (OWTW(NL).EQ.1.AND.DIRS(S,D).GE.10) MULT = 0.5
      IF (OWTW(NL).EQ.2.AND.DIRS(S,D).LT.10) MULT = 2.0
      DO 144 T = 1,15
        M(S,D,T) = M(S,D,T) * MULT
144     CONTINUE
      IF (M(S,D,15).NE.0.) GOTO 155
      DO 152 T=1,14
        DENOM = 0.
        DM1 = D - 1
        IF (D.EQ.1) DM1 = 1
        DO 145 N=1,DM1
          IF (N.GE.D.OR.M(S,N,15).EQ.0.) GOTO 145
          DENOM = DENOM + 1.
          IF (M(S,N,T).LT.0.5) GOTO 145
            M(S,D,T) = (M(S,N,T)+1)*(1+HF(R,S,D,N,T))/
     *          (1-HF(R,S,D,N,T)) + M(S,D,T) - 1
145       CONTINUE
        DP1 = D + 1
        IF (D.EQ.24) DP1 = 24
        DO 150 N=DP1,24
          IF (N.LE.D.OR.M(S,N,15).EQ.0.) GOTO 150
          DENOM = DENOM + 1.
          IF (M(S,N,T).LT.0.5) GOTO 150
            M(S,D,T) = (M(S,N,T)+1)*(1-HF(R,S,N,D,T))/
     *          (1+HF(R,S,N,D,T)) + M(S,D,T) - 1
150       CONTINUE
        IF(M(S,D,T).LT.0.) M(S,D,T) = 0
        IF(DENOM.NE.0.) M(S,D,T) = M(S,D,T) / DENOM
152     CONTINUE
155   CONTINUE
    IMP = MP(NL) * 1000
    SF(NL,S,15) = 0
     DO 157 D=1,24
      ID = D-1
      IF (ID.EQ.0) ID = 24
      DO 156 T=1,13
       IF (M(S,ID,15).LE.0.1) M(S,ID,15)=M(S,ID,15)-M(S,ID,T)
       PRNT(T) = INT(M(S,ID,T))
156     CONTINUE
      NUMTRK = M(S,ID,6)+M(S,ID,7)+M(S,ID,8)+M(S,ID,9)+M(S,ID,10)
      IF (M(S,ID,14).GT.FLOAT(NUMTRK)) M(S,ID,14) = NUMTRK
      PRNT(14) = INT(M(S,ID,14))
      PRNT(15) = INT(M(S,ID,15))
      IF (M(S,ID,15).LT.0.) SF(NL,S,15) = -1
      IREC = 4
      DM1 = D - 1
157     CONTINUE
```

```
 158     CONTINUE
C  ****************************************************************
C  *   FOLLOWING STATEMENTS COMPUTE AND STORE THE TOTALS FOR  THE   *
C  *   DATA SET.  RESET THE ARRAYS FOR READING THE NEXT LOCATION    *
C  ****************************************************************
C
 159  DO 160 S = 1,4
      IF (NL.EQ.0) GOTO 163
        NUMHRS(NL,S) = HRS(S)
        DO 160 T = 1,14
          SF(NL,S,T) = 0
          DO 160 D = 1,24
            SF(NL,S,T) = SF(NL,S,T) + M(S,D,T)
 160   CONTINUE
 163   DO 170 S = 1,4
       HRS(S) = 0
       SE(S) = 1
       DO 170 D = 1,24
         DOR(S,D) = 0
         MOR(S,D) = 0
         DIRS(S,D) = 0
         LUMP(S,D) = 0
         DO 170 T = 1,15
           M(S,D,T) = 0
 170   CONTINUE
     ICNT = 0
     GOTO 110
 190   NTOT = NL
     DO 200 R = 1,2
       DO 200 N = 1,4
         DO 200 D = 1,4
           DO 200 T = 1,14
             HF(R,1,N,D,T) = 0
 200   CONTINUE
C
C  ****************************************************************
C  *   FOLLOWING  STATEMENTS  USE  SEASONAL  TOTALS  TO  COMPUTE   *
C  *   RELATIONSHIPS  BETWEEN SEASONS  FOR  EACH  OF THE  VEHICLE   *
C  *   TYPES.                                                       *
C  ****************************************************************
C
      AMIN = .66667
      AMAX = 1.5
      AFMIN = (AMIN-1.) / (AMIN+1.)
      AFMAX = (AMAX-1.) / (AMAX+1.)
      BMIN = .25
      BMAX = 4.0
      BFMIN = (BMIN-1.) / (BMIN+1.)
```

```
      BFMAX = (BMAX-1.) / (BMAX+1.)
      DO 290 NL = 1,NTOT
        R = RM(NL)
        DO 290 D = 1,3
          IF (SF(NL,D,15).EQ.-10.) GOTO 290
          DP1 = D + 1
          DO 280 N = DP1,4
            IF (SF(NL,N,15).EQ.-10.) GOTO 280
            DO 270 T = 1,14
              DENOM = SF(NL,N,T) + SF(NL,D,T) + 2
              FACT = (SF(NL,N,T) - SF(NL,D,T)) / DENOM
              DENLOG = ALOG10(DENOM)
              IF (T.EQ.1.OR.T.EQ.4) GOTO 250
              IF (FACT.LT.AFMIN) FACT = AFMIN
              IF (FACT.GT.AFMAX) FACT = AFMAX
              GOTO 260
 250            IF (FACT.LT.BFMIN) FACT = BFMIN
              IF (FACT.GT.BFMAX) FACT = BFMAX
 260            HF(R,1,N,D,T) = (HF(R,1,N,D,T)*HF(R,1,D,N,T)+
     *           FACT*DENLOG)/(HF(R,1,D,N,T)+DENLOG)
              HF(R,1,D,N,T) = HF(R,1,D,N,T) + DENLOG
 270          CONTINUE
 280        CONTINUE
 290    CONTINUE
C
      OPEN(100,FILE='FILE100')
      OPEN(101,FILE='FILE101')
      OPEN(102,FILE='FILE102')
      OPEN(103,FILE='FILE103')
      OPEN(104,FILE='FILE104')
      DO 300 R=1,2
        DO 300 S=1,4
          DO 300 N=1,24
            DO 300 D=1,24
              DO 300 T=1,14
                WRITE(101,*)HF(R,S,N,D,T)
 300    CONTINUE
      WRITE(102,10000)NTOT
10000  FORMAT(I3)
10001  FORMAT(A8,2I4)
10002  FORMAT(A6,2X,A11,1X,A11)
      DO 310 NL=1,NTOT
        WRITE(102,10001)ROUTE(NL),RM(NL)
        WRITE(100,*)CN(NL),SN(NL),YR(NL),FED(NL),FC(NL),
     *          AADT(NL),MP(NL)
 310    CONTINUE
        DO 315 NL=1,NTOT
         DO 315 S=1,4
```

```
          DO 315 T=1,15
            WRITE(103,*)SF(NL,S,T)
315       CONTINUE
      WRITE(104,10002)'FILE12',FN1,FN2
      WRITE(104,10001)TROUT,OLDATE,F
      CLOSE(100,STATUS='KEEP')
      CLOSE(101)
      CLOSE(102)
      CLOSE(103)
      CLOSE(104)
      OPEN(200,FILE='FILE200')
      OPEN(201,FILE='FILE201')
      OPEN(202,FILE='FILE202')
      OPEN(203,FILE='FILE203')
       OPEN(204,FILE='FILE204')
       OPEN(205,FILE='FILE205')
      DO 400 S=1,4
         WRITE(200,*)SE(S)
         DO 450 N=1,24
           DO 450 T=1,15
             WRITE(201,*)M(S,N,T)
450    CONTINUE
400    CONTINUE
      DO 500 S=1,4
        DO 500 N=1,24
          WRITE(202,*)DIRS(S,N),LUMP(S,N)
500    CONTINUE
      DO 550 NL=1,NTOT
        WRITE(203,*)OWTW(NL)
550    CONTINUE
      WRITE(204,*)(HRS(S),S=1,4)
      WRITE(204,*)MONCT,HRCT
      DO 600 S=1,4
        DO 600 N=1,24
          WRITE(205,*)DOR(S,N),MOR(S,N)
600    CONTINUE
C  ****************************************************************
C  *  FORMAT STATEMENTS.                                          *
C  ****************************************************************
2000   FORMAT( 6X,I1,2X,3I2,4I4,8I3,11I2)
2005   FORMAT( 6X,I1,2X,3I2,I3,1X,I5,1X,I4,1X,I3,1X,I2,1X,I4,2I3,1X,
     *         I3,I4,I3,I4,I3,I3,I4)
2010   FORMAT( 6X,I1,2X,3I2,I3,1X,I5,1X,I4,1X,I3,1X,I2,1X,I4,2I3,1X,
     *         I3,I4,I3,I4,I3,I3,I4)
3000   FORMAT(I3,A3,1X,I2,T21,I1,T24,I2,T32,I6,T50,F6.3,T78,I2)
5000   FORMAT(A7,I2,A8,A4,7A8,I2)
5100   FORMAT(76A1)
5200   FORMAT('5CO',3A1,'STA',3A1,1X,67A1)
```

```
5300   FORMAT(' CO',3A1,'STA',3A1,I3,1X,A8,1X,67A1)
5400   FORMAT('1            STATION DESCRIPTION LISTING')
5500   FORMAT('1         STATIONS NOT USED TO CALCULATE RELATIONSHIPS')
       STOP
       END
       FUNCTION SGN(R)
C
C  ***************************************************************
C  *                                        *
C  *  SIGN ROUTINE                                  *
C  *                                        *
C  *  INPUT  : R - A REAL NUMBER                          *
C  *                                        *
C  *  RETURNS: 1 - IF R IS POSITIVE                       *
C  *           0 - IF R IS EQUAL TO 0                    *
C  *          -1 - IF R IS NEGATIVE                      *
C  *                                        *
C  ***************************************************************
C
       REAL R
       SGN = 0
       IF (R.LT.0) SGN = -1
       IF (R.GT.0) SGN = 1
       RETURN
       END
```

VCR2.FOR

```
$DEBUG
C  ****************************************************************
C  *                    CLASSIFICATION SUMMARY                    *
C  ****************************************************************
C  ******************  SUB PROGRAM : VCR2  ************************
C  ****************************************************************
C
      COMMON/FILES/IIN,IOUT,FILE099
      REAL*8 ROUTE(500),SN(500)
      REAL*4 COMP1,COMP2
      INTEGER CN(500),DIRS(4,24),FED(500)
      INTEGER FC(500)
      REAL*4 MP(500)
      INTEGER S,D,DP1,DM1,N,T,F
      REAL HF(2,4,24,24,14),SF(500,4,15),SFY(15),PERCOL,PERTRK
      INTEGER RM(500),R,YR(500),NL,AADT(500)
      INTEGER YF83,YF88
      INTEGER HRCT,MONCT,NTOT
      INTEGER ERRCT,OLDATE
      INTEGER ICNT,IOLD,ISTATE,BAD(500),LUMP(4,24)
      CHARACTER*14 IINF,IOUTF
      CHARACTER*11 FN1,FN2
C  ****************************************************************
C  *   INITIALIZATION OF THE VARIABLES  FOR THIS SUB-PROGRAM      *
C  ****************************************************************
      DATA SF/30000* 0./
      DATA BAD/500* 0/
      DATA LUMP/96* 0/
      NL = 0
      HRCT = 0
      MONCT = 0
      ERRCT = 0
      YF83 = 0
      YF88 = 0
      F = 0
      OLDATE = 0
      ISTATE = 0
      ICNT   = 0
      IOLD   = 0
      DO 5 R = 1,2
        DO 5 S = 1,4
          DO 5 N = 1,24
            DO 5 D = 1,24
              DO 5 T = 1,14
  5             HF(R,S,N,D,T) = 0.
C
```

```
C  ****************************************************************
C  *  READS THE DATA FROM  THE VEHICLE CLASSIFICATION TAPE        *
C  ****************************************************************
 111  FORMAT('')
      DO 6 N=1,40
        WRITE(*,111)
   6  CONTINUE
      WRITE(*,*)"                  ****** VCR2 is Running ******"
      DO 7 N=1,10
        WRITE(*,111)
   7  CONTINUE
      OPEN(99, ACCESS='SEQUENTIAL', STATUS = 'SCRATCH',
     * FORM='FORMATTED')
      OPEN(100,FILE='FILE100')
      OPEN(102,FILE='FILE102',STATUS='OLD',MODE='READ')
      OPEN(101,FILE='FILE101',STATUS='OLD',MODE='READ')
      OPEN(103,FILE='FILE103',STATUS='OLD',MODE='READ')
      OPEN(104,FILE='FILE104',STATUS='OLD',MODE='READ')
      OPEN(202,FILE='FILE202',STATUS='OLD')
      READ(102,9990)NTOT
9990   FORMAT(I3)
      DO 100 R=1,2
        DO 100 S=1,4
          DO 100 N=1,24
            DO 100 D=1,24
              DO 100 T=1,14
                READ(101,*)HF(R,S,N,D,T)
 100   CONTINUE
10000   FORMAT(I3)
10001   FORMAT(A8,2I4)
       DO 120 NL=1,NTOT
         READ(102,10001)ROUTE(NL),RM(NL)
         READ(100,*)CN(NL),SN(NL),YR(NL),FED(NL),
     *              FC(NL),AADT(NL),MP(NL)
         DO 125 S=1,4
           DO 125 T=1,15
             READ(103,*)SF(NL,S,T)
 125    CONTINUE
 120    CONTINUE
       READ(104,10005)IINF,FN1,FN2
       READ(104,10001)TROUT,OLDATE,F
       DO 150 S=1,4
         DO 150 N=1,24
          READ(202,*)DIRS(S,N),LUMP(S,N)
 150    CONTINUE
       CLOSE(100,STATUS='DELETE')
       CLOSE(101,STATUS='DELETE')
       CLOSE(102,STATUS='DELETE')
```

```
      CLOSE(103,STATUS='DELETE')
      CLOSE(104)
      CLOSE(202,STATUS='DELETE')
     OPEN(8, FILE =FN1, MODE ='WRITE',ACCESS='APPEND')
     OPEN(7, FILE = FN2, MODE ='WRITE',ACCESS='APPEND')
     OPEN(16, FILE = 'FILE016', MODE ='WRITE',ACCESS='APPEND')
C  ***************************************************************
C  *   IF ANY RELATIONSHIPS CANNOT BE  COMPUTED,  (EG.  SUMMER TO   *
C  *   WINTER)  AND  TWO RELATIONSHIPS EXIST WHICH  CONTAIN THOSE   *
C  *   SEASONS AND RELATE THEM TO ANOTHER SEASON, (EG.  SUMMER TO   *
C  *   FALL AND FALL TO WINTER)  FOLLOWING  STATEMENTS  USE THOSE   *
C  *   RELATIONS TO COMPUTE THE MISSING RELATIONSHIPS.              *
C  ***************************************************************
C
     DO 348 R = 1,2
       IR = 3 - R
       IF (HF(R,1,1,2,1).NE.0.) GOTO 300
         DO 298 T = 1,14
           COMP2 = (HF(R,1,4,1,T) - HF(R,1,4,2,T)) /
    *            (1. - HF(R,1,4,1,T) * HF(R,1,4,2,T))
           IF(HF(R,1,1,3,T).NE.0.0.AND.HF(R,1,2,3,T).NE.0.)GOTO 294
           IF(HF(R,1,1,4,T).NE.0.0.AND.HF(R,1,2,4,T).NE.0.)GOTO 292
             HF(R,1,2,1,T) = HF(IR,1,2,1,T)
           GOTO 298
 292       HF(R,1,2,1,T) = COMP2
           GOTO 298
 294       COMP1 = (HF(R,1,3,1,T) - HF(R,1,3,2,T)) /
    *            (1. - HF(R,1,3,1,T) * HF(R,1,3,2,T))
           IF(HF(R,1,1,4,T).NE.0.0.AND.HF(R,1,2,4,T).NE.0.)GOTO 296
             HF(R,1,2,1,T) = COMP1
           GOTO 298
 296       HF(R,1,2,1,T) = (COMP1 + COMP2) / 2
 298     CONTINUE
 300   IF (HF(R,1,1,3,1).NE.0.) GOTO 310
         DO 308 T = 1,14
           COMP2 = (HF(R,1,4,1,T) - HF(R,1,4,3,T)) /
    *            (1. - HF(R,1,4,1,T) * HF(R,1,4,3,T))
           IF(HF(R,1,1,2,T).NE.0.0.AND.HF(R,1,2,3,T).NE.0.)GOTO 304
           IF(HF(R,1,1,4,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 302
             HF(R,1,3,1,T) = HF(IR,1,3,1,T)
           GOTO 308
 302       HF(R,1,3,1,T) = COMP2
           GOTO 308
 304       COMP1 = (HF(R,1,2,1,T) + HF(R,1,3,2,T)) /
    *            (1. + HF(R,1,2,1,T) * HF(R,1,3,2,T))
           IF(HF(R,1,1,4,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 306
             HF(R,1,3,1,T) = COMP1
           GOTO 308
```

```
306         HF(R,1,3,1,T) = (COMP1 + COMP2) / 2
308       CONTINUE
310     IF (HF(R,1,1,4,1).NE.0.) GOTO 320
        DO 318 T = 1,14
          COMP2 = (HF(R,1,3,1,T) + HF(R,1,4,3,T)) /
     *          (1. + HF(R,1,3,1,T) * HF(R,1,4,3,T))
          IF(HF(R,1,1,2,T).NE.0.0.AND.HF(R,1,2,4,T).NE.0.)GOTO 314
          IF(HF(R,1,1,3,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 312
            HF(R,1,4,1,T) = HF(IR,1,4,1,T)
          GOTO 318
312         HF(R,1,4,1,T) = COMP2
          GOTO 318
314         COMP1 = (HF(R,1,2,1,T) + HF(R,1,4,2,T)) /
     *          (1. + HF(R,1,2,1,T) * HF(R,1,4,2,T))
          IF(HF(R,1,1,3,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 316
            HF(R,1,4,1,T) = COMP1
          GOTO 318
316         HF(R,1,4,1,T) = (COMP1 + COMP2) / 2
318       CONTINUE
320     IF (HF(R,1,2,3,1).NE.0.) GOTO 330
        DO 328 T = 1,14
          COMP2 = (HF(R,1,4,2,T) - HF(R,1,4,3,T)) /
     *          (1. - HF(R,1,4,2,T) * HF(R,1,4,3,T))
          IF(HF(R,1,1,2,T).NE.0.0.AND.HF(R,1,1,3,T).NE.0.)GOTO 324
          IF(HF(R,1,2,4,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 322
            HF(R,1,3,2,T) = HF(IR,1,3,2,T)
          GOTO 328
322         HF(R,1,3,2,T) = COMP2
          GOTO 328
324         COMP1 = (-HF(R,1,2,1,T) + HF(R,1,3,1,T)) /
     *          (1. - HF(R,1,2,1,T) * HF(R,1,3,1,T))
          IF(HF(R,1,2,4,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 326
            HF(R,1,3,2,T) = COMP1
          GOTO 328
326         HF(R,1,3,2,T) = (COMP1 + COMP2) / 2
328       CONTINUE
330     IF (HF(R,1,2,4,1).NE.0.) GOTO 340
        DO 338 T = 1,14
          COMP2 = (HF(R,1,3,2,T) + HF(R,1,4,3,T)) /
     *          (1. + HF(R,1,3,2,T) * HF(R,1,4,3,T))
          IF(HF(R,1,1,2,T).NE.0.0.AND.HF(R,1,1,4,T).NE.0.)GOTO 334
          IF(HF(R,1,2,3,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 332
            HF(R,1,4,2,T) = HF(IR,1,4,2,T)
          GOTO 338
332         HF(R,1,4,2,T) = COMP2
          GOTO 338
334         COMP1 = (-HF(R,1,2,1,T) + HF(R,1,4,1,T)) /
     *          (1. - HF(R,1,2,1,T) * HF(R,1,4,1,T))
```

```
          IF(HF(R,1,2,3,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 336
            HF(R,1,4,2,T) = COMP1
          GOTO 338
 336        HF(R,1,4,2,T) = (COMP1 + COMP2) / 2
 338      CONTINUE
 340    IF (HF(R,1,3,4,1).NE.0.) GOTO 350
        DO 348 T = 1,14
          COMP2 = (-HF(R,1,3,2,T) + HF(R,1,4,2,T)) /
     *          (1. - HF(R,1,3,2,T) * HF(R,1,4,2,T))
          IF(HF(R,1,1,3,T).NE.0.0.AND.HF(R,1,1,4,T).NE.0.)GOTO 344
          IF(HF(R,1,2,3,T).NE.0.0.AND.HF(R,1,2,4,T).NE.0.)GOTO 342
            HF(R,1,4,3,T) = HF(IR,1,4,3,T)
          GOTO 348
 342        HF(R,1,4,3,T) = COMP2
          GOTO 348
 344        COMP1 = (-HF(R,1,3,1,T) + HF(R,1,4,1,T)) /
     *          (1. - HF(R,1,3,1,T) * HF(R,1,4,1,T))
          IF(HF(R,1,2,3,T).NE.0.0.AND.HF(R,1,2,4,T).NE.0.)GOTO 346
            HF(R,1,4,3,T) = COMP1
          GOTO 348
 346        HF(R,1,4,3,T) = (COMP1 + COMP2) / 2
 348      CONTINUE
C
C  ****************************************************************
C  *   FOLLOWING  STATEMENTS   USE  THE SEASONAL RELATIONSHIPS TO   *
C  *   ESTIMATE MISSING SEASONAL COUNTS FROM EXISTING COUNTS.       *
C  ****************************************************************
 350   DO 450 NL = 1,NTOT
        R = RM(NL)
        DO 380 D = 1,4
          IF (SF(NL,D,15).NE.-10.) GOTO 380
            DO 370 T = 1,14
              DENOM = 0
              DM1 = D - 1
              IF (D.EQ.1) DM1 = 1
              DO 355 N=1,DM1
                IF (N.GE.D.OR.SF(NL,N,15).EQ.-10.) GOTO 355
                  DENOM = DENOM + 1
                IF (SF(NL,N,T).LT.0.5) GOTO 355
                  SF(NL,D,T) =(SF(NL,N,T)+1)* (1+HF(R,1,D,N,T))/
     *                (1-HF(R,1,D,N,T)) + SF(NL,D,T) - 1
 355            CONTINUE
              DP1 = D + 1
              IF (D.EQ.4) DP1 = 4
              DO 360 N=DP1,4
                IF (N.LE.D.OR.SF(NL,N,15).EQ.-10.) GOTO 360
                  DENOM = DENOM + 1
                IF (SF(NL,N,T).LT.0.5) GOTO 360
```

```
                SF(NL,D,T) =(SF(NL,N,T)+1)* (1-HF(R,1,N,D,T))/
     *             (1+HF(R,1,N,D,T)) + SF(NL,D,T) - 1
 360           CONTINUE
               IF (SF(NL,D,T).LT.0.) SF(NL,D,T) = 0
               IF (DENOM.NE.0.) SF(NL,D,T) = SF(NL,D,T)/DENOM
 370           CONTINUE
 380       CONTINUE
C
C  ***************************************************************
C  *   IF THE ESTIMATED NUMBER OF COAL TRUCKS IS GREATER THAN THE   *
C  *   NUMBER OF  ESTIMATED TRUCKS  THAT ARE OF  A SIZE  TO CARRY   *
C  *   COAL THEN  THE FOLLOWING STATEMENTS REDUCE THE  COAL TRUCK   *
C  *   ESTIMATE TO THE NUMBER OF TRUCKS THAT CAN CARRY COAL.   IF   *
C  *   ANY PART OF A COUNT IS ESTIMATED,  THE PROGRAM MAKES THAT    *
C  *   TOTAL NEGATIVE TO SIGNIFY THIS.                              *
C  ***************************************************************
C
       SFTOT = 0.
       DO 385 S=1,4
         NUMTRK = SF(NL,S,6) + SF(NL,S,7) + SF(NL,S,8) +
     *            SF(NL,S,9) + SF(NL,S,10)
         IF (SF(NL,S,14).GT.FLOAT(NUMTRK)) SF(NL,S,14) = NUMTRK
         MINUS = 1
         IF (SF(NL,S,15).LT.0.) MINUS = -1
         SF(NL,S,15)=0
         IF (YR(NL).LT.80) SF(NL,S,14)=0
         DO 385 T=1,13
           SF(NL,S,15)=SF(NL,S,15)+SF(NL,S,T)*MINUS
 385     CONTINUE
C
C  ***************************************************************
C  *   COMPUTATION OF TRUCK INFORMATION NECCESSARY FOR THE PRINTOUT *
C  ***************************************************************
C
       SFTOT = INT((ABS(SF(NL,1,15))+ABS(SF(NL,2,15))+
     *          ABS(SF(NL,3,15))+ABS(SF(NL,4,15))+2.)/4.)
       DO 390 T=1,15
      SFY(T)=INT((SF(NL,1,T)+SF(NL,2,T)+SF(NL,3,T)+SF(NL,4,T)+2.)/4.)
 390     CONTINUE
       IF (SFY(15).NE.SFTOT) SFY(15) = -SFTOT
       NUMTRK = SFY(4)+SFY(5)+SFY(6)+SFY(7)+SFY(8)+
     *          SFY(9)+SFY(10)+SFY(11)+SFY(12)+SFY(13)
       PERTRK = 0
       IF (SFTOT.NE.0.) PERTRK = (FLOAT(NUMTRK) / SFTOT) * 100
       SFY(14) = INT(SFY(14))
       PERCOL = 0
       IF (NUMTRK.NE.0) PERCOL = (SFY(14) / FLOAT(NUMTRK)) * 100
       AADT(NL) = IABS(AADT(NL))
```

```
      IF (AADT(NL).EQ.0) AADT(NL) = ABS(SFY(15))
396     IF (ABS(AADT(NL)*2.).GE.ABS(SFY(15)).AND.
     *     ABS(AADT(NL)*.5).LE.ABS(SFY(15))) GOTO 397
        WRITE(8,6070) CN(NL),SN(NL),AADT(NL),ROUTE(NL),MP(NL)
        BAD(NL) = -1
397     IF (PERTRK.LE.50.) GOTO 450
        WRITE(8,6080) CN(NL),SN(NL),NUMTRK,ROUTE(NL),MP(NL)
        BAD(NL) = -1
450   CONTINUE
10003  FORMAT(A8,4I4)
10002  FORMAT(A8,2I4)
10004  FORMAT(A6)
10005  FORMAT(A6,2X,A11,1X,A11)
      OPEN(300,FILE='FILE300')
      OPEN(301,FILE='FILE301')
      OPEN(304,FILE='FILE304')
      OPEN(305,FILE='FILE305')
      OPEN(306,FILE='FILE306')
      WRITE(304,10004)IINF
      WRITE(304,10003)TROUT,OLDATE,ICNT,NTOT,F
      DO 500 NL=1,NTOT
        WRITE(300,*)ROUTE(NL),RM(NL),BAD(NL)
        WRITE(301,*)CN(NL),SN(NL),YR(NL),FED(NL),
     *          FC(NL),AADT(NL),MP(NL)
500   CONTINUE

      DO 600 S=1,4
        DO 600 N=1,24
          WRITE(305,*)DIRS(S,N),LUMP(S,N)
600   CONTINUE
      DO 650 T=1,15
       WRITE(306,*)SFY(T)
650   CONTINUE
C  ****************************************************************
C  *   FORMAT STATEMENTS.                                         *
C  ****************************************************************
C
5400   FORMAT('1            STATION DESCRIPTION LISTING')
6070   FORMAT(1X,I3,1X,A3,' 7 :(',I8,')',A9,' AT MILEPOINT ',
     *       F8.3,' AADT EXCEEDS LIMITS')
6080   FORMAT(1X,I3,1X,A3,' 8 :(',I8,')',A9,' AT MILEPOINT ',
     *       F8.3,' TRUCKS OVER 50% OF VHCLS')
C
      STOP
      END
      FUNCTION SGN(R)
C
C  ****************************************************************
```

```
C  *                                          *
C  *  SIGN ROUTINE                                  *
C  *                                          *
C  *  INPUT  : R - A REAL NUMBER                          *
C  *                                          *
C  *  RETURNS: 1 - IF R IS POSITIVE                       *
C  *           0 - IF R IS EQUAL TO 0                    *
C  *          -1 - IF R IS NEGATIVE                      *
C  *                                          *
C  ************************************************************
C
      REAL R
      SGN = 0
      IF (R.LT.0) SGN = -1
      IF (R.GT.0) SGN = 1
      RETURN
      END
```

VCR3.FOR

```
$DEBUG
C  ****************************************************************
C  ******************  CLASSIFICATION SUMMARY **********************
C  * ************************************************************ *
C  ********************** SUB-PROGRAM : VCR3 **********************
C  ****************************************************************
C
      COMMON/FILES/IIN,IOUT,FILE099
      REAL*8 ROUTE(500),TROUT,SN(500)
      REAL*8 T1,T3,T4,T5,T6,T7,T8,T9,T10
      REAL*4 DENLOG,MP(500)
      INTEGER T2,CN(500),FED(500),DIRS(4,24),OWTW(500)
      INTEGER FC(500)
      REAL M(4,24,15),BLANK3,MULT
      INTEGER S,D,DP1,DM1,N,T,F,DIR,VC(14),OVCA(23),OVCB(15)
      REAL HF(2,4,24,24,14),SF(500,4,15),SFY(15),SE(4)
      INTEGER RM(500),R,HR,YR(500),NL,AADT(500),DOR(4,24),MOR(4,24)
      INTEGER CHRU,CHRI,YF83,YF88,HRS(4),NUMHRS(500,4)
      INTEGER MONTH,DAY,HRCT,MONCT
      INTEGER DISC(76),ERRCT,PRNT(15),COLHAL,OLDATE
      INTEGER ICNT,IOLD,ISTATE,BAD(500),LUMP(4,24)
      CHARACTER*14 IINF,IOUTF
      CHARACTER*11 FN1,FN2
C     CHARACTER*14 FILE07
C     CHARACTER*14 FILE016
C     CHARACTER*14 FILE06
C
C  ****************************************************************
C  *   INITIALIZE THE VARIABLES  FOR THIS SUB-PROGRAM             *
C  ****************************************************************
      DATA M/1440* 0./
C     DATA SF/30000* 0./
      DATA SE/4* 1./
      DATA HRS/4* 0/
      DATA FED/500* 0/
      DATA OWTW/500* 1/
      DATA BAD/500* 0/
      DATA DOR/96* 0/
      DATA MOR/96* 0/
      DATA DIRS/96* 0/
      DATA LUMP/96* 0/
      DATA CHRI/'I'/
      DATA CHRU/'U'/
      DATA BLANK/' '/
      DATA BLANK3/'   '/
      NL = 0
```

```
      HRCT = 0
      MONCT = 0
      ERRCT = 0
      YF83 = 0
      YF88 = 0
      F = 0
      OLDATE = 0
      ISTATE = 0
      ICNT   = 0
      IOLD   = 0
  111  FORMAT('')
     DO 6 N=1,40
       WRITE(*,111)
   6  CONTINUE
     WRITE(*,*)"                ****** VCR3 is Running ******"
     DO 7 N=1,10
       WRITE(*,111)
   7  CONTINUE
      OPEN(104,FILE='FILE104')
      READ(104,9999)FN1,FN2
 9999  FORMAT(8X,A11,1X,A11)
      CLOSE(104)
      OPEN(8, FILE = FN1, MODE ='WRITE',ACCESS='APPEND')
      OPEN(7, FILE = FN2, MODE ='WRITE',ACCESS='APPEND')
      OPEN(16, FILE = 'FILE016', MODE ='WRITE',ACCESS='APPEND')
C
C  ****************************************************************
C  *   READING   THE   DATA                                       *
C  ****************************************************************
10003  FORMAT(A8,4I4)
10004  FORMAT(A6)
      OPEN(304, FILE='FILE304',MODE='READ')
      READ(304,10004)IINF
      READ(304,10003)TROUT,OLDATE,ICNT,NTOT,F
      CLOSE(304,STATUS='DELETE')
      IOUTF=FILE099
      IIN = 12
      IOUT = 99
      OPEN(99, ACCESS='SEQUENTIAL', STATUS = 'SCRATCH',
     * FORM='FORMATTED')
      OPEN(IIN,FILE=IINF, STATUS='OLD', MODE='READ')
C  ****************************************************************
      OPEN(201,FILE='FILE201',MODE='READ')
      OPEN(200, FILE='FILE200',MODE='READ')
      OPEN(305,FILE='FILE305',MODE='READ')
      DO 100 S=1,4
        READ(200,*)SE(S)
        DO 110 N=1,24
```

```
          DO 110 T=1,15
            READ(201,*)M(S,N,T)
110  CONTINUE
100  CONTINUE
    CLOSE(200,STATUS='DELETE')
    CLOSE(201,STATUS='DELETE')
    DO 150 S=1,4
      DO 150 N=1,24
        READ(305,*)DIRS(S,N),LUMP(S,N)
150  CONTINUE
    CLOSE(305,STATUS='DELETE')
    OPEN(203,FILE='FILE203',MODE='READ')
    OPEN(204, FILE='FILE204',MODE='READ')
    OPEN(205,FILE='FILE205',MODE='READ')
    OPEN(300, FILE='FILE300',MODE='READ')
    OPEN(301,FILE='FILE301')
    DO 160 NL=1,NTOT
     READ(203,*)OWTW(NL)
160  CONTINUE
    DO 175 NL=1,NTOT
      READ(300,*)ROUTE(NL),RM(NL),BAD(NL)
      READ(301,*)CN(NL),SN(NL),YR(NL),FED(NL)
   *       ,FC(NL),AADT(NL),MP(NL)
175  CONTINUE
    READ(204,*)(HRS(S),S=1,4)
    READ(204,*)MONCT,HRCT
    DO 200 S=1,4
      DO 200 N=1,24
        READ(205,*)DOR(S,N),MOR(S,N)
200   CONTINUE
    CLOSE(203,STATUS='DELETE')
    CLOSE(204,STATUS='DELETE')
    CLOSE(205,STATUS='DELETE')
    CLOSE(300,STATUS='DELETE')
    CLOSE(301,STATUS='DELETE')
    OPEN(306,FILE='FILE306')
    DO 210 T=1,15
       READ(306,*)SFY(T)
210  CONTINUE
    CLOSE(306,STATUS='DELETE')
     NL = 0
     DO 505  R = 1,2
       DO 505  S = 1,4
         DO 505  N = 1,24
           DO 505  D = 1,24
             DO 505  T = 1,14
505              HF(R,S,N,D,T) = 0.
     DO 507  N = 1,500
```

```
        DO 507  S = 1,4
          DO 507  T = 1,15
            SF(N,S,T) = 0.
 507   CONTINUE
C510   CALL REREAD
      GOTO 520
 510   BACKSPACE 12
 520   READ (IIN,5000,END=590) T1,T2,TROUT,T3,T4,T5,
     *                   T6,T7,T8,T9,T10,ICSN
       IF (T2.GT.83) YF83 = 1
       IF (T2.GE.91) YF88 = 1
       YF88 = 0
       IF (ICSN-98) 525,532,535
 525   IF (F.NE.2) GOTO 526
       F = 1
       GOTO 559
 526   WRITE(IOUT,5000) T1,T2,TROUT,T3,T4,T5,T6,T7,
     *                   T8,T9,T10,ICSN
        IF (YF88.EQ.1) GOTO 527
        IF (YF83.EQ.1.AND.YF88.EQ.0) GOTO 524
C
C  ****************************************************************
C  *   READING THE  DATA  USING THE  PROPER FORMAT,               *
C  *   AND  IF NECESSARY  STATEMENTS 94  THRU 108 CONVERT THE     *
C  *   VEHICLE CODES 1 - 22 TO THE VEHICLE CODES 1 - 14.          *
C  ****************************************************************
C
        BACKSPACE 99
        READ (99,2000) DIR,MONTH,DAY,HR,(OVCA(I),I=1,23)
        VC( 1)=OVCA(22)
        VC( 2)=OVCA( 1) + OVCA( 2) + OVCA( 3)+ OVCA( 4)
        VC( 3)=OVCA( 5) + OVCA( 6)
        VC( 4)=OVCA(20) + OVCA(21)
        VC( 5)=OVCA( 7)
        VC( 6)=OVCA( 8)
        VC( 7)=OVCA( 9)
        VC( 8)=OVCA(10) + OVCA(11) + OVCA(18)
        VC( 9)=OVCA(12) + OVCA(19)
        VC(10)=OVCA(13) + OVCA(14) + OVCA(15)
        VC(11)=OVCA(16)
        VC(12)=OVCA(17)
        VC(13)=0
        VC(14)=OVCA(23)
        GOTO 528
 524     BACKSPACE 99
        READ (99,2005) DIR,MONTH,DAY,HR,(OVCB(I),I=1,15)
        VC( 1)=OVCB( 1)
        VC( 2)=OVCB( 2)
```

```
      VC( 3)=OVCB( 3)
      VC( 4)=OVCB( 4) + OVCB( 5)
      VC( 5)=OVCB( 6)
      VC( 6)=OVCB( 7)
      VC( 7)=OVCB( 8)
      VC( 8)=OVCB( 9)
      VC( 9)=OVCB(10)
      VC(10)=OVCB(11)
      VC(11)=OVCB(12)
      VC(12)=OVCB(13)
      VC(13)=OVCB(14)
      VC(14)=OVCB(15)
      GOTO 528
 527    BACKSPACE 99
      READ(99,2010) DIR,MONTH,DAY,HR,(VC(I),I=1,14)
 528    IF (HR.EQ.0) HR = 24
      IF (HR.LE.0.OR.HR.GT.24) GOTO 520
      IF (MONTH.LT.1.OR.MONTH.GT.12) GOTO 520
      DO 529 T = 1,14
        IF (VC(T).LT.0) VC(T) = 0
 529    CONTINUE
C  ****************************************************************
C  *   THE  SEASON  OF   THE  COUNT DETERMINATION.                *
C  *   THE NUMBER  OF  TRUCKS  COUNTED COMPUTATION.               *
C  *   REJECTS  THE  DATA IF THE NUMBER OF COAL TRUCK EXCEEDS THE   *
C  *   NUMBER OF TRUCKS.                                          *
C  ****************************************************************
C       IF (IABS((MONTH*31+DAY)-OLDATE).GT.15) OLDATE = MONTH*31+DAY
      ISEAS = MOD(INT(FLOAT(MONTH)/3.),4) + 1
      NUMTRK = VC(6) + VC(7) + VC(8) + VC(9) + VC(10)
      IF (VC(14).GT.NUMTRK) GOTO 520
C  ****************************************************************
C  *   DETERMINATION OF THE NUMBER OF DIRECTIONS COUNTED, AND      *
C  *   AVERAGING THE DATA IF MORE THAN  ONE COUNT IS TAKEN FOR A    *
C  *   PARTICULAR HOUR.                                           *
C  ****************************************************************
      DENOM = 1.
      DIRS(ISEAS,HR) = DIRS(ISEAS,HR) * 10 + DIR
      IF (DIRS(ISEAS,HR).GE.100) DENOM = .6666666
      M(ISEAS,HR,15) = 0
      DO 530 I=1,13
        M(ISEAS,HR,I)=(M(ISEAS,HR,I) + VC(I)) * DENOM
        M(ISEAS,HR,15)=M(ISEAS,HR,15)+M(ISEAS,HR,I)
 530    CONTINUE
      M(ISEAS,HR,14)=(M(ISEAS,HR,14) + VC(14)) * DENOM
      F = 1
      SE(ISEAS) = SE(ISEAS) * (.9 + .2111 * SGN(M(ISEAS,HR,15)))
      IF (DIRS(ISEAS,HR).LT.10) GOTO 510
```

```
        IF (MOD(DIRS(ISEAS,HR),11).NE.0) OWTW(NL+1) = 2
      GOTO 520
C  ***************************************************************
C  *   FOLLOWING STATEMENTS  ARE  EXECUTED WHEN  ALL  DATA  FOR A   *
C  *   SPECIFIC  LOCATION HAS BEEN  READ,  CLOSES THE DATA SET, AND *
C  *   WRITES  A VERBAL LOCATION DESCRIPTION TO THE FISRT DISK FILE *
C  ***************************************************************
 532   IF (F.EQ.2) GOTO 520
      IF (F.EQ.0) GOTO 559
      F = 2
      WRITE(99,5000) T1,T2,TROUT,T3,T4,T5,T6,T7,
     *                T8,T9,T10,ICSN
      BACKSPACE 99
      READ (99,5100) (DISC(I),I=1,76)
C     WRITE(16,5200) (DISC(I),I=1,6),(DISC(I),I=10,76)
      GOTO 520
C
C  ***************************************************************
C  *   STATEMENTS TO  DECIDE  THE ROAD TYPE                      *
C  ***************************************************************
 535   IF (F.EQ.0) GOTO 520
        IF (F.EQ.1) GOTO 559
        NL = NL + 1
C        WRITE( 8,5300) (DISC(I),I=1,6),T2,TROUT,(DISC(I),I=10,76)
        WRITE(99,5000) T1,T2,TROUT
        BACKSPACE 99
        READ (99,5100) (DISC(I),I=1,17)
        R = 2
        DO 540 I = 10,14
          IF (DISC(I).EQ.BLANK) GOTO 540
          IF (DISC(I).EQ.CHRU.OR.DISC(I).EQ.CHRI) R = 1
          GOTO 545
 540      CONTINUE
C
C  ***************************************************************
C  *   TO DECIDE IF THE LOCATION IS A COAL ROAD OR NOT           *
C  ***************************************************************
C
 545      RM(NL) = R
        NUMCOL = 0
        NUMTRK = 0
        DO 550 S = 1,4
          DO 550 N = 1,24
            NUMCOL = NUMCOL + M(S,N,14)
            DO 550 T = 4,13
              NUMTRK = NUMTRK + M(S,N,T)
 550      CONTINUE
        NUMCOL = NUMCOL * 100
```

```
      COLHAL = 0
      IF (NUMCOL.GE.NUMTRK) COLHAL = 1
C
C  ****************************************************************
C  *   TO  COMPUTE HOURLY CONVERSION FACTORS FOR EXISTING DATA     *
C  ****************************************************************
C
      DO 558 S = 1,4
        IF (SE(S).LE.1.) SF(NL,S,15) = -10
        IF (SE(S).LE.1.) GOTO 558
        IF (BAD(NL).EQ.-1) GOTO 558
        DO 557 D = 1,23
          IF (M(S,D,15).EQ.0.) GOTO 557
          DP1 = D + 1
          DO 555 N=DP1,24
            IF (M(S,N,15).EQ.0.) GOTO 555
              DO 552 T=1,13
                DENOM = M(S,N,T) + M(S,D,T) + 2
                DENLOG = ALOG10(DENOM)
                HF(R,S,N,D,T) = (HF(R,S,N,D,T)*HF(R,S,D,N,T)+
     *            (((M(S,N,T)-M(S,D,T))*DENLOG)/DENOM))/
     *             (HF(R,S,D,N,T)+DENLOG)
                HF(R,S,D,N,T) = HF(R,S,D,N,T) + DENLOG
  552           CONTINUE
              IF (COLHAL.EQ.0) GOTO 555
                DENOM = M(S,N,14) + M(S,D,14) + 2
                DENLOG = ALOG10(DENOM)
                HF(R,S,N,D,14)=(HF(R,S,N,D,14)*HF(R,S,D,N,14)+
     *            (((M(S,N,14)-M(S,D,14))*DENLOG)/DENOM))/
     *             (HF(R,S,D,N,14)+DENLOG)
                HF(R,S,D,N,14) = HF(R,S,D,N,14) + DENLOG
  555         CONTINUE
  557       CONTINUE
  558   CONTINUE
C
C  ****************************************************************
C  *   STORE AND RESET THE DATA SET.                              *
C  ****************************************************************
C
  559   DO 570 S = 1,4
          DO 560 D = 1,24
            DIRS(S,D) = 0
            DO 560 T = 1,15
              M(S,D,T) = 0.
  560       CONTINUE
  570   SE(S) = 1
      OLDATE = 0
      F = 0
```

```
      GOTO 520
C  ***************************************************************
C  *   TO CALCULATE RATIOS FOR HOURS IN WHICH DIRECT RELATIONSHIPS  *
C  *   ARE NOT AVAILABLE (I.E. IF HOUR 1 TO 3 IS NOT KNOWN BUT 1 TO *
C  *   2 AND 2 TO 3 ARE,  1 TO 3 MAY BE CALCULATED                 *
C  ***************************************************************
 590   NL = 0
      DO 598 R = 1,2
        DO 598 S = 1,4
          DO 598 D = 1,23
            DP1 = D + 1
            DO 598 N = DP1,24
              DO 598 T = 1,14
                IF (HF(R,S,D,N,T).NE.0.) GOTO 598
                DENOM = SGN(HF(R,1,D,N,T)) + SGN(HF(R,2,D,N,T))
     *                + SGN(HF(R,3,D,N,T)) + SGN(HF(R,4,D,N,T))
                IF (DENOM.EQ.0.) GOTO 595
                HF(R,S,N,D,T)=(HF(R,1,N,D,T) + HF(R,2,N,D,T)
     *                 + HF(R,3,N,D,T) + HF(R,4,N,D,T))/DENOM
                HF(R,S,D,N,T)=(HF(R,1,D,N,T) + HF(R,2,D,N,T)
     *                 + HF(R,3,D,N,T) + HF(R,4,D,N,T))/DENOM
                GOTO 598
 595              IR = 3 - R
                HF(R,S,N,D,T) = HF(IR,S,N,D,T)
                HF(R,S,D,N,T) = HF(IR,S,D,N,T)
 598   CONTINUE
 600   CONTINUE
C  ***************************************************************
C  *   READ THE DATA OFF THE DISK A SECOND TIME, DUE TO THE FACT   *
C  *   THAT THERE IS  TOO MUCH RAW DATA TO BE STORED               *
C  ***************************************************************
C
      REWIND 12
      BACKSPACE 12
 610   READ (12,5000,END=690) T1,T2,TROUT,T3,T4,T5,
     *                 T6,T7,T8,T9,T10,ICSN
      IF (ICSN-98) 620,634,635
 620   IF (F.NE.2) GOTO 621
      F = 1
      GOTO 663
 621   WRITE(99,5000) T1,T2,TROUT,T3,T4,T5,T6,T7,
     *                 T8,T9,T10,ICSN
      IF (YF88.EQ.1) GOTO 625
      IF (YF83.EQ.1.AND.YF88.EQ.0) GOTO 624
C
C  ***************************************************************
C  *   READ THE DATA USING THE PROPER FORMAT, AND IF NECESSARY     *
C  *   CONVERT THE VEHICLE CODES 1-22 TO THE VEHICLE CODES 1-14    *
```

```
C  ***************************************************************
C
      BACKSPACE 99
      READ (99,2000) DIR,MONTH,DAY,HR,(OVCA(I),I=1,23)
      VC( 1)=OVCA(22)
      VC( 2)=OVCA( 1) + OVCA( 2) + OVCA( 3)+ OVCA( 4)
      VC( 3)=OVCA( 5) + OVCA( 6)
      VC( 4)=OVCA(20) + OVCA(21)
      VC( 5)=OVCA( 7)
      VC( 6)=OVCA( 8)
      VC( 7)=OVCA( 9)
      VC( 8)=OVCA(10) + OVCA(11) + OVCA(18)
      VC( 9)=OVCA(12) + OVCA(19)
      VC(10)=OVCA(13) + OVCA(14) + OVCA(15)
      VC(11)=OVCA(16)
      VC(12)=OVCA(17)
      VC(13)=0
      VC(14)=OVCA(23)
      GOTO 626
624     BACKSPACE 99
      READ (99,2005) DIR,MONTH,DAY,HR,(OVCB(I),I=1,15)
      VC( 1)=OVCB( 1)
      VC( 2)=OVCB( 2)
      VC( 3)=OVCB( 3)
      VC( 4)=OVCB( 4) + OVCB( 5)
      VC( 5)=OVCB( 6)
      VC( 6)=OVCB( 7)
      VC( 7)=OVCB( 8)
      VC( 8)=OVCB( 9)
      VC( 9)=OVCB(10)
      VC(10)=OVCB(11)
      VC(11)=OVCB(12)
      VC(12)=OVCB(13)
      VC(13)=OVCB(14)
      VC(14)=OVCB(15)
      GOTO 626
625     BACKSPACE 99
      READ(99,2010) DIR,MONTH,DAY,HR,(VC(I),I=1,14)
626     IF (HR.EQ.0) HR = 24
      IF (HR.GE.0.AND.HR.LE.24) GOTO 628
        HRCT = HRCT + 1
        GOTO 610
628     IF (MONTH.GT.0.AND.MONTH.LE.12) GOTO 629
        MONCT = MONCT + 1
        GOTO 610
629     DO 630 T = 1,14
        IF (VC(T).LT.0) VC(T) = 0
630     CONTINUE
```

```
C
C  ***************************************************************
C  *   STORE THE SEASON, MONTH, DAY, HOUR, AND DIRECTION OF TRAFFIC *
C  *   FLOW FOR THE LOCATION.                                      *
C  ***************************************************************
C
       ISEAS = MOD(INT(FLOAT(MONTH)/3.),4) + 1
       DOR(ISEAS,HR) = DAY
       MOR(ISEAS,HR) = MONTH
       DENOM = 1
       DIRS(ISEAS,HR) = DIRS(ISEAS,HR) * 10 + DIR
       IF (DIRS(ISEAS,HR).GE.100) DENOM = .666666
       M(ISEAS,HR,15) = 0
       DO 631 I=1,13
         M(ISEAS,HR,I)=(M(ISEAS,HR,I) + VC(I))*DENOM
         M(ISEAS,HR,15) = M(ISEAS,HR,15) + M(ISEAS,HR,I)
 631     CONTINUE
       M(ISEAS,HR,14)=(M(ISEAS,HR,14) + VC(14))*DENOM
       F = 1
       IF (SF(NL+1,ISEAS,15).GT.-10) GOTO 610
         ICNT = 1
         LUMP(ISEAS,HR) = 1
     GOTO 610
C
C  ***************************************************************
C  *   FOLLOWING  STATEMENTS  ARE  EXECUTED  WHEN ALL  DATA FOR A   *
C  *   SPECIFIC  LOCATION HAS BEEN  READ.                          *
C  ***************************************************************
C
 634   IF (F.EQ.0) GOTO 610
      F = 2
      GOTO 610
 635   IF (F.EQ.0) GOTO 610
       IF (F.EQ.1) GOTO 663
       NL = NL + 1
       ROUTE(NL) = TROUT
       WRITE(99,5000) T1,T2,ROUTE(NL),T3,T4,T5,
     *                  T6,T7,T8,T9,T10,ICSN
       BACKSPACE 99
       READ(99,3000) CN(NL),SN(NL),YR(NL),
     *              FED(NL),FC(NL),AADT(NL),MP(NL)
C
C  ***************************************************************
C  *   USE THE HOURLY FACTORS COMPUTED PREVIOUSLY TO SUPPLY INFOR-  *
C  *   MATION FOR THOSE HOURS FOR WHICH THERE IS NO COLLECTED DATA  *
C  ***************************************************************
C
       IF (ICNT.EQ.0) GOTO 643
```

```
      DO 641 S = 1,4
        ICNT = 0
        DO 636 T = 1,14
          SF(NL,S,T) = 0.
636       CONTINUE
        DO 638 N = 1,24
          IF (LUMP(S,N).EQ.0) GOTO 638
          ICNT = ICNT + 1
          DO 637 T = 1,14
            SF(NL,S,T) = SF(NL,S,T) + M(S,N,T)
637         CONTINUE
638       CONTINUE
        IF (ICNT.EQ.0) GOTO 641
          R = 0
          SF(NL,S,15) = -1
          DO 640 N = 1,24
            IF (LUMP(S,N).EQ.0) GOTO 640
            LUMP(S,N) = 0
            R = R + 1
            M(S,N,15) = 0
            DO 639 T = 1,13
            M(S,N,T)=INT(FLOAT(R)*SF(NL,S,T)/FLOAT(ICNT))-
     *              INT(FLOAT(R-1)*SF(NL,S,T)/FLOAT(ICNT))
            M(S,N,15) = M(S,N,15) + M(S,N,T)
639           CONTINUE
           M(S,N,14)=INT(FLOAT(R)*SF(NL,S,14)/FLOAT(ICNT))-
     *              INT(FLOAT(R-1)*SF(NL,S,14)/FLOAT(ICNT))
640       CONTINUE
        ICNT = 0
641     CONTINUE
643     R = RM(NL)
      F = 0
      DO 658 S = 1,4
        IF (SF(NL,S,15).EQ.-10.) GOTO 658
        DO 655 D = 1,24
          IF (DIRS(S,D).GT.0) HRS(S) = HRS(S) + 1
          MULT = 1.
          IF (OWTW(NL).EQ.1.AND.DIRS(S,D).GE.10) MULT = 0.5
          IF (OWTW(NL).EQ.2.AND.DIRS(S,D).LT.10) MULT = 2.0
          DO 644 T = 1,15
            M(S,D,T) = M(S,D,T) * MULT
644         CONTINUE
          IF (M(S,D,15).NE.0.) GOTO 655
          DO 652 T=1,14
            DENOM = 0.
            DM1 = D - 1
            IF (D.EQ.1) DM1 = 1
            DO 645 N=1,DM1
```

```
                IF (N.GE.D.OR.M(S,N,15).EQ.0.) GOTO 645
                DENOM = DENOM + 1.
                IF (M(S,N,T).LT.0.5) GOTO 645
                  M(S,D,T) = (M(S,N,T)+1)*(1+HF(R,S,D,N,T))/
     *              (1-HF(R,S,D,N,T)) + M(S,D,T) - 1
  645           CONTINUE
              DP1 = D + 1
              IF (D.EQ.24) DP1 = 24
              DO 650 N=DP1,24
                IF (N.LE.D.OR.M(S,N,15).EQ.0.) GOTO 650
                DENOM = DENOM + 1.
                IF (M(S,N,T).LT.0.5) GOTO 650
                  M(S,D,T) = (M(S,N,T)+1)*(1-HF(R,S,N,D,T))/
     *              (1+HF(R,S,N,D,T)) + M(S,D,T) - 1
  650           CONTINUE
              IF(M(S,D,T).LT.0.) M(S,D,T) = 0
              IF(DENOM.NE.0.) M(S,D,T) = M(S,D,T) / DENOM
  652         CONTINUE
  655       CONTINUE
      IMP = MP(NL) * 1000
      WRITE (16,4150) CN(NL),SN(NL),ROUTE(NL),IMP,YR(NL)
C     WRITE(900,*)CN(NL),SN(NL),ROUTE(NL)
      SF(NL,S,15) = 0
      DO 657 D=1,24
        ID = D-1
        IF (ID.EQ.0) ID = 24
        DO 656 T=1,13
          IF (M(S,ID,15).LE.0.1) M(S,ID,15)=M(S,ID,15)-M(S,ID,T)
          PRNT(T) = INT(M(S,ID,T))
  656     CONTINUE
        NUMTRK = M(S,ID,6)+M(S,ID,7)+M(S,ID,8)+M(S,ID,9)+M(S,ID,10)
        IF (M(S,ID,14).GT.FLOAT(NUMTRK)) M(S,ID,14) = NUMTRK
        PRNT(14) = INT(M(S,ID,14))
        PRNT(15) = INT(M(S,ID,15))
        IF (M(S,ID,15).LT.0.) SF(NL,S,15) = -1
        IREC = 4
        DM1 = D - 1
        WRITE (16,4300) IREC,MOR(S,ID),DOR(S,ID),DM1,
     *                  (PRNT(T),T=1,15)
  657   CONTINUE
  658 CONTINUE
C
C  ***************************************************************
C  *   FOLLOWING STATEMENTS COMPUTE AND STORE THE TOTALS  FOR  THE  *
C  *   DATA SET, AND RESET THE ARRAYS FOR READING THE NEXT LOCATION *
C  ***************************************************************
C
  659 DO 660 S = 1,4
```

```
      IF (NL.EQ.0) GOTO 663
        NUMHRS(NL,S) = HRS(S)
        DO 660 T = 1,14
          SF(NL,S,T) = 0
          DO 660 D = 1,24
            SF(NL,S,T) = SF(NL,S,T) + M(S,D,T)
660   CONTINUE
663   DO 670 S = 1,4
      HRS(S) = 0
      SE(S) = 1
      DO 670 D = 1,24
        DOR(S,D) = 0
        MOR(S,D) = 0
        DIRS(S,D) = 0
        LUMP(S,D) = 0
        DO 670 T = 1,15
          M(S,D,T) = 0
670   CONTINUE
    ICNT = 0
    GOTO 610
690   NTOT = NL
    DO 700 R = 1,2
      DO 700 N = 1,4
        DO 700 D = 1,4
          DO 700 T = 1,14
            HF(R,1,N,D,T) = 0
700   CONTINUE
    ENDFILE 16
C
C  ****************************************************************
C  *   USE SEASONAL TOTALS TO COMPUTE RELATIONSHIPS BETWEEN SEASONS *
C  *   FOR  EACH  OF THE  VEHICLE  TYPES                            *
C  ****************************************************************
C
    AMIN = .66667
    AMAX = 1.5
    AFMIN = (AMIN-1) / (AMIN+1)
    AFMAX = (AMAX-1) / (AMAX+1)
    BMIN = .25
    BMAX = 4.0
    BFMIN = (BMIN-1) / (BMIN+1)
    BFMAX = (BMAX-1) / (BMAX+1)
    DO 790 NL = 1,NTOT
      R = RM(NL)
      DO 790 D = 1,3
        IF (SF(NL,D,15).EQ.-10.) GOTO 790
        DP1 = D + 1
        DO 780 N = DP1,4
```

```
            IF (SF(NL,N,15).EQ.-10.) GOTO 780
            DO 770 T = 1,14
              DENOM = SF(NL,N,T) + SF(NL,D,T) + 2
              FACT = (SF(NL,N,T) - SF(NL,D,T)) / DENOM
              DENLOG = ALOG10(DENOM)
              IF (T.EQ.1.OR.T.EQ.4) GOTO 750
              IF (FACT.LT.AFMIN) FACT = AFMIN
              IF (FACT.GT.AFMAX) FACT = AFMAX
              GOTO 760
 750            IF (FACT.LT.BFMIN) FACT = BFMIN
              IF (FACT.GT.BFMAX) FACT = BFMAX
 760            HF(R,1,N,D,T) = (HF(R,1,N,D,T)*HF(R,1,D,N,T)+
     *             FACT*DENLOG)/(HF(R,1,D,N,T)+DENLOG)
              HF(R,1,D,N,T) = HF(R,1,D,N,T) + DENLOG
 770          CONTINUE
 780        CONTINUE
 785      CONTINUE
 790    CONTINUE
        CLOSE(201)
        CLOSE(200)
        CLOSE(203)
        CLOSE(204)
        CLOSE(205)
        CLOSE(300)
        CLOSE(305)
C
        OPEN(401,FILE='FILE401')
        OPEN(402,FILE='FILE402')
        OPEN(406,FILE='FILE406')

        WRITE(401,*)NTOT,HRCT,MONCT,F
        DO 800 NL=1,NTOT
          WRITE(402,*)CN(NL),SN(NL),YR(NL),FED(NL),
     *         FC(NL),AADT(NL),MP(NL)
9990      FORMAT(A8,2I4)
          WRITE(406,9990)ROUTE(NL),RM(NL),BAD(NL)
 800    CONTINUE
        CLOSE(401)
        CLOSE(402)
        CLOSE(406)
        OPEN(403,FILE='FILE403')
        OPEN(404,FILE='FILE404')
        OPEN(405,FILE='FILE405')
        OPEN(407,FILE='FILE407')
        DO 850 R=1,2
          DO 850 S=1,4
            DO 850 N=1,24
              DO 850 D=1,24
```

```
              DO 850 T=1,14
                WRITE(403,*)HF(R,S,N,D,T)
 850   CONTINUE
      DO 900 NL=1,NTOT
         DO 900 S=1,4
           WRITE(405,*)NUMHRS(NL,S)
           DO 900 T=1,15
             WRITE(404,*)SF(NL,S,T)
 900   CONTINUE
      DO 950 T=1,15
        WRITE(407,*)SFY(T)
 950   CONTINUE
C  ****************************************************************
2000   FORMAT( 6X,I1,2X,3I2,4I4,8I3,11I2)
2005   FORMAT( 6X,I1,2X,3I2,I3,1X,I5,1X,I4,1X,I3,1X,I2,1X,I4,2I3,1X,
     *         I3,I4,I3,I4,I3,I3,I4)
2010   FORMAT( 6X,I1,2X,3I2,I3,1X,I5,1X,I4,1X,I3,1X,I2,1X,I4,2I3,1X,
     *         I3,I4,I3,I4,I3,I3,I4)
3000   FORMAT(I3,A3,1X,I2,T21,I1,T24,I2,T32,I6,T50,F6.3,T78,I2)
4150   FORMAT('1CO',I3,'STA',A3,'RTE ',A8,'MP',I6,'YR',I2)
4300   FORMAT(I1,3I2,I4,I6,I5,I4,I5,3I4,I5,5I4,I7,1X,I3,1X,A3,1X,I2,
     *        T88,I1)
5000   FORMAT(A7,I2,A8,A4,7A8,I2)
5100   FORMAT(76A1)
5200   FORMAT('5CO',3A1,'STA',3A1,1X,67A1)
5300   FORMAT(' CO',3A1,'STA',3A1,I3,1X,A8,1X,67A1)
      CLOSE(12,STATUS='DELETE')
C
      STOP
      END
      FUNCTION SGN(R)
C
C  ****************************************************************
C  *                                          *
C  *  SIGN ROUTINE                                    *
C  *                                          *
C  *  INPUT  : R - A REAL NUMBER                          *
C  *                                          *
C  *  RETURNS: 1 - IF R IS POSITIVE                       *
C  *           0 - IF R IS EQUAL TO 0                   *
C  *          -1 - IF R IS NEGATIVE                     *
C  *                                          *
C  ****************************************************************
C
      REAL R
      SGN = 0
      IF (R.LT.0) SGN = -1
      IF (R.GT.0) SGN = 1
```

```
RETURN
END
```

VCR4.FOR

```
$DEBUG
C  ****************************************************************
C  ******************** CLASSIFICATION SUMMARY **********************
C  ****************************************************************
C  ********************** SUB-PROGRAM : VCR4  ************************
C  ****************************************************************
      COMMON/FILES/IIN,IOUT,FILE099
      REAL*8 ROUTE(1500),SN(1500)
      REAL*4 COMP1,COMP2
      INTEGER CN(1500),FED(1500),OWTW(1500)
      INTEGER FC(1500),REC,NTOT
      REAL MP(1500),BLANK3
      INTEGER S,D,DP1,DM1,N,T,F
      REAL HF(2,4,24,24,14),SF(1500,4,15),SFY(115),PERCOL,PERTRK
      INTEGER RM(1500),R,YR(1500),NL,AADT(1500)
      INTEGER YF83,YF88,NUMHRS(1500,4)
      INTEGER MONTH,DAY,HOUR,HRCT,MONCT
      INTEGER ERRCT,PRNT(115),BAD(1500)
     CHARACTER*84 TEXT
     CHARACTER*4 XF(13)
     CHARACTER*8 FN
     CHARACTER*11 FN1
     CHARACTER*11 FN2,FN3,FN4
     CHARACTER*2 YR1
     INTEGER NN,IFC,IN,K
     DATA (XF(I),I=1,13)/'FC01','FC02','FC06','FC07','FC08','FC09',
    *'FC11','FC12','FC14','FC16','FC17','FC18','FC19'/
C
C  ****************************************************************
C  *   INITIALIZE  THE VARIABLES  FOR THIS SUB-PROGRAM            *
C  ****************************************************************
C     DATA SF/30000* 0./
C     DATA FED/500* 0/
      DATA OWTW/1500* 1/
      DATA BLANK3/'   '/
      NL = 0
      HRCT = 0
      MONCT = 0
      ERRCT = 0
      YF83 = 0
      YF88 = 0
      F = 0
C     OLDATE = 0
C     ISTATE = 0
C     ICNT   = 0
C     IOLD   = 0
```

```
      OPEN(104,FILE='FILE104')
      READ(104,9999)FN2,FN3
      BACKSPACE 104
      READ(104,9998)YR1
 9998  FORMAT(T14,A2)
 9999  FORMAT(8X,A11,1X,A11)
      CLOSE(104,STATUS='DELETE')
      FN4='VCRYR'//YR1//'.OUT'
      OPEN(8, FILE = FN2, MODE ='WRITE',ACCESS='APPEND')
      OPEN(7, FILE = FN3, MODE ='WRITE',ACCESS='APPEND')
      OPEN(16, FILE = 'FILE016', MODE ='WRITE',ACCESS='APPEND')
C  ****************************************************************
C  *   LOAD DATA FILES                                            *
C  ****************************************************************
C
      OPEN(401,FILE='FILE401',MODE='READ')
      OPEN(402,FILE='FILE402',MODE='READ')
      OPEN(403,FILE='FILE403',MODE='READ')
      OPEN(404,FILE='FILE404',MODE='READ')
      OPEN(405,FILE='FILE405',MODE='READ')
      OPEN(406,FILE='FILE406')
      OPEN(407,FILE='FILE407',MODE='READ')

 111  FORMAT('')
     DO 6 N=1,40
       WRITE(*,111)
  6  CONTINUE
     WRITE(*,*)"                ****** VCR4 is Running ******"
     DO 7 N=1,10
       WRITE(*,111)
  7  CONTINUE


      DO 100 R=1,2
        DO 100 S=1,4
          DO 100 N=1,24
            DO 100 D=1,24
              DO 100 T=1,14
                READ(403,*)HF(R,S,N,D,T)
 100   CONTINUE
      READ(401,*)NTOT,HRCT,MONCT,F
      DO 150  NL=1,NTOT
        READ(402,*)CN(NL),SN(NL),YR(NL),FED(NL),
    *       FC(NL),AADT(NL),MP(NL)
        READ(406,9111)ROUTE(NL),RM(NL),BAD(NL)
9111      FORMAT(A8,2I4)
 150   CONTINUE
      DO 200 NL=1,NTOT
```

```
      DO 200 S=1,4
        READ(405,*)NUMHRS(NL,S)
        DO 200 T=1,15
          READ(404,*)SF(NL,S,T)
200   CONTINUE
      DO 225 T=1,15
        READ(407,*)SFY(T)
225   CONTINUE
      CLOSE(401,STATUS='DELETE')
      CLOSE(402,STATUS='DELETE')
      CLOSE(403,STATUS='DELETE')
      CLOSE(404,STATUS='DELETE')
      CLOSE(405,STATUS='DELETE')
      CLOSE(406,STATUS='DELETE')
      CLOSE(407,STATUS='DELETE')
C  ***************************************************************
C  *   IF ANY RELATIONSHIPS CANNOT BE  COMPUTED,  (EG.  SUMMER TO   *
C  *   WINTER)  AND  TWO RELATIONSHIPS EXIST WHICH  CONTAIN THOSE   *
C  *   SEASONS AND RELATE THEM TO ANOTHER SEASON, (EG.  SUMMER TO   *
C  *   FALL AND FALL TO WINTER)  FOLLOWING  STATEMENTS  USE THOSE   *
C  *   RELATIONS TO COMPUTE THE MISSING RELATIONSHIPS.              *
C  ***************************************************************
C
      DO 848 R = 1,2
        IR = 3 - R
        IF (HF(R,1,1,2,1).NE.0.) GOTO 800
          DO 798 T = 1,14
            COMP2 = (HF(R,1,4,1,T) - HF(R,1,4,2,T)) /
     *           (1. - HF(R,1,4,1,T) * HF(R,1,4,2,T))
            IF(HF(R,1,1,3,T).NE.0.0.AND.HF(R,1,2,3,T).NE.0.)GOTO 794
            IF(HF(R,1,1,4,T).NE.0.0.AND.HF(R,1,2,4,T).NE.0.)GOTO 792
              HF(R,1,2,1,T) = HF(IR,1,2,1,T)
            GOTO 798
792         HF(R,1,2,1,T) = COMP2
            GOTO 798
794         COMP1 = (HF(R,1,3,1,T) - HF(R,1,3,2,T)) /
     *           (1. - HF(R,1,3,1,T) * HF(R,1,3,2,T))
            IF(HF(R,1,1,4,T).NE.0.0.AND.HF(R,1,2,4,T).NE.0.)GOTO 796
              HF(R,1,2,1,T) = COMP1
            GOTO 798
796         HF(R,1,2,1,T) = (COMP1 + COMP2) / 2
798       CONTINUE
800     IF (HF(R,1,1,3,1).NE.0.) GOTO 810
          DO 808 T = 1,14
            COMP2 = (HF(R,1,4,1,T) - HF(R,1,4,3,T)) /
     *           (1. - HF(R,1,4,1,T) * HF(R,1,4,3,T))
            IF(HF(R,1,1,2,T).NE.0.0.AND.HF(R,1,2,3,T).NE.0.)GOTO 804
            IF(HF(R,1,1,4,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 802
```

```
        HF(R,1,3,1,T) = HF(IR,1,3,1,T)
      GOTO 808
802     HF(R,1,3,1,T) = COMP2
      GOTO 808
804     COMP1 = (HF(R,1,2,1,T) + HF(R,1,3,2,T)) /
     *       (1. + HF(R,1,2,1,T) * HF(R,1,3,2,T))
      IF(HF(R,1,1,4,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 806
        HF(R,1,3,1,T) = COMP1
      GOTO 808
806     HF(R,1,3,1,T) = (COMP1 + COMP2) / 2
808   CONTINUE
810   IF (HF(R,1,1,4,1).NE.0.) GOTO 820
      DO 818 T = 1,14
        COMP2 = (HF(R,1,3,1,T) + HF(R,1,4,3,T)) /
     *       (1. + HF(R,1,3,1,T) * HF(R,1,4,3,T))
      IF(HF(R,1,1,2,T).NE.0.0.AND.HF(R,1,2,4,T).NE.0.)GOTO 814
      IF(HF(R,1,1,3,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 812
        HF(R,1,4,1,T) = HF(IR,1,4,1,T)
      GOTO 818
812     HF(R,1,4,1,T) = COMP2
      GOTO 818
814     COMP1 = (HF(R,1,2,1,T) + HF(R,1,4,2,T)) /
     *       (1. + HF(R,1,2,1,T) * HF(R,1,4,2,T))
      IF(HF(R,1,1,3,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 816
        HF(R,1,4,1,T) = COMP1
      GOTO 818
816     HF(R,1,4,1,T) = (COMP1 + COMP2) / 2
818   CONTINUE
820   IF (HF(R,1,2,3,1).NE.0.) GOTO 830
      DO 828 T = 1,14
        COMP2 = (HF(R,1,4,2,T) - HF(R,1,4,3,T)) /
     *       (1. - HF(R,1,4,2,T) * HF(R,1,4,3,T))
      IF(HF(R,1,1,2,T).NE.0.0.AND.HF(R,1,1,3,T).NE.0.)GOTO 824
      IF(HF(R,1,2,4,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 822
        HF(R,1,3,2,T) = HF(IR,1,3,2,T)
      GOTO 828
822     HF(R,1,3,2,T) = COMP2
      GOTO 828
824     COMP1 = (-HF(R,1,2,1,T) + HF(R,1,3,1,T)) /
     *       (1. - HF(R,1,2,1,T) * HF(R,1,3,1,T))
      IF(HF(R,1,2,4,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 826
        HF(R,1,3,2,T) = COMP1
      GOTO 828
826     HF(R,1,3,2,T) = (COMP1 + COMP2) / 2
828   CONTINUE
830   IF (HF(R,1,2,4,1).NE.0.) GOTO 840
      DO 838 T = 1,14
        COMP2 = (HF(R,1,3,2,T) + HF(R,1,4,3,T)) /
```

```
     *            (1. + HF(R,1,3,2,T) * HF(R,1,4,3,T))
                IF(HF(R,1,1,2,T).NE.0.0.AND.HF(R,1,1,4,T).NE.0.)GOTO 834
                IF(HF(R,1,2,3,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 832
                  HF(R,1,4,2,T) = HF(IR,1,4,2,T)
                GOTO 838
  832           HF(R,1,4,2,T) = COMP2
                GOTO 838
  834           COMP1 = (-HF(R,1,2,1,T) + HF(R,1,4,1,T)) /
     *            (1. - HF(R,1,2,1,T) * HF(R,1,4,1,T))
                IF(HF(R,1,2,3,T).NE.0.0.AND.HF(R,1,3,4,T).NE.0.)GOTO 836
                  HF(R,1,4,2,T) = COMP1
                GOTO 838
  836           HF(R,1,4,2,T) = (COMP1 + COMP2) / 2
  838         CONTINUE
  840       IF (HF(R,1,3,4,1).NE.0.) GOTO 850
            DO 848 T = 1,14
              COMP2 = (-HF(R,1,3,2,T) + HF(R,1,4,2,T)) /
     *            (1. - HF(R,1,3,2,T) * HF(R,1,4,2,T))
              IF(HF(R,1,1,3,T).NE.0.0.AND.HF(R,1,1,4,T).NE.0.)GOTO 844
              IF(HF(R,1,2,3,T).NE.0.0.AND.HF(R,1,2,4,T).NE.0.)GOTO 842
                HF(R,1,4,3,T) = HF(IR,1,4,3,T)
              GOTO 848
  842           HF(R,1,4,3,T) = COMP2
              GOTO 848
  844           COMP1 = (-HF(R,1,3,1,T) + HF(R,1,4,1,T)) /
     *            (1. - HF(R,1,3,1,T) * HF(R,1,4,1,T))
              IF(HF(R,1,2,3,T).NE.0.0.AND.HF(R,1,2,4,T).NE.0.)GOTO 846
                HF(R,1,4,3,T) = COMP1
              GOTO 848
  846           HF(R,1,4,3,T) = (COMP1 + COMP2) / 2
  848         CONTINUE
C
C  ****************************************************************
C  *   FOLLOWING STATEMENTS  USE  THE  SEASONAL  RELATIONSHIPS TO   *
C  *   ESTIMATE MISSING SEASONAL COUNTS FROM EXISTING COUNTS.       *
C  ****************************************************************
C
  850   WRITE (7,6100)
        DO 950 NL = 1,NTOT
          R = RM(NL)
          DO 880 D = 1,4
            IF (SF(NL,D,15).NE.-10.) GOTO 880
              DO 870 T = 1,14
                DENOM = 0
                DM1 = D - 1
                IF (D.EQ.1) DM1 = 1
                DO 855 N=1,DM1
                  IF (N.GE.D.OR.SF(NL,N,15).EQ.-10.) GOTO 855
```

```
                DENOM = DENOM + 1
              IF (SF(NL,N,T).LT.0.5) GOTO 855
                SF(NL,D,T) =(SF(NL,N,T)+1)* (1+HF(R,1,D,N,T))/
     *             (1-HF(R,1,D,N,T)) + SF(NL,D,T) - 1
  855         CONTINUE
            DP1 = D + 1
            IF (D.EQ.4) DP1 = 4
            DO 860 N=DP1,4
              IF (N.LE.D.OR.SF(NL,N,15).EQ.-10.) GOTO 860
                DENOM = DENOM + 1
              IF (SF(NL,N,T).LT.0.5) GOTO 860
                SF(NL,D,T) =(SF(NL,N,T)+1)* (1-HF(R,1,N,D,T))/
     *             (1+HF(R,1,N,D,T)) + SF(NL,D,T) - 1
  860         CONTINUE
            IF (SF(NL,D,T).LT.0.) SF(NL,D,T) = 0
            IF (DENOM.NE.0.) SF(NL,D,T) = SF(NL,D,T)/DENOM
  870       CONTINUE
  880     CONTINUE
C
C  ****************************************************************
C  *   IF THE ESTIMATED NUMBER OF COAL TRUCKS IS GREATER THAN THE   *
C  *   NUMBER OF  ESTIMATED TRUCKS  THAT ARE OF  A SIZE  TO CARRY   *
C  *   COAL THEN THE COAL TRUCK ESTIMATED IS REDUCED TO THE NUMBER  *
C  *   OF TRUCKS THAT CAN CARRY COAL. IF ANY PART OF  A COUNT  IS   *
C  *   ESTIMATED, THE PROGRAM MAKE THAT TOTAL NEGATIVE TO SIGNIFY   *
C  *   THIS.                                                        *
C  ****************************************************************
C
        SFTOT = 0.
        DO 885 S=1,4
          NUMTRK = SF(NL,S,6) + SF(NL,S,7) + SF(NL,S,8) +
     *             SF(NL,S,9) + SF(NL,S,10)
          IF (SF(NL,S,14).GT.FLOAT(NUMTRK)) SF(NL,S,14) = NUMTRK
          MINUS = 1
          IF (SF(NL,S,15).LT.0.) MINUS = -1
          SF(NL,S,15)=0
          IF (YR(NL).LT.80) SF(NL,S,14)=0
          DO 885 T=1,13
            SF(NL,S,15)=SF(NL,S,15)+SF(NL,S,T)*MINUS
C           WRITE(*,*)SF(1,S,T)
  885     CONTINUE
C
C  ****************************************************************
C  *   CALCULATE  TRUCK  INFORMATION NECESSARY FOR THE PRINTOUT     *
C  ****************************************************************
C
        SFTOT = INT((ABS(SF(NL,1,15))+ABS(SF(NL,2,15))+
     *           ABS(SF(NL,3,15))+ABS(SF(NL,4,15))+2.)/4.)
```

```
      DO 890 T=1,15
     SFY(T)=INT((SF(NL,1,T)+SF(NL,2,T)+SF(NL,3,T)+SF(NL,4,T)+2.)/4.)
890     CONTINUE
      IF (SFY(15).NE.SFTOT) SFY(15) = -SFTOT
      NUMTRK = SFY(4)+SFY(5)+SFY(6)+SFY(7)+SFY(8)+
     *         SFY(9)+SFY(10)+SFY(11)+SFY(12)+SFY(13)
      PERTRK = 0
      IF (SFTOT.NE.0.) PERTRK = (FLOAT(NUMTRK) / SFTOT) * 100
      SFY(14) = INT(SFY(14))
      PERCOL = 0
      IF (NUMTRK.NE.0) PERCOL = (SFY(14) / FLOAT(NUMTRK)) * 100
      AADT(NL) = IABS(AADT(NL))
C
C  ***************************************************************
C  *   PRINT OUT THE DATA  IN AN ORDERLY MANNER                  *
C  ***************************************************************
C
      IF(MOD(NL,3).EQ.1) CALL HEADER(YR(1))
      WRITE(8,1010)
      WRITE(8,1100) CN(NL),(SF(NL,1,T),T=1,15),NUMHRS(NL,1)
      WRITE(8,1110) SN(NL)
      WRITE(8,1120) ROUTE(NL),(SF(NL,2,T),T=1,15),NUMHRS(NL,2)
      WRITE(8,1130) MP(NL)
      WRITE(8,1140) FED(NL),FC(NL),(SF(NL,3,T),T=1,15),
     *              NUMHRS(NL,3)
      WRITE(8,1150) OWTW(NL)
      WRITE(8,1160) AADT(NL),(SF(NL,4,T),T=1,15),NUMHRS(NL,4)
      WRITE(8,1170) PERTRK
      WRITE(8,1180) PERCOL,(SFY(T),T=1,15)
      WRITE(8,1190)
      WRITE(8,1010)
      WRITE(8,1010)
      OPEN(20,FILE='FILE20')
      WRITE(20,1) CN(NL),SN(NL),ROUTE(NL),MP(NL),FC(NL),AADT(NL),
     *  PERTRK,PERCOL
1      FORMAT(I3,1X,A3,1X,A8,1X,F8.3,I2,1X,I6,1X,F8.3,F8.4)
C  ***************************************************************
C  *   IF ANY ERRORS ARE FOUND,  STATEMENTS 701 THRU 754 SEND THE  *
C  *   ERROR AND A MESSAGE BACK TO THE INITIATING TERMINAL.        *
C  ***************************************************************
C
      IF (CN(NL).GT.0.AND.CN(NL).LE.120) GOTO 891
        ERRCT = ERRCT + 1
        WRITE(7,6000) CN(NL),SN(NL),CN(NL),ROUTE(NL),MP(NL)
891     IF (SN(NL).NE.BLANK3) GOTO 893
        ERRCT = ERRCT + 1
        WRITE(7,6010) CN(NL),SN(NL),ROUTE(NL),MP(NL)
893     IF (MP(NL).GE.0.) GOTO 895
```

```
          ERRCT = ERRCT + 1
          WRITE(7,6030) CN(NL),SN(NL),MP(NL),ROUTE(NL)
 895      IF ((FED(NL).GT.0.AND.FED(NL).LE.4).OR.FED(NL).EQ.8) GOTO 896
          ERRCT = ERRCT + 1
          WRITE(7,6040) CN(NL),SN(NL),FED(NL),ROUTE(NL),MP(NL)
 896      IF (AADT(NL).NE.0) GOTO 897
          AADT(NL) = ABS(SFY(15))
          ERRCT = ERRCT + 1
 897      IF (ABS(AADT(NL)*2.).GE.ABS(SFY(15)).AND.
     *      ABS(AADT(NL)*.5).LE.ABS(SFY(15))) GOTO 898
          IF (AADT(NL).EQ.0) AADT(NL) = ABS(SFY(15))
          ERRCT = ERRCT + 1
          WRITE(7,6070) CN(NL),SN(NL),AADT(NL),ROUTE(NL),MP(NL)
 898      IF (PERTRK.LE.50.) GOTO 899
          ERRCT = ERRCT + 1
          WRITE(7,6080) CN(NL),SN(NL),NUMTRK,ROUTE(NL),MP(NL)
 899      IMP = MP(NL) * 1000
C         BACKSPACE 16
          OPEN(17,FILE=FN4,MODE='WRITE')
          WRITE (17,4200) CN(NL),SN(NL),ROUTE(NL),
     *         IMP,YR(NL),AADT(NL),FED(NL),FC(NL),CN(NL),SN(NL),FC(NL)
          IREC = 2
          MONTH = 1
          DAY = 0
          HOUR = 99
          REC = 2
          DO 900 T=1,15
            PRNT(T) = INT(SF(NL,1,T))
 900      CONTINUE
          WRITE (17,4300) IREC,MONTH,DAY,HOUR,(PRNT(T),T=1,15),
     *         CN(NL),SN(NL),FC(NL),REC
          MONTH = 4
          REC = 3
          DO 905 T=1,15
            PRNT(T) = INT(SF(NL,2,T))
 905      CONTINUE
          WRITE (17,4300) IREC,MONTH,DAY,HOUR,(PRNT(T),T=1,15),
     *         CN(NL),SN(NL),FC(NL),REC
          MONTH = 7
          REC = 4
          DO 910 T=1,15
            PRNT(T) = INT(SF(NL,3,T))
 910      CONTINUE
          WRITE (17,4300) IREC,MONTH,DAY,HOUR,(PRNT(T),T=1,15),
     *         CN(NL),SN(NL),FC(NL),REC
          MONTH = 10
          REC = 5
          DO 915 T=1,15
```

```
          PRNT(T) = INT(SF(NL,4,T))
 915      CONTINUE
          WRITE (17,4300) IREC,MONTH,DAY,HOUR,(PRNT(T),T=1,15),
     *          CN(NL),SN(NL),FC(NL),REC
          IREC = 3
          MONTH = 0
          REC = 6
          DO 920 T=1,15
            PRNT(T) = INT(SFY(T))
 920      CONTINUE
          WRITE (17,4300) IREC,MONTH,DAY,HOUR,(PRNT(T),T=1,15),
     *          CN(NL),SN(NL),FC(NL),REC
 950   CONTINUE
       WRITE(8,8000)
C
C  ****************************************************************
C  *   IF ANY  ERRORS CAME UP, PRINT A WARNING AT THE BOTTOM OF THE *
C  *   PRINTOUT.                                                  *
C  ****************************************************************
C
       IF (ERRCT.EQ.0) GOTO 1000
          WRITE(8,8100)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8300) ERRCT
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8400)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8200)
          WRITE(8,8100)
          WRITE(8,8100)
 1000  WRITE (7,7000) HRCT
       WRITE (7,7010) MONCT
       WRITE (*,*) '                ***** JOB COMPLETED ******'
```

```
      WRITE (*,7011)FN2,FN3
C     WRITE (*,*) '                OUTPUT FILES: FILE07  AND FILE08'
 7011     FORMAT ('                  OUTPUT FILES: ',A11,' & ',A11)
      CLOSE(8)
      CLOSE(7)
      CLOSE(16)
      CLOSE(17)
C
     OPEN(17,FILE=FN4,STATUS='OLD',MODE='READ')
     FN1='VCRYR'//YR1//'.EAL'
     OPEN(27,FILE=FN1)
     REWIND 17
  10 READ(17,2000,END=11)TEXT,IFC,NN
 2000 FORMAT(A84,I2,1X,I1)
     IF (NN.NE.1 .AND. NN.NE.6) GOTO 10
     WRITE(27,2000)TEXT,IFC,NN
     GOTO 10
  11 CLOSE(27)
     K=100
     DO 5 I=1,13
       K=K+1
       FN=XF(I)//'YR'//YR1
       OPEN(K,FILE=FN)
       IF (K.EQ.102) K=105
       IF (K.EQ.109) K=110
       IF (K.EQ.112) K=113
       IF (K.EQ.114) K=115
  5  CONTINUE
     REWIND 17
  12 READ(17,2000,END=99)TEXT,IFC,NN
     IF (NN.NE.1 .AND. NN.NE.6) GOTO 12
        IF (IFC.EQ.1) IN=101
        IF (IFC.EQ.2) IN=102
        IF (IFC.EQ.3) IN=103
        IF (IFC.EQ.4) IN=104
        IF (IFC.EQ.5) IN=105
        IF (IFC.EQ.6) IN=106
        IF (IFC.EQ.7) IN=107
        IF (IFC.EQ.8) IN=108
        IF (IFC.EQ.9) IN=109
        IF (IFC.EQ.10) IN=110
        IF (IFC.EQ.11) IN=111
        IF (IFC.EQ.12) IN=112
        IF (IFC.EQ.13) IN=113
        IF (IFC.EQ.14) IN=114
        IF (IFC.EQ.15) IN=115
        IF (IFC.EQ.16) IN=116
        IF (IFC.EQ.17) IN=117
```

```
      IF (IFC.EQ.18) IN=118
      IF (IFC.EQ.19) IN=119
      IF (IFC.LT.1 .AND. IFC.GT.19) GOTO 12
      WRITE(IN,2000)TEXT,IFC,NN
      GOTO 12
   99 K=100
     DO 15 I=1,19
        K=K+1
   13    CLOSE(K)
   15 CONTINUE
C  ****************************************************************
C  *   FOLLOWING ARE FORMAT STATEMENTS.                           *
C  ****************************************************************
C
1010   FORMAT(25X)
1100   FORMAT(' COUNTY       ',I3,1X,'WINTER ',
     *      F5.0,2F7.0,F4.0,F6.0,3F5.0,F6.0,5F5.0,F8.0,I7)
1110   FORMAT(' STATION      ',A3)
1120   FORMAT(' ROUTE    ',A8,' SPRING ',
     *      F5.0,2F7.0,F4.0,F6.0,3F5.0,F6.0,5F5.0,F8.0,I7)
1130   FORMAT(' MILE PT. ',F8.3)
1140   FORMAT(' FED AID ',I1,1X,'FUNC ',I2,1X,'SUMMER ',
     *      F5.0,2F7.0,F4.0,F6.0,3F5.0,F6.0,5F5.0,F8.0,I7)
1150   FORMAT(' DIRS COUNTED   ',I2)
1160   FORMAT(' AADT     ',I8,1X,'FALL   ',
     *      F5.0,2F7.0,F4.0,F6.0,3F5.0,F6.0,5F5.0,F8.0,I7)
1170   FORMAT(' % TRUCKS ',F8.1)
1180   FORMAT(' % TRK W/C',F8.1,1X,'ANNUAL ',
     *      F5.0,2F7.0,F4.0,F6.0,3F5.0,F6.0,5F5.0,F8.0)
1190   FORMAT('         ',9X,'AVERAGE')
4200   FORMAT('1CO',I3,'STA',A3,'RTE ',A8,'MP',I6,'YR',I2,'AADT',I6,
     *  'FED. AID',I1,'FUNC',I2,T77,I3,T81,A3,T85,I2,T88,'1')
4300   FORMAT(I1,3I2,I4,I6,I5,I4,I5,3I4,I5,5I4,I7,1X,I3,1X,A3,1X,I2,
     *      T88,I1)
6000   FORMAT(1X,I3,1X,A3,' 1 :(',I3,')   ',A9,' AT MILEPOINT ',
     *      F8.3,' COUNTY EXCEEDS LIMITS')
6010   FORMAT(1X,I3,4X,' 2 :(',A3,')   ',A9,' AT MILEPOINT ',
     *      F8.3,' STATION NUMBER BLANK')
6030   FORMAT(1X,I3,1X,A3,' 4 :(',F8.3,')',A9,
     *      ' MILE POINT NEGATIVE')
6040   FORMAT(1X,I3,1X,A3,' 5 :(',I1,')',A9,' AT MILEPOINT ',
     *      F8.3,' FEDERAL AID INCORRECT')
6070   FORMAT(1X,I3,1X,A3,' 7 :(',I8,')',A9,' AT MILEPOINT ',
     *      F8.3,' AADT EXCEEDS LIMITS')
6080   FORMAT(1X,I3,1X,A3,' 8 :(',I8,')',A9,' AT MILEPOINT ',
     *      F8.3,' TRUCKS OVER 50% OF VHCLS')
6090   FORMAT(1X,I3,1X,A3,' 7 :(',I8,')',A9,' AT MILEPOINT ',
     *      F8.3,' ZERO AADT         ')
```

```
6100   FORMAT(20X,'ERROR LISTING FOR CLASSIFICATION ESTIMATION')
7000   FORMAT(20X,I4,' PIECES OF DATA WITH UNUSABLE HOUR  NUMBERS')
7010   FORMAT(20X,I4,' PIECES OF DATA WITH UNUSABLE MONTH NUMBERS')
8000   FORMAT('1             ************************************',
     *'************************************************************')
8100   FORMAT('              ************************************',
     *'************************************************************')
8200   FORMAT('              ***',T114,'***')
8300   FORMAT('              ***',T45,'THERE HAVE BEEN ',I5,
     * ' WARNING(S) OR ERROR(S) PRODUCED',T114,'***')
8400   FORMAT('              ***',T45,'THE ERROR FILE HAS BEEN SENT',
     * ' TO THE INITIATING TERMINAL',T114,'***')
C
      STOP
      END
      SUBROUTINE HEADER(IYR)
C
C  ****************************************************************
C  *                                              *
C  *       THIS HEADER IS PRINTED AT THE TOP OF EACH PAGE.        *
C  *                                              *
C  ****************************************************************
C
        WRITE (8,1000) IYR
        WRITE (8,1010)
        WRITE (8,1010)
        WRITE (8,1010)
        WRITE (8,1010)
        WRITE (8,1020)
        WRITE (8,1030)
        WRITE (8,1040)
        WRITE (8,1050)
        WRITE (8,1060)
        WRITE (8,1070)
        WRITE (8,1080)
        WRITE (8,1090)
        WRITE (8,1010)
        WRITE (8,1010)
        WRITE (8,1010)
1000   FORMAT('1',42X,'DAILY VOLUMES BY VEHICLE TYPE FOR 19',I2)
1010   FORMAT(25X)
1020   FORMAT(25X,'+----+------+------+---+---------------+-------',
     *         '--------+--------------+----+-------+--------+')
1030   FORMAT(25X,'| MC |      |      | B |    SINGLE     | TRACTO',
     *         'R TRUCK |TRACTOR TRUCK |  T |       |  # OF  |')
1040   FORMAT(25X,'| OY |      |OTHER | U |     UNIT      |SINGLE ',
     *         'TRAILER |MULTI-TRAILER | CR |       |  HOURS |')
1050   FORMAT(25X,'| TC |PASNGR|2 AXLE| S +-----+----+----+----+--',
```

```
     *         '---+----+----+----+----+----+ OU | TOTAL |   OF   |')
  1060   FORMAT(25X,'| OL | CARS |4 TIRE| S |2AXLE| 3  |4 OR|4 OR|  ',
     *         '5  |6 OR|5 OR| 6  |7 OR| AC |       |  DATA  |')
  1070   FORMAT(25X,'| RE |      |VEHCLS| E |  6  |AXLE|MORE|LESS|AX',
     *         'LE |MORE|LESS|AXLE|MORE| LK |       |   PER  |')
  1080   FORMAT(25X,'|  S |      |      | S |TIRES|    |AXLE|AXLE|  ',
     *         '   |AXLE|AXLE|    |AXLE|  S |       | SEASON |')
  1090   FORMAT(25X,'+----+------+------+---+-----+----+----+----+--',
     *         '---+----+----+----+----+----+-------+--------+')
       RETURN
       END
       FUNCTION SGN(R)
C
C  ****************************************************************
C  *                                      *
C  *  SIGN ROUTINE                                *
C  *                                      *
C  *  INPUT  : R - A REAL NUMBER                      *
C  *                                      *
C  *  RETURNS: 1 - IF R IS POSITIVE                   *
C  *           0 - IF R IS EQUAL TO 0                   *
C  *          -1 - IF R IS NEGATIVE                     *
C  *                                      *
C  ****************************************************************
C
       REAL R
       SGN = 0
       IF (R.LT.0) SGN = -1
       IF (R.GT.0) SGN = 1
       RETURN
       END
```

B.2 WEIGHT DATA PROCESSING PROGRAM CODE

01_WGT_1LINE_TO_2LINES.SAS

THIS PROGRAM CONVERTS ONE LINE OF WEIGHT DATA WHEN THERE ARE MORE THAN FIVE AXLES TO TWO LINES WHICH IS COMPATIBLE WITH EXISTING PROGRAMS.

DATA YEAR AND FUNCTIONAL CLASSIFICATION CODES ARE ADDED

CHECKS ARE MADE FOR VALID TOTAL AXLE WEIGHTS AND TOTAL AXLE SPACINGS

VEHICLE TYPES BELOW TYPE 5 ARE EXCLUDED

```
OPTIONS MISSING=' '  LINESIZE=132;
DATA IN;
 INFILE 'P:\KYWIM11.WGT' MISSOVER;
 INPUT @1 A $123. @4 STAL $6.  @7 STA $3.  @10 DIR 1.  @11 LANE 1.  @14 MO 2.
      @16 DA 2.  @18 HR 2.  @20 VT 2.  @25 TWT 4.  @29 NAX 2.  @31 AWT 3.
      @34 ABS 3.  @37 BWT 3.  @40 BCS 3.  @43 CWT 3.  @46 CDS 3.  @49 DWT 3.
      @52 DES 3.  @55 EWT 3.  @58 EFS 3.  @61 FWT 3.  @64 FGS 3.  @67 GWT 3.
      @70 GHS 3.  @73 HWT 3.  @76 HIS 3.  @79 IWT 3.  @82 IJS 3.  @85 JWT 3.
      @88 JKS 3.  @91 KWT 3.  @94 KLS 3.  @97 LWT 3.  @100 LMS 3.  @103 MWT 3.
      @106 MNS 3.  @109 NWT 3.  @112 NOS 3.  @115 OWT 3.  @118 OUS 3.  @121 PWT 3.;

 TWB=SUM(ABS,BCS,CDS,DES,EFS,FGS,GHS,HIS,IJS,JKS,KLS,LMS,MNS,NOS,OUS);
 IF TWT="    " OR NAX="  " THEN DELETE;
 IF VT<5 THEN DELETE;


DATA OUT;
 SET IN;
 FILE 'P:\KYWIM11.7CD';
 IF CWT=' ' THEN CWT=0;
 ELSE IF DWT=' ' THEN DWT=0;
 ELSE IF EWT=' ' THEN EWT=0;
 ELSE IF FWT=' ' THEN FWT=0;
 ELSE IF GWT=' ' THEN GWT=0;
 ELSE IF HWT=' ' THEN HWT=0;
 ELSE IF IWT=' ' THEN IWT=0;
 ELSE IF JWT=' ' THEN JWT=0;
 ELSE IF KWT=' ' THEN KWT=0;
 ELSE IF LWT=' ' THEN LWT=0;
 ELSE IF MWT=' ' THEN MWT=0;
 ELSE IF NWT=' ' THEN NWT=0;
 ELSE IF OWT=' ' THEN OWT=0;
 ELSE IF PWT=' ' THEN PWT=0;


 IF EFS=' ' THEN DO;
 PUT @1 '721'  @4 FC Z2.  @6 STA $3.  @9 DIR 1.  @10 YR Z2.  @12 MO Z2.  @14 DA Z2.  @ 16 HR Z2.
    @22 VT Z2.  @24 LANE 1.  @42 TWT 4.  @46 AWT 3.  @49 BWT 3.  @52 CWT 3.  @55 DWT 3.  @58 EWT 3.
    @61 ABS 3.  @64 BCS 3.  @67 CDS 3.  @70 DES 3.  @73 TWB 4.  @80 "0";
 END;
```

```
 ELSE IF EFS^=' ' THEN DO;
 PUT @1 '721'  @4 FC Z2.  @6 STA $3.  @9 DIR 1.  @10 YR 2.  @12 MO Z2.  @14 DA Z2.  @ 16 HR Z2.
    @22 VT Z2.  @24 LANE 1.  @42 TWT 4.  @46 AWT 3.  @49 BWT 3.  @52 CWT 3.  @55 DWT 3.  @58 EWT 3.
    @61 ABS 3.  @64 BCS 3.  @67 CDS 3.  @70 DES 3.  @73 TWB 4.  @80 "1"  /
    @1 '721'  @4 FC Z2.  @6 STA $3.  @9 DIR 1.  @10 YR 2.  @12 MO Z2.  @14 DA Z2.  @ 16 HR Z2.
   @22 VT 2.  @24 LANE 1.  @29 FWT 3.  @32 GWT 3.  @35 HWT 3.  @38 IWT 3. @41 JWT 3.  @44 KWT 3.
    @47 LWT 3.  @50 MWT 3.  @53 EFS 3.  @56 FGS 3.  @59 GHS 3.  @62 HIS 3.  @65 IJS 3.
    @68 JKS 3.  @71 KLS 3.  @74 LMS 3.  @80 "9";
 END;

 ELSE IF MNS^=' ' THEN DO;
 PUT @1 '721'  @4 FC Z2.  @6 STA $3.  @9 DIR 1.  @10 YR 2.  @12 MO Z2.  @14 DA Z2.  @ 16 HR Z2.
    @22 VT 2.  @24 LANE 1.  @42 TWT 4.  @46 AWT 3.  @49 BWT 3.  @52 CWT 3.  @55 DWT 3.  @58 EWT 3.
    @61 ABS 3.  @64 BCS 3.  @67 CDS 3.  @70 DES 3.  @73 TWB 4.  @80 "1"  /
    @1 '721'  @4 FC Z2.  @6 STA $3.  @9 DIR 1.  @10 YR 2.  @12 MO Z2.  @14 DA Z2.  @ 16 HR Z2.
    @22 VT Z2.  @24 LANE Z1.  @29 FWT 3.  @32 GWT 3.  @35 HWT 3.  @38 IWT 3. @41 JWT 3.  @44 KWT 3.
    @47 LWT 3.  @50 MWT 3.  @53 EFS 3.  @56 FGS 3.  @59 GHS 3.  @62 HIS 3.  @65 IJS 3.
    @68 JKS 3.  @71 KLS 3.  @74 LMS 3.  @80 "2"  /
    @1 '721'  @4 FC Z2.  @6 STA $3.  @9 DIR 1.  @10 YR 2.  @12 MO Z2.  @14 DA Z2.  @ 16 HR Z2.
    @22 VT Z2.  @24 LANE 1.  @29 NWT 3.  @32 OWT 3.  @35 PWT 3.  @56 MNS 3.  @59 NOS 3.  @62 OUS 3.
@80 "9";
 END;


RUN;
```

02_METRIC_TO_ENGLISH.SAS

THIS PROGRAM CONVERTS AXLE WEIGHTS AND SPACINGS FROM METRIC TO ENGLISH FORMAT

```
OPTIONS MISSING=' '  LINESIZE=132;
DATA METCONV;
  INFILE 'P:\KYWIM11.7CD';

  INPUT @80 R 1. @;
  IF R=0 THEN DO;
    INPUT @1 A $CHAR34.  @42 TWT 4.  @46 AWT 3.  @49 BWT 3.  @52 CWT 3.  @55 DWT 3.  @58 EWT 3.
          @61 ABS 3.  @64 BCS 3.  @67 CDS 3.  @70 DES 3.  @73 TWB 4.;
  END;
  ELSE IF R=1 THEN
    INPUT @1 A $CHAR34.  @42 TWT 4.  @46 AWT 3.  @49 BWT 3.  @52 CWT 3.  @55 DWT 3.  @58 EWT 3.
          @61 ABS 3.  @64 BCS 3.  @67 CDS 3.  @70 DES 3.  @73 TWB 4.  /
          @1 B $CHAR34.  @29 FWT 3.  @32 GWT 3.  @35 HWT 3.  @38 IWT 3. @41 JWT 3.  @44 KWT 3.
          @47 LWT 3.  @50 MWT 3.  @53 EFS 3.  @56 FGS 3.  @59 GHS 3.  @62 HIS 3.  @65 IJS 3.
          @68 JKS 3.  @71 KLS 3.  @74 LMS 3.;

  TWT=TWT*2.204622279;  AWT=AWT*2.204622279;  BWT=BWT*2.204622279;
  CWT=CWT*2.204622279;  DWT=DWT*2.204622279;  EWT=EWT*2.204622279;
  FWT=FWT*2.204622279;  GWT=GWT*2.204622279;  HWT=HWT*2.204622279;
  IWT=IWT*2.204622279;  JWT=JWT*2.204622279;  KWT=KWT*2.204622279;
  LWT=LWT*2.204622279;  MWT=MWT*2.204622279;
  TWB=TWB*3.280839895013123;  ABS=ABS*3.280839895013123;  BCS=BCS*3.280839895013123;
  CDS=CDS*3.280839895013123;  DES=DES*3.280839895013123;  EFS=EFS*3.280839895013123;
  FGS=FGS*3.280839895013123;  GHS=GHS*3.280839895013123;  HIS=HIS*3.280839895013123;
  IJS=IJS*3.280839895013123;  JKS=JKS*3.280839895013123;  KLS=KLS*3.280839895013123;
  LMS=LMS*3.280839895013123;

  IF AWT>999 OR BWT>999 OR CWT>999 OR DWT>999 OR EWT>999 OR FWT>999 OR GWT>999 OR
HWT>999 OR IWT>999
     OR JWT>999 OR KWT>999 OR LWT>999 OR MWT>999 OR TWT>9999 THEN DELETE;
  IF ABS>999 OR BCS>999 OR CDS>999 OR DES>999 OR EFS>999 OR FGS>999 OR GHS>999 OR HIS>999 OR
IJS>999
     OR JKS>999 OR KLS>999 OR LMS>999 OR TWB>9999 THEN DELETE;

  FILE 'P:\KYWIM11_E.7CD';
  IF R=0 THEN DO;
    PUT @1 A $CHAR34.  @42 TWT 4.  @46 AWT 3.  @49 BWT 3.  @52 CWT 3.  @55 DWT 3.  @58 EWT 3.
        @61 ABS 3.  @64 BCS 3.  @67 CDS 3.  @70 DES 3.  @73 TWB 4.  @80 "0";
  END;
  ELSE IF R=1 THEN
    PUT @1 A $CHAR34.  @42 TWT 4.  @46 AWT 3.  @49 BWT 3.  @52 CWT 3.  @55 DWT 3.  @58 EWT 3.
        @61 ABS 3.  @64 BCS 3.  @67 CDS 3.  @70 DES 3.  @73 TWB 4.  @80 "1"  /
        @1 B $CHAR34.  @29 FWT 3.  @32 GWT 3.  @35 HWT 3.  @38 IWT 3. @41 JWT 3.  @44 KWT 3.
        @47 LWT 3.  @50 MWT 3.  @53 EFS 3.  @56 FGS 3.  @59 GHS 3.  @62 HIS 3.  @65 IJS 3.
        @68 JKS 3.  @71 KLS 3.  @74 LMS 3.  @80 "9";



RUN;
```

03_WIM_EDIT.FOR

THIS PROGRAM EDITS WEIGHT DATA AND OUTPUT AXLELOAD DISTRIBUTIONS FOR EACH AGGREGATE GROUP AND TRUCK TYPE

OBSERVATIONS WITH DATA ERRORS ARE EXCLUDED

```
C
C    THIS VERSION WRITES THE SAME WEIGHT VALUE TO THE OUTPUT FILE INSTEAD
C    OF CONVERTING IT TO A STATIC EQUIVALENT VALUE.   7/99
C    REF:
C
C   * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *WIM00010
C      SET "IERR" TO DESIRED CODE TO PRINT REJECTED RECORDS                WIM00020
C        1 = AXLES WITHOUT WEIGHT                                          WIM00030
C        2 = AXLE SPACINGS LESS THAN 1.5 FEET                              WIM00040
C        3 = MISMATCH BETWEEN NUMBERS OF AXLE WEIGHTS AND SPACINGS         WIM00050
C        4 = >10% DIFFERENCE BETWEEN GROSS WEIGHT AND SUM OF AXLE WEIGHT   WIM00060
C        5 = >10% DIFFERENCE BETWEEN WHEELBASE AND SUM OF AXLE SPACINGS    WIM00070
C        9 = ALL REJECTED RECORDS                                          WIM00080
C        0 = NO REJECTED RECORDS                                           WIM00090
C   * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *WIM00100
       BLOCK DATA                                                          WIM00110
       COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                          WIM00120
       COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS    WIM00130
       DATA (FAEQ(I),I=1,7)/3557.2423,3076.3085,2939.1856,4890.9914,       WIM00140
      1   2926.9924,38041.8808,4947.4293/                                  WIM00150
       DATA (FBEQ(I),I=1,7)/0.8932135,0.735836,0.720307,0.7463315,         WIM00160
      1   0.8693042,0.2044157,0.8634991/                                   WIM00170
       DATA (RAEQ(I),I=1,7)/3864.2167,2972.9051,3503.9253,4055.0787,       WIM00180
      1   1611.5815,37421.673,15454.4434/                                  WIM00190
       DATA (RBEQ(I),I=1,7)/0.8619752,0.7179301,0.7393744,0.8017253,       WIM00200
      1   0.9220579,0.2192902,0.6378718/                                   WIM00210
       DATA C5A,C5B,C6A,C6B/01.0549759,0.0432813,0.85158123,0.05135361/    WIM00220
       DATA C5NA,C5NB,C6NA,C6NB/0.0760435,0.0354375,0.7438038,0.0312612/   WIM00230
       DATA C5S,C6S,C5NS,C6NS/.4368124,1.3110553,.1334733,.9876060/        WIM00240
       END                                                                 WIM00250
       CHARACTER*64 FNAME,DATANM,FOUT2,A*80,AA*80                          WIM00260
       CHARACTER FIN*64,FOUT*64,FERR*64,FCHK*64
       COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT   WIM00270
       COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)            WIM00280
       COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV        WIM00290
       IERR=9                                                              WIM00300
C   * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *WIM00310
C      VARIABLE NAME DEFINITIONS                                           WIM00320
C   * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *WIM00330
C      ITYPE IS USED TO IDENTIFY PAVEMENT SURFACE MATERIAL, "1"=FLEXIBLE   WIM00340
C        AMD "2"=RIGID.                                                    WIM00350
C      FAEQ AND FBEQ ARE CONSTANTS OF STRAIGHT LINE EQUATIONS TO CONVERT   WIM00360
C        WIM AXLELOADS TO STATIC AXLELOADS FOR FLEXIBLE PAVEMENTS.         WIM00370
C      RAEQ AND RBEQ ARE CONSTANTS OF STRAIGHT LINE EQUATIONS TO CONVERT   WIM00380
C        WIM AXLELOADS TO STATIC AXLELOADS FOR RIGID PAVEMENTS.            WIM00390
C      IAXS(1),IAXS(2),...IAXS(70), ARE AXLE SPACINGS.                     WIM00400
C      IAW(1),IAW(2),..IAW(70), ARE DYNAMIC WIM AXLELOADS.                 WIM00410
```

```
C     ISW(1),ISW(2),..ISW(70), ARE CALCULATED STATIC AXLELOADS.          WIM00420
C     I=1 CORRESPONDS TO A STEERING AXLE.                                WIM00430
C     I=2 CORRESPONDS TO A DRIVE SINGLE AXLE.                            WIM00440
C     I=3 CORRESPONDS TO A TRAILER SINGLE AXLE.                          WIM00450
C     I=4 CORRESPONDS TO A DRIVE TANDEM AXLE GROUP IF AXLE SPACING IS    WIM00460
C       LESS THAN OR EQUAL TO 5.0 FEET.                                  WIM00470
C     I=5 CORRESPONDS TO A TRAILER TANDEM AXLE GROUP IF AXLE SPACING IS  WIM00480
C       LESS THAN OR EQUAL TO 5.0 FEET.                                  WIM00490
C     I=6 CORRESPONDS TO A DRIVE TRIDEM AXLE GROUP IF FIRST TO THIRD     WIM00500
C       AXLE-SPACING IS LESS THAN OR EQUAL TO 10.0 FEET.                 WIM00510
C     I=7 CORRESPONDS TO A TRAILER TRIDEM AXLE GROUP IF FIRST TO THIRD   WIM00520
C       AXLE-SPACING IS LESS THAN OR EQUAL TO 10.0 FEET.                 WIM00530
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *WIM00540
      WRITE(*,'(A)')' Specify Input File Name-'
      READ(*,'(A)')FIN
      OPEN(5,FILE=FIN,STATUS='OLD',BLANK='ZERO')
      WRITE(*,'(A)')' Specify Output File Name-'
      READ(*,'(A)')FOUT
      OPEN(6,FILE=FOUT,STATUS='UNKNOWN')
      OPEN(7,FILE='CHECK.LST',STATUS='UNKNOWN')
      OPEN(8,FILE='ERROR.LST',STATUS='UNKNOWN')
      ITYPE=1                                                            WIM00550
      IFORM=1                                                            WIM00560
      NERRORS0=0                                                         WIM00570
      NERRORS1=0                                                         WIM00580
      NERRORS2=0                                                         WIM00590
      NERRORS3=0                                                         WIM00600
      NERRORS4=0                                                         WIM00610
      NERRORS5=0                                                         WIM00620
      NGOOD=0                                                            WIM00630
      IAFC=0                                                             WIM00640
   22 CALL INIT                                                          WIM00650
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *WIM00660
C     READ DATA FROM WIM FILE                                            WIM00670
C  * *  * * * * * * * * * * * * * * * * * * * * * * * * *               WIM00680
      READ(5,4,END=1234) AA                                              WIM00690
    4 FORMAT(A80)                                                        WIM00700
      IF(AA(1:1).EQ.'2') THEN                                            WIM00710
         WRITE(8,4)AA                                                    WIM00720
C        WRITE(6,4)AA                                                    WIM00730
         GOTO 22                                                         WIM00740
      ELSE IF(IFORM.EQ.1) THEN                                           WIM00750
      READ(AA,1000)ITWRC,ISC,FC,SIN,IDT,IYR,IAMO,IDAY,IHOUR,             WIM00760
     1 (BD(J),J=1,9),(ICH(J),J=1,6),ICMOD,ITWT,(IAW(L),L=1,5),           WIM00770
     2 (IAXS(M),M=1,4),IAXTOT,IRSN,ICONTN(JCONT)                         WIM00780
 1000 FORMAT(T1,I1,T2,I2,T4,A2,T6,A3,T9,I1,T10,4I2,T18,9A1,T27,6I1,      WIM00790
     1 T36,I6,T42,I4,T46,5I3,T61,4I3,T73,I4,T77,I3,T80,I1)               WIM00800
C1000 FORMAT(I1,I2,A2,A3,I1,4I2,9A1,6I1,3X,I6,I4,9I3,I4,I3,I1)          WIM00810
      ELSE                                                               WIM00820
      READ(AA,1002)ITWRC,ISC,FC,SIN,IDT,IYR,IAMO,IDAY,IHOUR,             WIM00830
     1 (BD(J),J=1,9),(ICH(J),J=1,6),ICMOD,ITWT,(IAW(L),L=1,5),           WIM00840
     2 (IAXS(M),M=1,4),IAXTOT,IRSN,ICONTN(JCONT)                         WIM00850
 1002 FORMAT(I1,I2,A2,A3,I1,4I2,9A1,6I1,3X,I7,I4,9I3,I3,I3,I1)          WIM00860
      END IF                                                             WIM00870
```

```
      IAFC=IAFC+1                                                       WIM00880
      IF (IAFC.EQ.1) WRITE (8,2222) FC,IYR                              WIM00890
 2222 FORMAT (5X,'FUNCTIONAL CLASS= ',A2,5X,'YEAR= 19',I2)              WIM00900
C     IF(IAXS(2).EQ.0.AND.IAXS(1).LE.120) GOTO 22                       WIM00910
      IF(IAXS(2).EQ.0.AND.IAXS(1).LE.120) THEN                          WIM00920
          NERRORS0=NERRORS0+1                                           WIM00930
      IF (IERR.EQ.0.OR.IERR.EQ.9)                                       WIM00940
     *    WRITE(8,1110)ITWRC,ISC,FC,SIN,IDT,IYR,IAMO,IDAY,IHOUR,         WIM00950
     1    (BD(J),J=1,9),(ICH(J),J=1,6),ICMOD,ITWT,(IAW(L),L=1,5),        WIM00960
     2    (IAXS(M),M=1,4),IAXTOT,IRSN,ICONTN(JCONT)                      WIM00970
 1110 FORMAT(I1,I2,A2,A3,I1,4I2,9A1,6I1,3X,I6,I4,9I3,I4,I3,I1,2X,'E0')  WIM00980
          GO TO 22                                                      WIM00990
      ELSE                                                              WIM01000
      END IF                                                            WIM01010
   16 IF(ICONTN(JCONT).EQ.0.OR.ICONTN(JCONT).EQ.9) GOTO 19              WIM01020
      IF(ICONTN(JCONT).GE.1.AND.ICONTN(JCONT).LE.8) THEN                WIM01030
      LRT=JCONT*8-2                                                     WIM01040
      LOP=JCONT*8+5                                                     WIM01050
      MRT=JCONT*8-3                                                     WIM01060
      MOP=JCONT*8+4                                                     WIM01070
      JCONT=JCONT+1                                                     WIM01080
   15 READ(5,1001)ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,(BD(J),J=1,6),   WIM01090
     1 (IAW(L),L=LRT,LOP),(IAXS(M),M=MRT,MOP),IRSN,ICONTN(JCONT)        WIM01100
 1001 FORMAT(I1,I2,A2,A3,1X,4I2,6A1,5X,17I3,I1)                         WIM01110
      IF(ICONTN(JCONT).GE.1.AND.ICONTN(JCONT).LE.8) GOTO 16             WIM01120
      ENDIF                                                             WIM01130
   19 CONTINUE                                                          WIM01140
C     IAFC=IAFC+1                                                       WIM01150
C     IF (IAFC.EQ.1) WRITE (8,2222) FC,IYR                              WIM01160
C2222 FORMAT (5X,'FUNCTIONAL CLASS= ',A2,5X,'YEAR= 19',I2)              WIM01170
C     ************************************************************      WIM01180
C     THE FOLLOWING SECTION PERFORMS EDIT CHECKS ON THE WEIGHT DATA     WIM01190
C     AND REPORTS THE RESULTS.                          1/93            WIM01200
C     ************************************************************      WIM01210
C                                                                       WIM01220
C     NERRORS0 = NO. OF ERRORS:  TWO-AXLE VEHICLE LESS THAN 12 FEET     WIM01230
C     NERRORS1 = NO. OF ERRORS:  AXLE WITHOUT WEIGHT                    WIM01240
C     NERRORS2 = NO. OF ERRORS:  AXLE SPACING LESS THAN 1.5 FEET        WIM01250
C     NERRORS3 = NO. OF ERRORS:  MISMATCH BETWEEN NUMBERS OF AXLE       WIM01260
C       WEIGHTS AND SPACINGS                                            WIM01270
C     NERRORS4 = NO. OF ERRORS:  10+ PERCENT DIFFERENCE BETWEEN GROSS   WIM01280
C       WEIGHT AND SUM OF AXLE WEIGHTS                                  WIM01290
C     NERRORS5 = NO. OF ERRORS:  10+ PERCENT DIFFERENCE BETWEEN         WIM01300
C       WHEELBASE AND SUM OF AXLE SPACINGS                              WIM01310
C                                                                       WIM01320
      SUMW=0.0                                                          WIM01330
      SUMS=0.0                                                          WIM01340
      NA=0                                                              WIM01350
      NB=0                                                              WIM01360
      IWMAX=0                                                           WIM01370
      ISMAX=0                                                           WIM01380
      DO 11 I=1,15                                                      WIM01390
      IF (IAW(I).GT.0) NA=NA+1                                          WIM01400
      IF (IAW(I).GT.0) IWMAX=I                                          WIM01410
```

```
      SUMW=SUMW+IAW(I)                                                  WIM01420
      IF (IAXS(I).GT.0) NB=NB+1                                         WIM01430
      IF (IAXS(I).GT.0) ISMAX=I                                         WIM01440
      SUMS=SUMS+IAXS(I)                                                 WIM01450
   11 CONTINUE                                                          WIM01460
      DO 121 I=1,IWMAX                                                  WIM01470
      IF (IAW(I).EQ.0) THEN                                             WIM01480
          NERRORS1=NERRORS1+1                                           WIM01490
      IF (IERR.EQ.1.OR.IERR.EQ.9)                                       WIM01500
     *    WRITE(8,1111)ITWRC,ISC,FC,SIN,IDT,IYR,IAMO,IDAY,IHOUR,        WIM01510
     1    (BD(J),J=1,9),(ICH(J),J=1,6),ICMOD,ITWT,(IAW(L),L=1,5),       WIM01520
     2    (IAXS(M),M=1,4),IAXTOT,IRSN,ICONTN(JCONT)                     WIM01530
 1111 FORMAT(I1,I2,A2,A3,I1,4I2,9A1,6I1,3X,I6,I4,9I3,I4,I3,I1,2X,'E1')  WIM01540
          GO TO 22                                                      WIM01550
      ELSE                                                              WIM01560
      END IF                                                            WIM01570
  121 CONTINUE                                                          WIM01580
      DO 141 I=1,ISMAX                                                  WIM01590
      IF (IAXS(I).LT.15) THEN                                           WIM01600
          NERRORS2=NERRORS2+1                                           WIM01610
      IF (IERR.EQ.2.OR.IERR.EQ.9)                                       WIM01620
     *    WRITE(8,1112)ITWRC,ISC,FC,SIN,IDT,IYR,IAMO,IDAY,IHOUR,        WIM01630
     1    (BD(J),J=1,9),(ICH(J),J=1,6),ICMOD,ITWT,(IAW(L),L=1,5),       WIM01640
     2    (IAXS(M),M=1,4),IAXTOT,IRSN,ICONTN(JCONT)                     WIM01650
 1112 FORMAT(I1,I2,A2,A3,I1,4I2,9A1,6I1,3X,I6,I4,9I3,I4,I3,I1,2X,'E2')  WIM01660
          GO TO 22                                                      WIM01670
      ELSE                                                              WIM01680
      END IF                                                            WIM01690
  141 CONTINUE                                                          WIM01700
      IF (NA.NE.(NB+1)) THEN                                            WIM01710
          NERRORS3=NERRORS3+1                                           WIM01720
      IF (IERR.EQ.3.OR.IERR.EQ.9)                                       WIM01730
     *    WRITE(8,1113)ITWRC,ISC,FC,SIN,IDT,IYR,IAMO,IDAY,IHOUR,        WIM01740
     1    (BD(J),J=1,9),(ICH(J),J=1,6),ICMOD,ITWT,(IAW(L),L=1,5),       WIM01750
     2    (IAXS(M),M=1,4),IAXTOT,IRSN,ICONTN(JCONT)                     WIM01760
 1113 FORMAT(I1,I2,A2,A3,I1,4I2,9A1,6I1,3X,I6,I4,9I3,I4,I3,I1,2X,'E3')  WIM01770
          GO TO 22                                                      WIM01780
      ELSE                                                              WIM01790
      END IF                                                            WIM01800
      TWT=ITWT                                                          WIM01810
      AXTOT=IAXTOT                                                      WIM01820
      IF (SUMW.LT.(TWT*.9).OR.SUMW.GT.(TWT*1.1)) THEN                   WIM01830
          NERRORS4=NERRORS4+1                                           WIM01840
      IF (IERR.EQ.4.OR.IERR.EQ.9)                                       WIM01850
     *    WRITE(8,1114)ITWRC,ISC,FC,SIN,IDT,IYR,IAMO,IDAY,IHOUR,        WIM01860
     1    (BD(J),J=1,9),(ICH(J),J=1,6),ICMOD,ITWT,(IAW(L),L=1,5),       WIM01870
     2    (IAXS(M),M=1,4),IAXTOT,IRSN,ICONTN(JCONT)                     WIM01880
 1114 FORMAT(I1,I2,A2,A3,I1,4I2,9A1,6I1,3X,I6,I4,9I3,I4,I3,I1,2X,'E4')  WIM01890
          GO TO 22                                                      WIM01900
      ELSE                                                              WIM01910
      END IF                                                            WIM01920
      IF (SUMS.LT.(AXTOT*.9).OR.SUMS.GT.(AXTOT*1.1)) THEN               WIM01930
          NERRORS5=NERRORS5+1                                           WIM01940
      IF (IERR.EQ.5.OR.IERR.EQ.9)                                       WIM01950
```

```
     *    WRITE(8,1115)ITWRC,ISC,FC,SIN,IDT,IYR,IAMO,IDAY,IHOUR,        WIM01960
     1    (BD(J),J=1,9),(ICH(J),J=1,6),ICMOD,ITWT,(IAW(L),L=1,5),       WIM01970
     2    (IAXS(M),M=1,4),IAXTOT,IRSN,ICONTN(JCONT)                     WIM01980
 1115 FORMAT(I1,I2,A2,A3,I1,4I2,9A1,6I1,3X,I6,I4,9I3,I4,I3,I1,2X,'E5')  WIM01990
         GO TO 22                                                       WIM02000
      ELSE                                                              WIM02010
      END IF                                                            WIM02020
      NGOOD=NGOOD+1                                                     WIM02030
C     ************************************************************      WIM02040
C     END OF DATA EDIT SECTION                                          WIM02050
C     ************************************************************      WIM02060
      IF(IAXS(1).GT.55) GOTO 245                                        WIM02070
C     ************************************************************      WIM02080
C     THE FOLLOWING SECTION TO 245 APPLIES TO TRUCKS HAVING 2 STEERING  WIM02090
C     AXLES (EX: CRANES, DRILL RIGS, TRANSIT MIXERS, EUROPEAN DUMPS)    WIM02100
C     ************************************************************      WIM02110
      IF(IAXS(1).LE.55.AND.IAXS(2).LE.55.AND.IAXS(3).GT.55) GOTO 241    WIM02120
      IF(IAXS(1).LE.55.AND.IAXS(2).GT.55) GOTO 243                      WIM02130
  241 DO 242 J=4,13                                                     WIM02140
      IF(IAXS(J).EQ.0) GOTO 244                                         WIM02150
  242 CONTINUE                                                          WIM02160
  244 ICT(1)=2                                                          WIM02170
      ICT(2)=J                                                          WIM02180
      CALL TRTRI(1,2,3)                                                 WIM02190
      IF(J.EQ.5) CALL DRTAN(4,5)                                        WIM02200
      IF(J.EQ.6) CALL DRTRI(4,5,6)                                      WIM02210
      IF(J.EQ.7) CALL DQUAD(4,5,6,7)                                    WIM02220
      GOTO 100                                                          WIM02230
  243 DO 246 J=3,13                                                     WIM02240
      IF(IAXS(J).EQ.0) GOTO 247                                         WIM02250
  246 CONTINUE                                                          WIM02260
  247 ICT(1)=2                                                          WIM02270
      ICT(2)=J                                                          WIM02280
      CALL TRTAN(1,2)                                                   WIM02290
      IF(J.EQ.4) CALL DRTAN(3,4)                                        WIM02300
      IF(J.EQ.5) CALL DRTRI(3,4,5)                                      WIM02310
      IF(J.EQ.6) CALL DQUAD(3,4,5,6)                                    WIM02320
      GOTO 100                                                          WIM02330
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *      WIM02340
C *   CHECK AXLE SPACING TO DETERMINE AXLE CONFIGURATION  *             WIM02350
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *      WIM02360
  245 AS=0                                                              WIM02370
      DO 24 I=1,13                                                      WIM02380
      IF (IAXS(I).EQ.0) GOTO 25                                         WIM02390
      AS=AS+0.1*IAXS(I)                                                 WIM02400
      AI=I                                                              WIM02410
      AV=AS/AI                                                          WIM02420
   24 CONTINUE                                                          WIM02430
   25 J=I-1                                                             WIM02440
      IF (J.EQ.0) GOTO 28                                               WIM02450
      ACK=12.5*AV                                                       WIM02460
      ICK=ACK                                                           WIM02470
      DO 29 K=1,J                                                       WIM02480
      IF (IAXS(K).GT.ICK) GOTO 28                                       WIM02490
```

```
   29 CONTINUE                                                          WIM02500
      IF(IAXS(2).EQ.0) GOTO 28                                          WIM02510
      ICT(1)=9                                                          WIM02520
      ICT(2)=2                                                          WIM02530
      CALL STEER                                                        WIM02540
      CALL DRSING                                                       WIM02550
      DO 26 L=3,I                                                       WIM02560
      M=L                                                               WIM02570
      ICT(M)=1                                                          WIM02580
      CALL TRSING(M)                                                    WIM02590
   26 CONTINUE                                                          WIM02600
      IF (I.LE.4) THEN                                                  WIM02610
      ICT(1)=3                                                          WIM02620
      ICT(4)=0                                                          WIM02630
      ELSE                                                              WIM02640
      ICT(1)=5                                                          WIM02650
      END IF                                                            WIM02660
      IF (I.EQ.3.OR.I.EQ.5) THEN                                        WIM02670
      ICT(3)=1                                                          WIM02680
      ELSE                                                              WIM02690
      ICT(3)=2                                                          WIM02700
      END IF                                                            WIM02710
      IF (I.GE.5) THEN                                                  WIM02720
      ICT(4)=2                                                          WIM02730
      END IF                                                            WIM02740
      GOTO 100                                                          WIM02750
   28 IF(IAXS(2).EQ.0) GOTO 21                                          WIM02760
      IF(IAXS(3).EQ.0) GOTO 31                                          WIM02770
      IF(IAXS(4).EQ.0) GOTO 41                                          WIM02780
      IF(IAXS(5).EQ.0) GOTO 51                                          WIM02790
      IF(IAXS(6).EQ.0) GOTO 61                                          WIM02800
      IF(IAXS(7).EQ.0) GOTO 71                                          WIM02810
      IF(IAXS(8).EQ.0) GOTO 81                                          WIM02820
      GOTO 91                                                           WIM02830
C  * * * * * * * 2-AXLE TRUCK * * *                                     WIM02840
   21 CALL STEER                                                        WIM02850
      CALL DRSING                                                       WIM02860
      IWT=ISW(1)+ISW(2)                                                 WIM02870
      IF(IAXS(1).GE.200) THEN                                           WIM02880
      ICT(1)=1                                                          WIM02890
      ICT(2)=9                                                          WIM02900
      GOTO 100                                                          WIM02910
      ELSE IF(IAXS(2).EQ.0.AND.IAXS(1).GE.88.AND.IAXS(1).LE.145.AND.    WIM02920
     1   IWT.GT.30.AND.IWT.LE.66) THEN                                  WIM02930
      ICT(1)=2                                                          WIM02940
      ICT(2)=0                                                          WIM02950
      GOTO 100                                                          WIM02960
      ELSE IF(IAXS(2).EQ.0.AND.IAXS(1).LT.115.AND.IWT.GT.66.AND.        WIM02970
     1    IWT.LE.90) THEN                                               WIM02980
      ICT(1)=2                                                          WIM02990
      ICT(2)=1                                                          WIM03000
      GOTO 100                                                          WIM03010
      ELSE IF(IWT.GT.90) THEN                                           WIM03020
      ICT(1)=2                                                          WIM03030
```

```
      ICT(2)=2                                                          WIM03040
      GOTO 100                                                          WIM03050
      ELSE                                                              WIM03060
      ICT(1)=2                                                          WIM03070
      ICT(2)=2                                                          WIM03080
      GOTO 100                                                          WIM03090
      END IF                                                            WIM03100
C  * * * * * * * 3-AXLE TRUCK * * *                                    WIM03110
   31 CALL STEER                                                        WIM03120
      IF(IAXS(1).GE.190.AND.IAXS(2).LE.50) THEN                         WIM03130
      ICT(1)=1                                                          WIM03140
      ICT(2)=9                                                          WIM03150
      CALL STEER                                                        WIM03160
      CALL DRTAN(2,3)                                                   WIM03170
      GOTO 100                                                          WIM03180
      ELSE                                                              WIM03190
      END IF                                                            WIM03200
      IF(IAXS(2).LE.50) GOTO 33                                         WIM03210
      ICT(1)=3                                                          WIM03220
      ICT(2)=2                                                          WIM03230
      ICT(3)=1                                                          WIM03240
      CALL DRSING                                                       WIM03250
      CALL TRSING(3)                                                    WIM03260
      GOTO 100                                                          WIM03270
   33 CALL DRTAN(2,3)                                                   WIM03280
      ICT(1)=2                                                          WIM03290
      ICT(2)=3                                                          WIM03300
      GOTO 100                                                          WIM03310
C  * * * * * * * 4-AXLE TRUCK * * *                                    WIM03320
   41 CALL STEER                                                        WIM03330
      IF((IAXS(2)+IAXS(3)).LE.100) GOTO 43                              WIM03340
      IF(IAXS(2).LE.55) GOTO 42                                         WIM03350
      CALL DRSING                                                       WIM03360
      ICT(1)=3                                                          WIM03370
      ICT(2)=2                                                          WIM03380
      IF(IAXS(3).LE.50) THEN                                            WIM03390
            CALL TRTAN(3,4)                                             WIM03400
            ICT(3)=2                                                    WIM03410
            GOTO 100                                                    WIM03420
         ELSE                                                           WIM03430
            CALL TRSING(3)                                              WIM03440
            CALL TRSING(4)                                              WIM03450
            IF (IAXS(3).LE.80) THEN                                     WIM03460
               ICT(3)=2                                                 WIM03470
               ELSE                                                     WIM03480
               ICT(3)=7                                                 WIM03490
            END IF                                                      WIM03500
        GOTO 100                                                        WIM03510
      END IF                                                            WIM03520
   42 CALL DRTAN(2,3)                                                   WIM03530
      CALL TRSING(4)                                                    WIM03540
      ICT(1)=3                                                          WIM03550
      ICT(2)=3                                                          WIM03560
      ICT(3)=1                                                          WIM03570
```

```
      GOTO 100                                                          WIM03580
   43 CALL DRTRI(2,3,4)                                                 WIM03590
      ICT(1)=2                                                          WIM03600
      ICT(2)=4                                                          WIM03610
      GOTO 100                                                          WIM03620
C  * * * * * * * 5-AXLE TRUCK * * *                                    WIM03630
   51 CALL STEER                                                        WIM03640
      IAX23=IAXS(2)+IAXS(3)                                             WIM03650
      IAX24=IAXS(2)+IAXS(3)+IAXS(4)                                     WIM03660
      IAX34=IAXS(3)+IAXS(4)                                             WIM03670
      IF(IAX24.LE.150) GOTO 52                                          WIM03680
      IF(IAX23.LE.100) GOTO 58                                          WIM03690
      IF(IAXS(2).LE.55.AND.IAXS(4).LE.55) GOTO 53                       WIM03700
      IF(IAXS(2).LE.55.AND.IAXS(4).GT.55) GOTO 55                       WIM03710
      IF(IAXS(2).GT.55.AND.IAX34.LE.100) GOTO 56                        WIM03720
C     IF ABOVE 5 IF STATEMENTS ARE FALSE, THEN HAVE 5-AXLE DOUBLE BOTTOMWIM03730
      ICT(1)=5                                                          WIM03740
      ICT(2)=2                                                          WIM03750
      ICT(3)=1                                                          WIM03760
      ICT(4)=2                                                          WIM03770
      CALL DRSING                                                       WIM03780
      CALL TRSING(3)                                                    WIM03790
      CALL TRSING(4)                                                    WIM03800
      CALL TRSING(5)                                                    WIM03810
      GOTO 100                                                          WIM03820
   52 CALL DQUAD(2,3,4,5)                                               WIM03830
      ICT(1)=2                                                          WIM03840
      ICT(2)=5                                                          WIM03850
      GOTO 100                                                          WIM03860
   53 CALL DRTAN(2,3)                                                   WIM03870
      CALL TRTAN(4,5)                                                   WIM03880
      ICT(1)=3                                                          WIM03890
      ICT(2)=3                                                          WIM03900
      ICT(3)=2                                                          WIM03910
      GOTO 100                                                          WIM03920
   55 CALL DRTAN(2,3)                                                   WIM03930
      CALL TRSING(4)                                                    WIM03940
      CALL TRSING(5)                                                    WIM03950
      ICT(1)=3                                                          WIM03960
      ICT(2)=3                                                          WIM03970
            IF (IAXS(4).LE.80) THEN                                     WIM03980
               ICT(3)=2                                                 WIM03990
               ELSE                                                     WIM04000
               ICT(3)=7                                                 WIM04010
      GOTO 100                                                          WIM04020
      END IF                                                            WIM04030
   56 CALL DRSING                                                       WIM04040
      CALL TRTRI(3,4,5)                                                 WIM04050
      ICT(1)=3                                                          WIM04060
      ICT(2)=2                                                          WIM04070
      ICT(3)=3                                                          WIM04080
      GOTO 100                                                          WIM04090
   58 CALL DRTRI(2,3,4)                                                 WIM04100
      CALL TRSING(5)                                                    WIM04110
```

```
      ICT(1)=3                                                          WIM04120
      ICT(2)=4                                                          WIM04130
      ICT(3)=1                                                          WIM04140
      GOTO 100                                                          WIM04150
C  * * * * * * * 6-AXLE TRUCK * * *                                    WIM04160
   61 CALL STEER                                                        WIM04170
      IAXTR=IAXS(4)+IAXS(5)                                             WIM04180
      IAXD=IAXS(2)+IAXS(3)+IAXS(4)                                      WIM04190
      IF(IAXS(2).GT.55) GOTO 66                                         WIM04200
      IF(IAXD.LE.100.AND.IAXS(5).LE.55) GOTO 64                         WIM04210
      IF(IAXD.LE.100.AND.IAXS(5).GT.55.AND.IAXS(5).LT.120) GOTO 65      WIM04220
      IF(IAXS(2).LE.55.AND.IAXTR.LE.100) GOTO 62                        WIM04230
      IF(IAXS(2).LE.50.AND.IAXS(3).GT.50) GOTO 63                       WIM04240
C     IF ALL OF ABOVE ARE FALSE, THEN HAVE 5-AXLE TRACTOR + TRL. SNGL.  WIM04250
      CALL TRSING(6)                                                    WIM04260
      CALL DQUAD(2,3,4,5)                                               WIM04270
      ICT(1)=3                                                          WIM04280
      ICT(2)=5                                                          WIM04290
      ICT(3)=1                                                          WIM04300
      GOTO 100                                                          WIM04310
   62 CALL DRTAN(2,3)                                                   WIM04320
      CALL TRTRI(4,5,6)                                                 WIM04330
      ICT(1)=3                                                          WIM04340
      ICT(2)=3                                                          WIM04350
      ICT(3)=3                                                          WIM04360
      GOTO 100                                                          WIM04370
   63 CALL DRTAN(2,3)                                                   WIM04380
      CALL TRSING(4)                                                    WIM04390
      CALL TRSING(5)                                                    WIM04400
      CALL TRSING(6)                                                    WIM04410
      ICT(1)=5                                                          WIM04420
      ICT(2)=3                                                          WIM04430
      ICT(3)=1                                                          WIM04440
      ICT(4)=2                                                          WIM04450
      GOTO 100                                                          WIM04460
   64 CALL DRTRI(2,3,4)                                                 WIM04470
      CALL TRTAN(5,6)                                                   WIM04480
      ICT(1)=3                                                          WIM04490
      ICT(2)=4                                                          WIM04500
      ICT(3)=2                                                          WIM04510
      GOTO 100                                                          WIM04520
   65 CALL DRTRI(2,3,4)                                                 WIM04530
      CALL TRSING(5)                                                    WIM04540
      CALL TRSING(6)                                                    WIM04550
      ICT(1)=3                                                          WIM04560
      ICT(2)=4                                                          WIM04570
      ICT(3)=7                                                          WIM04580
      GOTO 100                                                          WIM04590
   66 CALL DRSING                                                       WIM04600
      IF(IAXS(4).GT.55.AND.IAXS(4).LT.100) THEN                         WIM04610
        CALL TRTAN(3,4)                                                 WIM04620
        CALL TRSING(5)                                                  WIM04630
        CALL TRSING(6)                                                  WIM04640
        ICT(1)=5                                                        WIM04650
```

```
        ICT(2)=2                                                        WIM04660
        ICT(3)=2                                                        WIM04670
        ICT(4)=2                                                        WIM04680
        GOTO 100                                                        WIM04690
      ELSE                                                              WIM04700
        CALL TQUAD(3,4,5,6)                                             WIM04710
        ICT(1)=3                                                        WIM04720
        ICT(2)=2                                                        WIM04730
        ICT(3)=4                                                        WIM04740
        GOTO 100                                                        WIM04750
      END IF                                                            WIM04760
C  * * * * * * * 7-AXLE TRUCK * * *                                     WIM04770
   71 CALL STEER                                                        WIM04780
      IAXD23=IAXS(2)+IAXS(3)                                            WIM04790
      IAXD24=IAXS(2)+IAXS(3)+IAXS(4)                                    WIM04800
      IAXD26=IAXS(2)+IAXS(3)+IAXS(4)+IAXS(5)+IAXS(6)                    WIM04810
      IAXT46=IAXS(4)+IAXS(5)+IAXS(6)                                    WIM04820
      IAXT56=IAXS(5)+IAXS(6)                                            WIM04830
      IF(IAXS(2).GT.55.AND.IAXD26.LE.250) THEN                          WIM04840
        CALL SEXTET(2,3,4,5,6,7)                                        WIM04850
        ICT(1)=2                                                        WIM04860
        ICT(2)=7                                                        WIM04870
        GOTO 100                                                        WIM04880
      ELSE IF(IAXS(2).GT.55.AND.IAXS(3).GT.50.AND.IAXS(4).GT.50.AND.    WIM04890
     1  IAXS(5).GT.50.AND.IAXS(6).GT.50) THEN                          WIM04900
        CALL DRSING                                                     WIM04910
        CALL TRSING(3)                                                  WIM04920
        CALL TRSING(4)                                                  WIM04930
        CALL TRSING(5)                                                  WIM04940
        CALL TRSING(6)                                                  WIM04950
        CALL TRSING(7)                                                  WIM04960
        ICT(1)=7                                                        WIM04970
        ICT(2)=2                                                        WIM04980
        ICT(3)=1                                                        WIM04990
        ICT(4)=2                                                        WIM05000
        ICT(5)=2                                                        WIM05010
        GOTO 100                                                        WIM05020
      ELSE IF(IAXS(2).GT.55.AND.IAXS(3).LE.50.AND.IAXS(4).GT.50.AND.    WIM05030
     1  IAXS(5).GT.50.AND.IAXS(6).GT.50) THEN                          WIM05040
        CALL TRTAN(3,4)                                                 WIM05050
        CALL TRSING(5)                                                  WIM05060
        CALL TRSING(6)                                                  WIM05070
        CALL TRSING(7)                                                  WIM05080
        ICT(1)=7                                                        WIM05090
        ICT(2)=2                                                        WIM05100
        ICT(3)=2                                                        WIM05110
        ICT(4)=2                                                        WIM05120
        GOTO 100                                                        WIM05130
      ELSE IF(IAXS(2).GT.55.AND.IAXS(3).GT.50.AND.IAXS(4).GT.50.AND.    WIM05140
     1  IAXS(5).GT.50.AND.IAXS(6).LE.50) THEN                          WIM05150
        CALL TRSING(3)                                                  WIM05160
        CALL TRSING(4)                                                  WIM05170
        CALL TRSING(5)                                                  WIM05180
        CALL TRTAN(6,7)                                                 WIM05190
```

```
      ICT(1)=7                                                          WIM05200
      ICT(2)=2                                                          WIM05210
      ICT(3)=1                                                          WIM05220
      ICT(4)=2                                                          WIM05230
      ICT(5)=2                                                          WIM05240
      GOTO 100                                                          WIM05250
    ELSE                                                                WIM05260
    END IF                                                              WIM05270
    IF(IAXS(2).LE.55.AND.IAXS(3).GT.50) GOTO 72                         WIM05280
    IF(IAXD23.LE.100.AND.IAXS(4).GT.50) GOTO 73                         WIM05290
    IF(IAXD24.LE.150) GOTO 74                                           WIM05300
 72 CALL DRTAN(2,3)                                                     WIM05310
      ICT(1)=3                                                          WIM05320
      ICT(2)=3                                                          WIM05330
    IF(IAXT46.LE.150) GOTO 77                                           WIM05340
    IF(IAXT46.GT.150.AND.IAXT56.LT.100) THEN                            WIM05350
      CALL TRTRI(5,6,7)                                                 WIM05360
      CALL TRSING(4)                                                    WIM05370
      ICT(3)=9                                                          WIM05380
      GOTO 100                                                          WIM05390
    ELSE IF (IAXT46.GT.150.AND.IAXS(6).LE.80) THEN                      WIM05400
      CALL TRTRI(4,5,6)                                                 WIM05410
      CALL TRSING(7)                                                    WIM05420
      ICT(3)=9                                                          WIM05430
      GOTO 100                                                          WIM05440
    ELSE IF(IAXT46.GT.150.AND.IAXS(6).GT.80) THEN                       WIM05450
        CALL TRTRI(4,5,6)                                               WIM05460
        CALL TRSING(7)                                                  WIM05470
        ICT(1)=5                                                        WIM05480
        ICT(2)=3                                                        WIM05490
        ICT(3)=3                                                        WIM05500
        ICT(4)=1                                                        WIM05510
        GOTO 100                                                        WIM05520
    ELSE                                                                WIM05530
        CALL TRTAN(4,5)                                                 WIM05540
        CALL TRSING(6)                                                  WIM05550
        CALL TRSING(7)                                                  WIM05560
        ICT(1)=5                                                        WIM05570
        ICT(2)=3                                                        WIM05580
        ICT(3)=2                                                        WIM05590
        ICT(4)=2                                                        WIM05600
    GOTO 100                                                            WIM05610
    ENDIF                                                               WIM05620
 77 CALL TQUAD(4,5,6,7)                                                 WIM05630
    ICT(1)=3                                                            WIM05640
    ICT(2)=3                                                            WIM05650
    ICT(3)=4                                                            WIM05660
    GOTO 100                                                            WIM05670
 73 CALL DRTRI(2,3,4)                                                   WIM05680
    IF(IAXT56.GT.100) GOTO 731                                          WIM05690
    CALL TRTRI(5,6,7)                                                   WIM05700
    ICT(1)=3                                                            WIM05710
    ICT(2)=4                                                            WIM05720
    ICT(3)=3                                                            WIM05730
```

```
      GOTO 100                                                          WIM05740
  731 CALL TRSING(5)                                                    WIM05750
      CALL TRSING(6)                                                    WIM05760
      CALL TRSING(7)                                                    WIM05770
      ICT(1)=5                                                          WIM05780
      ICT(2)=4                                                          WIM05790
      ICT(3)=1                                                          WIM05800
      ICT(3)=2                                                          WIM05810
      GOTO 100                                                          WIM05820
   74 CALL DQUAD(2,3,4,5)                                               WIM05830
      IF(IAXS(6).LE.50) GOTO 741                                        WIM05840
      CALL TRSING(6)                                                    WIM05850
      CALL TRSING(7)                                                    WIM05860
      IF(IAXS(6).GT.80) THEN                                            WIM05870
         ICT(1)=3                                                       WIM05880
         ICT(2)=5                                                       WIM05890
         ICT(3)=7                                                       WIM05900
         GOTO 100                                                       WIM05910
      ELSE                                                              WIM05920
         ICT(1)=4                                                       WIM05930
         ICT(2)=5                                                       WIM05940
         ICT(3)=7                                                       WIM05950
         GOTO 100                                                       WIM05960
      END IF                                                            WIM05970
  741 CALL TRTAN(6,7)                                                   WIM05980
      ICT(1)=3                                                          WIM05990
      ICT(2)=5                                                          WIM06000
      ICT(3)=2                                                          WIM06010
      GOTO 100                                                          WIM06020
C  * * * * * * * 8-AXLE TRUCK * * *                                    WIM06030
   81 CALL STEER                                                        WIM06040
      IAXD23=IAXS(2)+IAXS(3)                                            WIM06050
      IAXD24=IAXS(2)+IAXS(3)+IAXS(4)                                    WIM06060
      IAXD37=IAXS(3)+IAXS(4)+IAXS(5)+IAXS(6)+IAXS(7)                    WIM06070
      IAXD27=IAXS(2)+IAXS(3)+IAXS(4)+IAXS(5)+IAXS(6)+IAXS(7)            WIM06080
      IAXT34=IAXS(3)+IAXS(4)                                            WIM06090
      IAXT57=IAXS(5)+IAXS(6)+IAXS(7)                                    WIM06100
      IAXT67=IAXS(6)+IAXS(7)                                            WIM06110
      IF(IAXD27.LE.300) THEN                                            WIM06120
      CALL DRTAN(2,3)                                                   WIM06130
      CALL DRTAN(4,5)                                                   WIM06140
      CALL DRTRI(6,7,8)                                                 WIM06150
      ICT(1)=2                                                          WIM06160
      ICT(2)=8                                                          WIM06170
      GOTO 100                                                          WIM06180
      ELSE                                                              WIM06190
      END IF                                                            WIM06200
      IF(IAXS(2).GT.55) GOTO 815                                        WIM06210
      IF(IAXS(2).LE.55.AND.IAXS(3).GT.50) GOTO 82                       WIM06220
      IF(IAXD23.LE.100.AND.IAXS(4).GT.50) GOTO 83                       WIM06230
      IF(IAXD24.LE.150) GOTO 84                                         WIM06240
  815 CALL DRSING                                                       WIM06250
      IF(IAXD37.LE.300) THEN                                            WIM06260
      CALL TRTAN(3,4)                                                   WIM06270
```

```
      CALL TRTAN(5,6)                                                   WIM06280
      CALL TRTAN(7,8)                                                   WIM06290
      ICT(1)=3                                                          WIM06300
      ICT(2)=2                                                          WIM06310
      ICT(3)=6                                                          WIM06320
      GOTO 100                                                          WIM06330
      ELSE IF(IAXT34.LE.100.AND.IAXT67.LE.100.AND.IAXS(5).GT.80) THEN   WIM06340
      CALL TRTRI(3,4,5)                                                 WIM06350
      CALL TRTRI(6,7,8)                                                 WIM06360
      ICT(1)=5                                                          WIM06370
      ICT(2)=2                                                          WIM06380
      ICT(3)=3                                                          WIM06390
      ICT(4)=3                                                          WIM06400
      GOTO 100                                                          WIM06410
      ELSE IF(IAXS(2).GT.80.AND.IAXS(4).GT.80.AND.IAXS(6).GT.80.AND.    WIM06420
     1    IAXS(3).LE.50.AND.IAXS(5).LE.50.AND.IAXS(7).LE.50) THEN       WIM06430
      CALL TRTAN(3,4)                                                   WIM06440
      CALL TRTAN(5,6)                                                   WIM06450
      CALL TRTAN(7,8)                                                   WIM06460
      ICT(1)=5                                                          WIM06470
      ICT(2)=2                                                          WIM06480
      ICT(3)=2                                                          WIM06490
      ICT(4)=4                                                          WIM06500
      GOTO 100                                                          WIM06510
      ELSE                                                              WIM06520
      END IF                                                            WIM06530
   82 CALL DRTAN(2,3)                                                   WIM06540
      IF(IAXS(4).LE.50.AND.IAXS(5).GT.50.AND.IAXS(6).GT.80.AND.         WIM06550
     1    IAXS(7).LE.50) GOTO 88                                        WIM06560
      IF(IAXS(4).LE.50.AND.IAXS(5).LE.50.AND.IAXS(6).GT.80) GOTO 89     WIM06570
      CALL TQUINT(4,5,6,7,8)                                            WIM06580
      ICT(1)=3                                                          WIM06590
      ICT(2)=3                                                          WIM06600
      ICT(3)=5                                                          WIM06610
      GOTO 100                                                          WIM06620
   89 CALL TRTRI(4,5,6)                                                 WIM06630
      IF(IAXS(7).GT.50) GOTO 891                                        WIM06640
      CALL TRTAN(7,8)                                                   WIM06650
      ICT(1)=5                                                          WIM06660
      ICT(2)=3                                                          WIM06670
      ICT(3)=3                                                          WIM06680
      ICT(4)=2                                                          WIM06690
      GOTO 100                                                          WIM06700
  891 CALL TRSING(7)                                                    WIM06710
      CALL TRSING(8)                                                    WIM06720
      ICT(1)=5                                                          WIM06730
      ICT(2)=3                                                          WIM06740
      ICT(3)=3                                                          WIM06750
      ICT(4)=7                                                          WIM06760
      GOTO 100                                                          WIM06770
   88 CALL TRTAN(4,5)                                                   WIM06780
      CALL TRSING(6)                                                    WIM06790
      CALL TRTAN(7,8)                                                   WIM06800
      ICT(1)=5                                                          WIM06810
```

```
    ICT(2)=3                                                          WIM06820
    ICT(3)=2                                                          WIM06830
    ICT(4)=3                                                          WIM06840
    GOTO 100                                                          WIM06850
 83 CALL DRTRI(2,3,4)                                                 WIM06860
    IF(IAXT57.LT.150) THEN                                            WIM06870
    CALL TQUAD(5,6,7,8)                                               WIM06880
    ICT(1)=3                                                          WIM06890
    ICT(2)=4                                                          WIM06900
    ICT(3)=4                                                          WIM06910
    GOTO 100                                                          WIM06920
    ELSE IF(IAXT67.LE.100.AND.IAXS(7).GT.50) THEN                     WIM06930
    CALL TRTRI(5,6,7)                                                 WIM06940
    CALL TRSING(8)                                                    WIM06950
    ICT(1)=5                                                          WIM06960
    ICT(2)=4                                                          WIM06970
    ICT(3)=3                                                          WIM06980
    ICT(4)=1                                                          WIM06990
    GOTO 100                                                          WIM07000
    ELSE                                                              WIM07010
832 CALL TRTAN(5,6)                                                   WIM07020
    IF(IAXS(7).LE.50) THEN                                            WIM07030
    CALL TRTAN(7,8)                                                   WIM07040
    ICT(1)=5                                                          WIM07050
    ICT(2)=4                                                          WIM07060
    ICT(3)=2                                                          WIM07070
    ICT(4)=2                                                          WIM07080
    GOTO 100                                                          WIM07090
    ELSE                                                              WIM07100
    CALL TRSING(7)                                                    WIM07110
    CALL TRSING(8)                                                    WIM07120
    ICT(1)=5                                                          WIM07130
    ICT(2)=4                                                          WIM07140
    ICT(3)=2                                                          WIM07150
    ICT(4)=7                                                          WIM07160
    GOTO 100                                                          WIM07170
    END IF                                                            WIM07180
    END IF                                                            WIM07190
 84 CALL DQUAD(2,3,4,5)                                               WIM07200
    IF(IAXT67.LE.100) GOTO 841                                        WIM07210
    IF(IAXS(6).GT.80.AND.IAXS(7).LE.50) THEN                          WIM07220
    CALL TRSING(6)                                                    WIM07230
    CALL TRTAN(7,8)                                                   WIM07240
    ICT(1)=4                                                          WIM07250
    ICT(2)=5                                                          WIM07260
    ICT(3)=3                                                          WIM07270
    GOTO 100                                                          WIM07280
    ELSE                                                              WIM07290
    CALL TRSING(6)                                                    WIM07300
    CALL TRSING(7)                                                    WIM07310
    CALL TRSING(8)                                                    WIM07320
    ICT(1)=5                                                          WIM07330
    ICT(2)=5                                                          WIM07340
    ICT(3)=1                                                          WIM07350
```

```
      ICT(4)=2                                                          WIM07360
      GOTO 100                                                          WIM07370
      END IF                                                            WIM07380
  841 CALL TRTRI(6,7,8)                                                 WIM07390
      ICT(1)=3                                                          WIM07400
      ICT(2)=5                                                          WIM07410
      ICT(3)=3                                                          WIM07420
      GOTO 100                                                          WIM07430
C  * * * * * * * 9-AXLE TRUCK * * *                                    WIM07440
   91 CALL STEER                                                        WIM07450
      IAX23=IAXS(2)+IAXS(3)                                             WIM07460
      IAX24=IAXS(2)+IAXS(3)+IAXS(4)                                     WIM07470
      IAX25=IAXS(2)+IAXS(3)+IAXS(4)+IAXS(5)                             WIM07480
      IAXT45=IAXS(4)+IAXS(5)                                            WIM07490
      IAXT56=IAXS(5)+IAXS(6)                                            WIM07500
      IAXT67=IAXS(6)+IAXS(7)                                            WIM07510
      IAXT78=IAXS(7)+IAXS(8)                                            WIM07520
      IAXT68=IAXS(6)+IAXS(7)+IAXS(8)                                    WIM07530
      IAXT58=IAXS(5)+IAXS(6)+IAXS(7)+IAXS(8)                            WIM07540
      IAXT48=IAXS(4)+IAXS(5)+IAXS(6)+IAXS(7)+IAXS(8)                    WIM07550
      IF(IAXS(2).LE.55.AND.IAXS(3).GT.80.AND.IAXT48.LE.250) THEN        WIM07560
          CALL DRTAN(2,3)                                               WIM07570
          CALL TRTRI(4,5,6)                                             WIM07580
          CALL TRTRI(7,8,9)                                             WIM07590
          ICT(1)=3                                                      WIM07600
          ICT(2)=3                                                      WIM07610
          ICT(3)=6                                                      WIM07620
          GOTO 100                                                      WIM07630
      ELSE IF(IAXS(2).LE.55.AND.IAXS(3).GT.80.AND.IAXT45.LE.100.AND.    WIM07640
     1    IAXS(6).GT.80.AND.IAXS(7).GT.80.AND.IAXS(8).LE.50) THEN       WIM07650
          CALL DRTAN(2,3)                                               WIM07660
          CALL TRTRI(4,5,6)                                             WIM07670
          CALL TRSING(7)                                                WIM07680
          CALL TRTAN(8,9)                                               WIM07690
          ICT(1)=5                                                      WIM07700
          ICT(2)=3                                                      WIM07710
          ICT(3)=3                                                      WIM07720
          ICT(4)=8                                                      WIM07730
          GOTO 100                                                      WIM07740
      ELSE                                                              WIM07750
          CALL DRTAN(2,3)                                               WIM07760
          CALL TRTAN(4,5)                                               WIM07770
          IF(IAXS(5).GT.80.AND.IAXS(6).GT.80) THEN                      WIM07780
                CALL TRSING(6)                                          WIM07790
                CALL TRSING(7)                                          WIM07800
                CALL TRSING(8)                                          WIM07810
                CALL TRSING(9)                                          WIM07820
                ICT(1)=7                                                WIM07830
                ICT(2)=3                                                WIM07840
                ICT(3)=2                                                WIM07850
                ICT(4)=2                                                WIM07860
                ICT(5)=2                                                WIM07870
                GOTO 100                                                WIM07880
          ELSE IF(IAXS(5).GT.80.AND.IAXS(6).LE.50.AND.IAXS(7).GT.80.AND.WIM07890
```

```
1          IAXS(8).LE.50) THEN                                          WIM07900
           CALL TRTAN(6,7)                                              WIM07910
           CALL TRTAN(8,9)                                              WIM07920
           ICT(1)=5                                                     WIM07930
           ICT(2)=3                                                     WIM07940
           ICT(3)=2                                                     WIM07950
           ICT(4)=9                                                     WIM07960
           GOTO 100                                                     WIM07970
     END IF                                                             WIM07980
 END IF                                                                 WIM07990
 IF(IAX23.LE.100.AND.IAXS(4).GT.80.AND.IAXT58.LE.200) THEN              WIM08000
     CALL DRTRI(2,3,4)                                                  WIM08010
     CALL TRTAN(5,6)                                                    WIM08020
     CALL TRTRI(7,8,9)                                                  WIM08030
     ICT(1)=3                                                           WIM08040
     ICT(2)=4                                                           WIM08050
     ICT(3)=5                                                           WIM08060
     GOTO 100                                                           WIM08070
 ELSE IF(IAX23.LE.100.AND.IAXS(4).GT.80.AND.IAXT56.LE.100.AND.          WIM08080
1    IAXS(7).GT.80.AND.IAXS(8).GT.80) THEN                              WIM08090
           CALL DRTRI(2,3,4)                                            WIM08100
           CALL TRTAN(5,6)                                              WIM08110
           CALL TRSING(7)                                               WIM08120
           CALL TRSING(8)                                               WIM08130
           CALL TRSING(9)                                               WIM08140
           ICT(1)=7                                                     WIM08150
           ICT(2)=4                                                     WIM08160
           ICT(3)=2                                                     WIM08170
           ICT(4)=1                                                     WIM08180
           ICT(5)=7                                                     WIM08190
           GOTO 100                                                     WIM08200
 ELSE IF(IAX23.LE.100.AND.IAXS(4).GT.80.AND.IAXT56.LE.100.AND.          WIM08210
1    IAXS(7).GT.80.AND.IAXS(8).LE.80) THEN                              WIM08220
           CALL DRTRI(2,3,4)                                            WIM08230
           CALL TRTRI(5,6,7)                                            WIM08240
           CALL TRTAN(8,9)                                              WIM08250
           ICT(1)=5                                                     WIM08260
           ICT(2)=4                                                     WIM08270
           ICT(3)=3                                                     WIM08280
           ICT(4)=7                                                     WIM08290
           GOTO 100                                                     WIM08300
 ELSE IF(IAX23.LE.100.AND.IAXS(4).GT.80.AND.IAXS(5).LE.50.AND.          WIM08310
1    IAXS(6).GT.80.AND.IAXS(7).GT.80.AND.IAXS(8).LE.80) THEN            WIM08320
           CALL DRTRI(2,3,4)                                            WIM08330
           CALL TRTAN(5,6)                                              WIM08340
           CALL TRSING(7)                                               WIM08350
           CALL TRTAN(8,9)                                              WIM08360
           ICT(1)=5                                                     WIM08370
           ICT(2)=4                                                     WIM08380
           ICT(3)=2                                                     WIM08390
           ICT(4)=8                                                     WIM08400
           GOTO 100                                                     WIM08410
 ELSE                                                                   WIM08420
 ENDIF                                                                  WIM08430
```

```
     IF(IAX24.LE.150.AND.IAXS(5).GT.80.AND.IAXT68.LE.150) THEN          WIM08440
         CALL DQUAD(2,3,4,5)                                            WIM08450
         CALL TQUAD(6,7,8,9)                                            WIM08460
         ICT(1)=3                                                       WIM08470
         ICT(2)=5                                                       WIM08480
         ICT(3)=4                                                       WIM08490
         GOTO 100                                                       WIM08500
     ELSE IF(IAX24.LE.150.AND.IAXS(5).GT.80.AND.IAXT67.LE.100.AND.      WIM08510
    1    IAXS(8).GT.80) THEN                                            WIM08520
               CALL DQUAD(2,3,4,5)                                      WIM08530
               CALL TRTRI(6,7,8)                                        WIM08540
               CALL TRSING(9)                                           WIM08550
               ICT(1)=5                                                 WIM08560
               ICT(2)=5                                                 WIM08570
               ICT(3)=3                                                 WIM08580
               ICT(4)=1                                                 WIM08590
               GOTO 100                                                 WIM08600
     ELSE IF(IAX24.LE.150.AND.IAXS(5).GT.80.AND.IAXS(6).LE.50.AND.      WIM08610
    1    IAXS(7).GT.80.AND.IAXS(8).GT.80) THEN                          WIM08620
               CALL DQUAD(2,3,4,5)                                      WIM08630
               CALL TRTAN(6,7)                                          WIM08640
               CALL TRSING(8)                                           WIM08650
               CALL TRSING(9)                                           WIM08660
               ICT(1)=5                                                 WIM08670
               ICT(2)=5                                                 WIM08680
               ICT(3)=2                                                 WIM08690
               ICT(4)=7                                                 WIM08700
               GOTO 100                                                 WIM08710
     ELSE                                                               WIM08720
     ENDIF                                                              WIM08730
     IF(IAX25.LE.200.AND.IAXT78.LE.100) THEN                            WIM08740
         CALL DQUINT(2,3,4,5,6)                                         WIM08750
         CALL TRTRI(7,8,9)                                              WIM08760
         ICT(1)=3                                                       WIM08770
         ICT(2)=6                                                       WIM08780
         ICT(3)=3                                                       WIM08790
         GOTO 100                                                       WIM08800
     ELSE                                                               WIM08810
     ENDIF                                                              WIM08820
 100 CALL SUM                                                           WIM08830
     CALL COAL                                                          WIM08840
 200 CALL PRN                                                           WIM08850
     GOTO 22                                                            WIM08860
1234 CONTINUE                                                           WIM08870
     WRITE (8,550) NERRORS0                                             WIM08880
     WRITE (8,551) NERRORS1                                             WIM08890
     WRITE (8,552) NERRORS2                                             WIM08900
     WRITE (8,553) NERRORS3                                             WIM08910
     WRITE (8,554) NERRORS4                                             WIM08920
     WRITE (8,555) NERRORS5                                             WIM08930
     WRITE (8,556) NGOOD                                                WIM08940
 550 FORMAT (I6,5X,'TWO-AXLE VEHICLES LESS THAN 12 FEET')               WIM08950
 551 FORMAT (I6,5X,'AXLE(S) WITHOUT WEIGHT')                            WIM08960
 552 FORMAT (I6,5X,'AXLE SPACING(S) LESS THAT 1.5 FEET')                WIM08970
```

```
  553 FORMAT (I6,5X,'MISMATCHES BETWEEN NUMBERS OF AXLE WEIGHTS AND SPACWIM08980
     *INGS')                                                            WIM08990
  554 FORMAT (I6,5X,'>10 PERCENT DIFFERENCE BETWEEN GROSS WEIGHT AND SUMWIM09000
     * OF AXLE WEIGHT')                                                 WIM09010
  555 FORMAT (I6,5X,'>10 PERCENT DIFFERENCE BETWEEN WHEELBASE AND SUM OFWIM09020
     * AXLE SPACINGS')                                                 WIM09030
  556 FORMAT (I6,5X,'SUCCESSFUL ENTRIES')                              WIM09040
C     RETURN                                                           WIM09050
      END                                                              WIM09060
C  * * * * END OF MAIN PROGRAM * * * * * * * *                       WIM09070
      SUBROUTINE INIT                                                  WIM09080
      COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                       WIM09090
      COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS WIM09100
      COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT WIM09110
     1,AMEQ,AA                                                         WIM09120
      COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)         WIM09130
      COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV     WIM09140
      DO 10 I=1,13                                                     WIM09150
      IAW(I)=0                                                         WIM09160
      ISW(I)=0                                                         WIM09170
   10 CONTINUE                                                         WIM09180
      DO 11 J=1,12                                                     WIM09190
      IAXS(J)=0                                                        WIM09200
   11 CONTINUE                                                         WIM09210
      DO 12 K=1,6                                                      WIM09220
      ICT(K)=0                                                         WIM09230
   12 CONTINUE                                                         WIM09240
      DO 14 IK=1,10                                                    WIM09250
      ICONTN(IK)=0                                                     WIM09260
   14 CONTINUE                                                         WIM09270
      ITWRC=0                                                          WIM09280
      AV=0.                                                            WIM09290
      ICK=0                                                            WIM09300
      ISC=0                                                            WIM09310
      FC=0                                                             WIM09320
      SIN=0                                                            WIM09330
      IYR=0                                                            WIM09340
      IDT=0                                                            WIM09350
      IAMO=0                                                           WIM09360
      IDAY=0                                                           WIM09370
      IHOUR=0                                                          WIM09380
      IVTC=0                                                           WIM09390
      ITWT=0                                                           WIM09400
      ITOTWT=0                                                         WIM09410
      IAXTOT=0                                                         WIM09420
      IRSN=0                                                           WIM09430
      ICMOD=99999                                                      WIM09440
      CNVEQ=0                                                          WIM09450
      CV=0                                                             WIM09460
      IAX2=0                                                           WIM09470
      IAX3=0                                                           WIM09480
      IAXD=0                                                           WIM09490
      IAXTR=0                                                          WIM09500
      IAXD4=0                                                          WIM09510
```

```
      IAXT4=0                                                                WIM09520
      IAXD3=0                                                                WIM09530
      IAXT3=0                                                                WIM09540
      IAXT3=0                                                                WIM09550
      IAXT6=0                                                                WIM09560
      IAXT5=0                                                                WIM09570
      IAXT31=0                                                               WIM09580
      IAXT32=0                                                               WIM09590
      IAXTOT=0                                                               WIM09600
      JCONT=1                                                                WIM09610
      IAFC=0                                                                 WIM09620
      RETURN                                                                 WIM09630
      END                                                                    WIM09640
      SUBROUTINE STEER                                                       WIM09650
      COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                             WIM09660
      COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS       WIM09670
      COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT      WIM09680
     1,AMEQ,AA                                                               WIM09690
      COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)               WIM09700
      COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV           WIM09710
      IF(ITYPE.EQ.2) GOTO 20                                                 WIM09720
      A=FAEQ(1)                                                              WIM09730
      B=FBEQ(1)                                                              WIM09740
      GOTO 25                                                                WIM09750
   20 A=RAEQ(1)                                                              WIM09760
      B=RBEQ(1)                                                              WIM09770
   25 CONTINUE                                                               WIM09780
      W=100*IAW(1)                                                           WIM09790
      S=0.01*(A+B*W)                                                         WIM09800
      ISW(1)=S                                                               WIM09810
      RETURN                                                                 WIM09820
      END                                                                    WIM09830
      SUBROUTINE DRSING                                                      WIM09840
      COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                             WIM09850
      COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS       WIM09860
      COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT      WIM09870
     1,AMEQ,AA                                                               WIM09880
      COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)               WIM09890
      COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV           WIM09900
      IF(ITYPE.EQ.2) GOTO 20                                                 WIM09910
      A=FAEQ(2)                                                              WIM09920
      B=FBEQ(2)                                                              WIM09930
      GOTO 25                                                                WIM09940
   20 A=RAEQ(2)                                                              WIM09950
      B=RBEQ(2)                                                              WIM09960
   25 CONTINUE                                                               WIM09970
      W=100*IAW(2)                                                           WIM09980
      S=0.01*(A+B*W)                                                         WIM09990
      ISW(2)=S                                                               WIM10000
      RETURN                                                                 WIM10010
      END                                                                    WIM10020
      SUBROUTINE TRSING(I1)                                                  WIM10030
      COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                             WIM10040
      COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS       WIM10050
```

```
      COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT WIM10060
     1,AMEQ,AA                                                         WIM10070
      COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)         WIM10080
      COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV     WIM10090
      IF(ITYPE.EQ.2) GOTO 20                                           WIM10100
      A=FAEQ(3)                                                        WIM10110
      B=FBEQ(3)                                                        WIM10120
      GOTO 25                                                          WIM10130
   20 A=RAEQ(3)                                                        WIM10140
      B=RBEQ(3)                                                        WIM10150
   25 CONTINUE                                                         WIM10160
C  3  FORMAT(' TRSING =',I5,' IAW(I1)=',I5,' ISW(I1)=',I5)             WIM10170
      W=100*IAW(I1)                                                    WIM10180
      S=0.01*(A+B*W)                                                   WIM10190
      ISW(I1)=S                                                        WIM10200
C     WRITE(*,3)I1,IAW(I1),ISW(I1)                                     WIM10210
      RETURN                                                           WIM10220
      END                                                              WIM10230
      SUBROUTINE DRTAN(I1,I2)                                          WIM10240
      COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                       WIM10250
      COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS WIM10260
      COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT WIM10270
     1,AMEQ,AA                                                         WIM10280
      COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)         WIM10290
      COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV     WIM10300
      IF(ITYPE.EQ.2) GOTO 20                                           WIM10310
      A=FAEQ(4)                                                        WIM10320
      B=FBEQ(4)                                                        WIM10330
      GOTO 25                                                          WIM10340
   20 A=RAEQ(4)                                                        WIM10350
      B=RBEQ(4)                                                        WIM10360
   25 CONTINUE                                                         WIM10370
      P1=100*IAW(I1)                                                   WIM10380
      P2=100*IAW(I2)                                                   WIM10390
      W=P1+P2                                                          WIM10400
      WT=(A+B*W)                                                       WIM10410
      ISW(I1)=0.01*WT*P1/W                                             WIM10420
      ISW(I2)=0.01*WT*P2/W                                             WIM10430
      RETURN                                                           WIM10440
      END                                                              WIM10450
      SUBROUTINE TRTAN(I1,I2)                                          WIM10460
      COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                       WIM10470
      COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS WIM10480
      COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT WIM10490
     1,AMEQ,AA                                                         WIM10500
      COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)         WIM10510
      COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV     WIM10520
      IF(ITYPE.EQ.2) GOTO 20                                           WIM10530
      A=FAEQ(5)                                                        WIM10540
      B=FBEQ(5)                                                        WIM10550
      GOTO 25                                                          WIM10560
   20 A=RAEQ(5)                                                        WIM10570
      B=RBEQ(5)                                                        WIM10580
   25 CONTINUE                                                         WIM10590
```

```
   P1=100*IAW(I1)                                                        WIM10600
   P2=100*IAW(I2)                                                        WIM10610
   W=P1+P2                                                               WIM10620
   WT=(A+B*W)                                                            WIM10630
   ISW(I1)=0.01*WT*P1/W                                                  WIM10640
   ISW(I2)=0.01*WT*P2/W                                                  WIM10650
   RETURN                                                                WIM10660
   END                                                                   WIM10670
   SUBROUTINE DRTRI(I1,I2,I3)                                            WIM10680
   COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                            WIM10690
   COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS      WIM10700
   COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT     WIM10710
  1,AMEQ,AA                                                              WIM10720
   COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)              WIM10730
   COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV          WIM10740
   IF(ITYPE.EQ.2) GOTO 20                                                WIM10750
   A=FAEQ(6)                                                             WIM10760
   B=FBEQ(6)                                                             WIM10770
   GOTO 25                                                               WIM10780
20 A=RAEQ(6)                                                             WIM10790
   B=RBEQ(6)                                                             WIM10800
25 CONTINUE                                                              WIM10810
   P1=100*IAW(I1)                                                        WIM10820
   P2=100*IAW(I2)                                                        WIM10830
   P3=100*IAW(I3)                                                        WIM10840
   W=P1+P2+P3                                                            WIM10850
   WT=(A+B*W)                                                            WIM10860
   ISW(I1)=0.01*WT*P1/W                                                  WIM10870
   ISW(I2)=0.01*WT*P2/W                                                  WIM10880
   ISW(I3)=0.01*WT*P3/W                                                  WIM10890
   RETURN                                                                WIM10900
   END                                                                   WIM10910
   SUBROUTINE TRTRI(I1,I2,I3)                                            WIM10920
   COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                            WIM10930
   COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS      WIM10940
   COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT     WIM10950
  1,AMEQ,AA                                                              WIM10960
   COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)              WIM10970
   COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV          WIM10980
   IF(ITYPE.EQ.2) GOTO 20                                                WIM10990
   A=FAEQ(7)                                                             WIM11000
   B=FBEQ(7)                                                             WIM11010
   GOTO 25                                                               WIM11020
20 A=RAEQ(7)                                                             WIM11030
   B=RBEQ(7)                                                             WIM11040
25 CONTINUE                                                              WIM11050
   P1=100*IAW(I1)                                                        WIM11060
   P2=100*IAW(I2)                                                        WIM11070
   P3=100*IAW(I3)                                                        WIM11080
   W=P1+P2+P3                                                            WIM11090
   WT=(A+B*W)                                                            WIM11100
   ISW(I1)=0.01*WT*P1/W                                                  WIM11110
   ISW(I2)=0.01*WT*P2/W                                                  WIM11120
   ISW(I3)=0.01*WT*P3/W                                                  WIM11130
```

```
 RETURN                                                                 WIM11140
 END                                                                    WIM11150
 SUBROUTINE DQUAD(I1,I2,I3,I4)                                          WIM11160
 COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                             WIM11170
 COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS       WIM11180
 COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT      WIM11190
1,AMEQ,AA                                                               WIM11200
 COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)               WIM11210
 COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV           WIM11220
 CALL DRTAN(I1,I2)                                                      WIM11230
 CALL DRTAN(I3,I4)                                                      WIM11240
 RETURN                                                                 WIM11250
 END                                                                    WIM11260
 SUBROUTINE TQUAD(I1,I2,I3,I4)                                          WIM11270
 COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                             WIM11280
 COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS       WIM11290
 COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT      WIM11300
1,AMEQ,AA                                                               WIM11310
 COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)               WIM11320
 COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV           WIM11330
 CALL TRTAN(I1,I2)                                                      WIM11340
 CALL TRTAN(I3,I4)                                                      WIM11350
 RETURN                                                                 WIM11360
 END                                                                    WIM11370
 SUBROUTINE DQUINT(I1,I2,I3,I4,I5)                                      WIM11380
 COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                             WIM11390
 COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS       WIM11400
 COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT      WIM11410
1,AMEQ,AA                                                               WIM11420
 COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)               WIM11430
 COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV           WIM11440
 CALL DRTAN(I1,I2)                                                      WIM11450
 CALL DRTRI(I3,I4,I5)                                                   WIM11460
 RETURN                                                                 WIM11470
 END                                                                    WIM11480
 SUBROUTINE TQUINT(I1,I2,I3,I4,I5)                                      WIM11490
 COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                             WIM11500
 COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS       WIM11510
 COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT      WIM11520
1,AMEQ,AA                                                               WIM11530
 COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)               WIM11540
 COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV           WIM11550
 CALL TRTAN(I1,I2)                                                      WIM11560
 CALL TRTRI(I3,I4,I5)                                                   WIM11570
 RETURN                                                                 WIM11580
 END                                                                    WIM11590
 SUBROUTINE SEXTET(I1,I2,I3,I4,I5,I6)                                   WIM11600
 COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                             WIM11610
 COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS       WIM11620
 COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT      WIM11630
1,AMEQ,AA                                                               WIM11640
 COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)               WIM11650
 COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV           WIM11660
 CALL TRTRI(I1,I2,I3)                                                   WIM11670
```

```
      CALL TRTRI(I4,I5,I6)                                                  WIM11680
      RETURN                                                                WIM11690
      END                                                                   WIM11700
      SUBROUTINE SUM                                                        WIM11710
      COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                            WIM11720
      COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS      WIM11730
      COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT     WIM11740
     1,AMEQ,AA                                                              WIM11750
      COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)              WIM11760
      COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV          WIM11770
      ITOTWT=0                                                              WIM11780
      DO 30 J=1,13                                                          WIM11790
      ITOTWT=ITOTWT+ISW(J)                                                  WIM11800
   30 CONTINUE                                                              WIM11810
      RETURN                                                                WIM11820
      END                                                                   WIM11830
C  SUBROUTINE COAL FOLLOWS                                                  WIM11840
      SUBROUTINE COAL                                                       WIM11850
      COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                            WIM11860
      COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS      WIM11870
      COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT     WIM11880
     1,AMEQ,AA                                                              WIM11890
      COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)              WIM11900
      COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV          WIM11910
      ICMOD=99999                                                           WIM11920
      ATOTWT=ITOTWT*.1                                                      WIM11930
      AX3=IAXS(3)*0.1                                                       WIM11940
      IF(ICT(1).EQ.3.AND.ICT(2).EQ.3.AND.ICT(3).EQ.2.AND.ITOTWT             WIM11950
     1 .GT.800) THEN                                                        WIM11960
          CNVEQ=C5NA+C5NB*ATOTWT+C5NS                                       WIM11970
          GOTO 10                                                           WIM11980
      ELSE IF(ICT(1).EQ.3.AND.ICT(2).EQ.3.AND.ICT(3).EQ.3.AND.ITOTWT        WIM11990
     1 .GT.900) THEN                                                        WIM12000
          CNVEQ=C6NA+C6NB*ATOTWT+C6NS                                       WIM12010
          GOTO 10                                                           WIM12020
      ELSE IF(ICT(1).EQ.2.AND.ICT(2).EQ.3.AND.ICT(3).EQ.0                   WIM12030
     1      .AND.ITOTWT.GT.600) THEN                                        WIM12040
          ICMOD=11200                                                       WIM12050
          GO TO 15                                                          WIM12060
      ELSE IF(ICT(1).EQ.2.AND.ICT(2).EQ.4.AND.ICT(3).EQ.0                   WIM12070
     1      .AND.ITOTWT.GT.750) THEN                                        WIM12080
          ICMOD=11200                                                       WIM12090
          GO TO 15                                                          WIM12100
      ELSE                                                                  WIM12110
      END IF                                                                WIM12120
      GO TO 15                                                              WIM12130
   10 CONTINUE                                                              WIM12140
      CV=ATOTWT/AX3                                                         WIM12150
      IF(CV.GE.CNVEQ) THEN                                                  WIM12160
   11 ICMOD=11200                                                           WIM12170
      ELSE                                                                  WIM12180
      END IF                                                                WIM12190
   15 CONTINUE                                                              WIM12200
      RETURN                                                                WIM12210
```

```
      END                                                                WIM12220
      SUBROUTINE PRN                                                     WIM12230
      COMMON/DAN/FAEQ(7),FBEQ(7),RAEQ(7),RBEQ(7)                         WIM12240
      COMMON/COL/C5A,C5B,C6A,C6B,C5NA,C5NB,C6NA,C6NB,C5S,C6S,C5NS,C6NS   WIM12250
      COMMON/DAT1/ITWRC,ISC,FC,SIN,IYR,IAMO,IDAY,IHOUR,ICMOD,ITWT,JCONT  WIM12260
      COMMON/DAT2/IAXS(70),IAXTOT,IRSN,ICONTN(10),ITYPE,ICT(6)           WIM12270
      COMMON/DAT3/IAW(70),ISW(70),BD(9),ICH(6),IDT,ITOTWT,CNVEQ,CV       WIM12280

C STMT 1103 WRITES UNADJUSTED WEIGHT VALUES TO OUTPUT DATASET  (7/99)
C           AND DOES NOT ASSIGN A VEHICLE TYPE CODE (7/00)
C STMT 1102 WRITES THE CONTINUATION RECORD

C1103 WRITE(6,1100)ITWRC,ISC,FC,SIN,IDT,IYR,IAMO,IDAY,IHOUR,(ICT(I),     WIM12290
C    1 I=1,6),(BD(J),J=7,9),(ICH(J),J=1,6),ICMOD,ITWT,(IAW(L),L=1,5),    WIM12300
C    2 (IAXS(M),M=1,4),IAXTOT,IRSN,ICONTN(1)                            WIM12310
 1103 WRITE(6,1100)ITWRC,ISC,FC,SIN,IDT,IYR,IAMO,IDAY,IHOUR,
     1(BD(J),J=7,9),(ICH(J),J=1,6),ICMOD,ITWT,(IAW(L),L=1,5),
     2 (IAXS(M),M=1,4),IAXTOT,IRSN,ICONTN(1)
      IF(ICONTN(1).EQ.0) GOTO 20                                         WIM12320
      DO 55 KK=1,JCONT                                                   WIM12330
      LRT=KK*8-2                                                         WIM12340
      LOP=KK*8+5                                                         WIM12350
      MRT=KK*8-3                                                         WIM12360
      MOP=KK*8+4                                                         WIM12370
C 1102 WRITE(6,1101)ITWRC,ISC,FC,SIN,IDT,IYR,IAMO,IDAY,IHOUR,(ICT(I),    WIM12380
C     1  I=1,6),(IAW(L),L=LRT,LOP),(IAXS(M),M=MRT,MOP),IRSN,ICONTN(KK+1)WIM12390
 1102 WRITE(6,1101)ITWRC,ISC,FC,SIN,IDT,IYR,IAMO,IDAY,IHOUR,(BD(J),
     1 J=7-9),(IAW(L),L=LRT,LOP),(IAXS(M),M=MRT,MOP),IRSN,ICONTN(KK+1)
      IF(ICONTN(KK+1).EQ.9) GOTO 20                                      WIM12400
   55 CONTINUE                                                           WIM12410
 1100 FORMAT(I1,I2,A2,A3,I1,4I2,6I1,3A1,6I1,3X,I5,1X,I4,9I3,I4,I3,I1)    WIM12420
 1101 FORMAT(I1,I2,A2,A3,I1,4I2,6I1,5X,17I3,I1)                         WIM12430
   20 RETURN                                                            WIM12440
      END                                                               WIM12450
```

04_UNIT_EAL.FOR

THIS PROGRAM PROCESSES WEIGHT DATA AND CALCULATES ESALS PER VEHICLE

EDIT CHECKS ARE PERMORMED

ESAL VALUES ARE USED IN A LATER PRORAM

```
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *EAL00010
C     SET "IERR" TO DESIRED CODE TO PRINT REJECTED RECORDS                  EAL00020
C       1 = EXCESSIVE GROSS WEIGHT                                          EAL00030
C       2 = NEGLIGIBLE GROSS WEIGHT                                         EAL00040
C       3 = EXCESSIVE WHELLBASE                                             EAL00050
C       4 = NEGLIGIBLE WHEELBASE                                            EAL00060
C       9 = ALL REJECTED RECORDS                                            EAL00070
C       0 = NO REJECTED RECORDS                                             EAL00080
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *EAL00090
      INTEGER NERRORS,NN,VEHTYPE,CARDC,NCT,NA,NB,IVEH(3,14),               EAL00100
     +NVT,NAR,COMMOD,VEHT,IXLE(3,14)                                       EAL00110
      CHARACTER*64 ALL*80                                                  EAL00120
      DIMENSION SEAL(3,14),EALPAXLE(3,14),EALPVEH(3,14),AXLEPVEH(3,14),    EAL00130
     +W(20),S(20),AXLE(3,14),VEH(3,14),VT(13)                              EAL00140
      IERR=9                                                               EAL00150
      VT(5)=10000                                                          EAL00160
      VT(6)=16000                                                          EAL00170
      VT(7)=20000                                                          EAL00180
      VT(8)=28000                                                          EAL00190
      VT(9)=30000                                                          EAL00200
      VT(10)=34000                                                         EAL00210
      VT(11)=38000                                                         EAL00220
      VT(12)=42000                                                         EAL00230
      VT(13)=46000                                                         EAL00240
      CST1 = -5.064445                                                     EAL00250
      CST2 = 4.000647                                                      EAL00260
      CST3 = 0.0                                                           EAL00270
      CSI1 = -5.064445                                                     EAL00280
      CSI2 = 4.000647                                                      EAL00290
      CSI3 = 0.0                                                           EAL00300
      CTA1 = -6.043113                                                     EAL00310
      CTA2 = 3.950734                                                      EAL00320
      CTA3 = 0.0                                                           EAL00330
      CTR1 = -6.618829                                                     EAL00340
      CTR2 = 3.932525                                                      EAL00350
      CTR3 = 0.0                                                           EAL00360
      CQU1 = -2.589482                                                     EAL00370
      CQU2 = -2.224981                                                     EAL00380
      CQU1 =  1.923512                                                     EAL00390
      NERRORS1=0                                                           EAL00400
      NERRORS2=0                                                           EAL00410
      NERRORS3=0                                                           EAL00420
      NERRORS4=0                                                           EAL00430
      NERRORS5=0                                                           EAL00440
```

```
      NGOOD=0                                                                  EAL00450
      NSTART=0                                                                 EAL00460
      NN=0                                                                     EAL00470
      DO 20 I=1,3                                                              EAL00480
      DO 10 J=1,14                                                             EAL00490
      SEAL(I,J)=0.0                                                            EAL00500
      AXLE(I,J)=0.0                                                            EAL00510
      VEH(I,J)=0.0                                                             EAL00520
      EALPAXLE(I,J)=0.0                                                        EAL00530
      EALPVEH(I,J)=0.0                                                         EAL00540
      AXLEPVEH(I,J)=0.0                                                        EAL00550
      IFC=1                                                                    EAL00560
   10 CONTINUE                                                                 EAL00570
   20 CONTINUE                                                                 EAL00580
  100 CONTINUE                                                                 EAL00590
      DO 30 K=1,20                                                             EAL00600
      W(K)=0.0                                                                 EAL00610
      S(K)=0.0                                                                 EAL00620
   30 CONTINUE                                                                 EAL00630
      READ(5,1000,END=888) FC,VEHTYPE,COMMOD,TW,(W(L),L=1,5),(S(L),L=1,4EAL00640
     +),TS,CARDC,ALL,IYR                                                       EAL00650
 1000 FORMAT(T4,I2,T18,I4,T36,I5,T42,F4.1,T46,9F3.1,T73,F4.1,T80,I1,T1, EAL00660
     +A80,T10,I2)                                                              EAL00670
      IF (IFC.EQ.1) WRITE (8,99) FC,IYR                                        EAL00680
   99 FORMAT (5X,'FUNCTIONAL CLASS= ',I2,5X,'YEAR= 19',I2,/)                   EAL00690
      IFC=IFC+1                                                                EAL00700
      IF (CARDC.EQ.9) GO TO 100                                                EAL00710
      NVT=0                                                                    EAL00720
      NN=NN+1                                                                  EAL00730
      IF (VEHTYPE.EQ.1900) NVT=4                                               EAL00740
      IF (VEHTYPE.EQ.2200) NVT=5                                               EAL00750
      IF (VEHTYPE.EQ.2300) NVT=6                                               EAL00760
      IF (VEHTYPE.EQ.2400) NVT=7                                               EAL00770
      IF (VEHTYPE.EQ.2500) NVT=7                                               EAL00780
      IF (VEHTYPE.EQ.3210) NVT=8                                               EAL00790
      IF (VEHTYPE.EQ.3310) NVT=8                                               EAL00800
      IF (VEHTYPE.EQ.3220) NVT=8                                               EAL00810
      IF (VEHTYPE.EQ.3270) NVT=8                                               EAL00820
      IF (VEHTYPE.EQ.3230) NVT=9                                               EAL00830
      IF (VEHTYPE.EQ.3320) NVT=9                                               EAL00840
      IF (VEHTYPE.EQ.3370) NVT=9                                               EAL00850
      IF (VEHTYPE.EQ.3410) NVT=9                                               EAL00860
      IF (VEHTYPE.EQ.3330) NVT=10                                              EAL00870
      IF (VEHTYPE.EQ.3350) NVT=10                                              EAL00880
      IF (VEHTYPE.EQ.3510) NVT=10                                              EAL00890
      IF (VEHTYPE.EQ.3340) NVT=10                                              EAL00900
      IF (VEHTYPE.EQ.3240) NVT=10                                              EAL00910
      IF (VEHTYPE.EQ.3430) NVT=10                                              EAL00920
      IF (VEHTYPE.EQ.3440) NVT=10                                              EAL00930
      IF (VEHTYPE.EQ.3450) NVT=10                                              EAL00940
      IF (VEHTYPE.EQ.3390) NVT=10                                              EAL00950
      IF (VEHTYPE.EQ.3520) NVT=10                                              EAL00960
      IF (VEHTYPE.EQ.3540) NVT=10                                              EAL00970
      IF (VEHTYPE.EQ.5212) NVT=11                                              EAL00980
```

```
      IF (VEHTYPE.EQ.5222) NVT=12                                       EAL00990
      IF (VEHTYPE.EQ.5312) NVT=12                                       EAL01000
      IF (VEHTYPE.EQ.5329) NVT=13                                       EAL01010
      IF (VEHTYPE.GE.5400.AND.VEHTYPE.LE.5499) NVT=13                   EAL01020
      IF (VEHTYPE.EQ.5531) NVT=13                                       EAL01030
      VEHT=VEHTYPE/100                                                  EAL01040
      IF (VEHT.EQ.92.AND.CARDC.NE.9)  NERRORS5=NERRORS5+1               EAL01050
      IF (NVT.NE.0) GO TO 110                                           EAL01060
C     IF (NVT.EQ.0) WRITE (7,22222) ALL                                 EAL01070
C2222 FORMAT(A80)                                                       EAL01080
      IF (NVT.EQ.0) GO TO 100                                           EAL01090
  110 CONTINUE                                                          EAL01100
      IF (COMMOD.EQ.99999) THEN                                         EAL01110
      NCT=1                                                             EAL01120
      ELSE                                                              EAL01130
      NCT=2                                                             EAL01140
      END IF                                                            EAL01150
      IF (CARDC.EQ.1) READ (5,1100) (W(L),L=6,13),(S(L),L=5,12),ALL2    EAL01160
 1100 FORMAT(T29,8F3.1,T53,8F3.1,T1,A80)                                EAL01170
      SUMW=0.0                                                          EAL01180
      SUMS=0.0                                                          EAL01190
      NA=0                                                              EAL01200
      NB=0                                                              EAL01210
      DO 40 I=1,15                                                      EAL01220
      IF (W(I).GT.0.0) NA=NA+1                                          EAL01230
      SUMW=SUMW+W(I)                                                    EAL01240
      IF (S(I).GT.0.0) NB=NB+1                                          EAL01250
      SUMS=SUMS+S(I)                                                    EAL01260
   40 CONTINUE                                                          EAL01270
      NSTART=NSTART+1                                                   EAL01280
C*********************************************************************EAL01290
C   THE FOLLOWING SECTION CHECKS FOR ERRORS                             EAL01300
C*********************************************************************EAL01310
C     NERRORS1 = NO. OF ERRORS:  EXCESSIVE GROSS WEIGHT                 EAL01320
C     NERRORS2 = NO. OF ERRORS:  NEGLIGIBLE GROSS WEIGHT                EAL01330
C     NERRORS3 = NO. OF ERRORS:  EXCESSIVE WHEELBASE                    EAL01340
C     NERRORS4 = NO. OF ERRORS:  NEGLIGIBLE WHEELBASE                   EAL01350
C     NERRORS5 = NO. OF ERRORS:  VEHICLE TYPE "921"                     EAL01360
C*********************************************************************EAL01370
      ALW=(20+(NVT-4)*45/9)/10.0                                        EAL01380
      UW=(660+(NVT-4)*1000/6)/10.0                                      EAL01390
      ALS=(20+(NVT-4)*104/9)/10.0                                       EAL01400
      US=(460+(NVT-4)*540/9)/10.0                                       EAL01410
      IF (TW.GT.UW.OR.TW.GT.180) THEN                                   EAL01420
          NERRORS1=NERRORS1+1                                           EAL01430
      IF (IERR.NE.1.AND.IERR.NE.9) GO TO 100                            EAL01440
      WRITE (8,1111) ALL                                                EAL01450
 1111 FORMAT(T1,A80)                                                    EAL01460
      WRITE (8,11111) NVT,TW,UW,NERRORS1                                EAL01470
11111 FORMAT(5X,'NVT=',I2,5X,'TW= ',F5.1,5X,'UW= ',F5.1,5X,'EXCESSIVE GREAL01480
     +OSS WT  ERROR NO.',I5)                                            EAL01490
          GO TO 100                                                     EAL01500
      ELSE                                                              EAL01510
      END IF                                                            EAL01520
```

```
      IF (TW.LT.ALW) THEN                                                EAL01530
          NERRORS2=NERRORS2+1                                            EAL01540
      IF (IERR.NE.2.AND.IERR.NE.9) GO TO 100                             EAL01550
      WRITE (8,1112) ALL                                                 EAL01560
 1112 FORMAT(T1,A80)                                                     EAL01570
      WRITE (8,2222) NVT,TW,ALW,NERRORS2                                 EAL01580
 2222 FORMAT(5X,'NVT=',I2,5X,'TW= ',F5.1,5X,'LW= ',F4.1,5X,'NEGLIGIBLE GEAL01590
     +ROSS WT  ERROR NO.',I5)                                            EAL01600
          GO TO 100                                                      EAL01610
      ELSE                                                               EAL01620
      END IF                                                             EAL01630
      IF (TS.GT.US.OR.TS.GT.100) THEN                                    EAL01640
          NERRORS3=NERRORS3+1                                            EAL01650
      IF (IERR.NE.3.AND.IERR.NE.9) GO TO 100                             EAL01660
          WRITE (8,1113) ALL                                             EAL01670
 1113 FORMAT(T1,A80)                                                     EAL01680
          WRITE (8,3333) NVT,TS,US,NERRORS3                              EAL01690
 3333 FORMAT(5X,'NVT=',I2,5X,'TS= ',F5.1,5X,'US= ',F4.1,5X,'EXCESSIVE WHEAL01700
     +EELBASE   ERROR NO.',I5)                                           EAL01710
          GO TO 100                                                      EAL01720
      ELSE                                                               EAL01730
      END IF                                                             EAL01740
      IF (TS.LT.ALS) THEN                                                EAL01750
          NERRORS4=NERRORS4+1                                            EAL01760
      IF (IERR.NE.4.AND.IERR.NE.9) GO TO 100                             EAL01770
          WRITE (8,1114) ALL                                             EAL01780
 1114 FORMAT(T1,A80)                                                     EAL01790
          WRITE (8,4444) NVT,TS,ALS,NERRORS4                             EAL01800
 4444 FORMAT(5X,'NVT=',I2,5X,'TS= ',F5.1,5X,'LS= ',F4.1,5X,'NEGLIGIBLE WEAL01810
     +HEELBASE   ERROR NO.',I5)                                          EAL01820
          GO TO 100                                                      EAL01830
      ELSE                                                               EAL01840
      END IF                                                             EAL01850
      IF (CARDC.NE.9.) NGOOD=NGOOD+1                                     EAL01860
      NAR=NA                                                             EAL01870
      EAL=0.0                                                            EAL01880
C*********************************************************************EAL01890
C     THE NEXT STATMENT LIMITS THE EALS FOR STEERING AXLE LOADS IN      EAL01900
C     EXCESS OF 22,500 LBS.                                             EAL01910
C*********************************************************************EAL01920
      IF (W(1).GT.22.5) THEN                                             EAL01930
      EAL=EAL+5.39                                                       EAL01940
      ELSE                                                               EAL01950
      EAL=EAL+10.0**(CST1+CST2*ALOG10(W(1))+CST3*ALOG10(W(1))**2)        EAL01960
      END IF                                                             EAL01970
      NAR=NAR-1                                                          EAL01980
      DO 50 I=1,14                                                       EAL01990
      W(I)=W(I+1)                                                        EAL02000
      S(I)=S(I+1)                                                        EAL02010
   50 CONTINUE                                                           EAL02020
  200 CONTINUE                                                           EAL02030
      IF (NAR.EQ.0) GO TO 300                                            EAL02040
C*********************************************************************EAL02050
C     THE NEXT STATMENT LIMITS THE EALS FOR QUAD AXLE GROUPS WITH LOADS EAL02060
```

```
C     IN EXCESS OF 150,000 LBS.                                          EAL02070
C*****************************************************************EAL02080
      IF (S(1).GT.0.0.AND.S(2).GT.0.0.AND.S(3).GT.0.0.AND.(S(1)+S(2)+S(3EAL02090
     +)).LE.18.0.AND.(W(1)+W(2)+W(3)+W(4)).LE.150.0) THEN               EAL02100
      EAL=EAL+10.0**(CQO1+CQO2*ALOG10(W(1)+W(2)+W(3)+W(4))+CQO3*        EAL02110
     +ALOG10(W(1)+W(2)+W(3)+W(4))**2)                                   EAL02120
      ELSE IF (S(1).GT.0.0.AND.S(2).GT.0.0.AND.S(3).GT.0.0.AND.(S(1)+S(2EAL02130
     +)+S(3)).LE.18.0.AND.(W(1)+W(2)+W(3)+W(4)).GT.150.0) THEN          EAL02140
      EAL=EAL+58.3                                                      EAL02150
      NAR=NAR-4                                                         EAL02160
      DO 55 I=1,14                                                      EAL02170
      W(I)=W(I+4)                                                       EAL02180
      S(I)=S(I+4)                                                       EAL02190
   55 CONTINUE                                                          EAL02200
      GO TO 200                                                         EAL02210
      END IF                                                            EAL02220
C*****************************************************************EAL02230
C     THE NEXT STATMENT LIMITS THE EALS FOR TRIDEM AXLE GROUPS WITH     EAL02240
C     LOADS IN EXCESS OF 112,500 LBS.                                   EAL02250
C*****************************************************************EAL02260
      IF (S(1).GT.0.0.AND.S(2).GT.0.0.AND.(S(1)+S(2)).LE.13.0.AND.      EAL02270
     +(W(1)+W(2)+W(3)).LE.112.5) THEN                                   EAL02280
      EAL=EAL+10.0**(CTR1+CTR2*ALOG10(W(1)+W(2)+W(3))+CTR3*             EAL02290
     +ALOG10(W(1)+W(2)+W(3))**2)                                        EAL02300
      ELSE IF (S(1).GT.0.0.AND.S(2).GT.0.0.AND.(S(1)+S(2)).LE.13.0.AND. EAL02310
     +(W(1)+W(2)+W(3)).GT.112.5) THEN                                   EAL02320
      EAL=EAL+59.0                                                      EAL02330
      NAR=NAR-3                                                         EAL02340
      DO 60 I=1,14                                                      EAL02350
      W(I)=W(I+3)                                                       EAL02360
      S(I)=S(I+3)                                                       EAL02370
   60 CONTINUE                                                          EAL02380
      GO TO 200                                                         EAL02390
      END IF                                                            EAL02400
C*****************************************************************EAL02410
C     THE NEXT STATMENT LIMITS THE EALS FOR TANDEM AXLE GROUPS WITH      EAL02420
C     LOADS IN EXCESS OF 75,000 LBS.                                    EAL02430
C*****************************************************************EAL02440
      IF (S(1).GT.0.0.AND.S(1).LE.8.0.AND.(W(1)+W(2)).GT.75.0) THEN     EAL02450
      EAL=EAL+63.0                                                      EAL02460
      ELSE IF (S(1).GT.0.0.AND.S(1).LE.8.0.AND.(W(1)+W(2)).LE.75.0)     EAL02470
     +THEN                                                              EAL02480
      EAL=EAL+10.0**(CTA1+CTA2*ALOG10(W(1)+W(2))+CTA3*ALOG10(W(1)+      EAL02490
     +W(2))**2)                                                         EAL02500
      NAR=NAR-2                                                         EAL02510
      DO 70 I=1,14                                                      EAL02520
      W(I)=W(I+2)                                                       EAL02530
      S(I)=S(I+2)                                                       EAL02540
   70 CONTINUE                                                          EAL02550
      GO TO 200                                                         EAL02560
      END IF                                                            EAL02570
C     IF (W(1).EQ.0.0)  WRITE (7,1170) NN                               EAL02580
C1170 FORMAT (T5,'NN=',I5)                                              EAL02590
C*****************************************************************EAL02600
```

```
C     THE NEXT STATMENT LIMITS THE EALS FOR OTHER SINGLE AXLES WITH     EAL02610
C     LOADS IN EXCESS OF 37,500 LBS.                                    EAL02620
C*****************************************************************EAL02630
      IF (W(1).GT.37.5) THEN                                           EAL02640
      EAL=EAL+78.6                                                     EAL02650
      ELSE                                                             EAL02660
      EAL=EAL+10.0**(CSI1+CSI2*ALOG10(W(1))+CSI3*(ALOG10(W(1))**2))    EAL02670
      END IF                                                           EAL02680
      NAR=NAR-1                                                        EAL02690
      DO 80 I=1,14                                                     EAL02700
      W(I)=W(I+1)                                                      EAL02710
      S(I)=S(I+1)                                                      EAL02720
   80 CONTINUE                                                         EAL02730
      GO TO 200                                                        EAL02740
  300 CONTINUE                                                         EAL02750
C     WRITE (7,9999) EAL,VEH(NCT,NVT),NCT,NVT                          EAL02760
C9999 FORMAT (T5,F8.5,5X,F6.0,5X,'VEH(',I1,',',I2,')')                 EAL02770
      SEAL(NCT,NVT)=SEAL(NCT,NVT)+EAL                                  EAL02780
      AXLE(NCT,NVT)=AXLE(NCT,NVT)+NA                                   EAL02790
      VEH(NCT,NVT)=VEH(NCT,NVT)+1                                      EAL02800
      TWT=TW*1000                                                      EAL02810
      PALOAD=TWT-VT(NVT)                                               EAL02820
      IF (PALOAD.LE.0) GO TO 100                                       EAL02830
      WRITE(7,5555) FC,NVT,EAL,TWT,PALOAD,VT(NVT)                      EAL02840
 5555 FORMAT (T5,I2,T15,I2,T25,F8.4,T35,F8.0,T45,F8.0,T55,F8.0)        EAL02850
      GO TO 100                                                        EAL02860
  888 CONTINUE                                                         EAL02870
      DO 90 J=1,13                                                     EAL02880
      SEAL(3,J)=SEAL(1,J)+SEAL(2,J)                                    EAL02890
      AXLE(3,J)=AXLE(1,J)+AXLE(2,J)                                    EAL02900
      VEH(3,J)=VEH(1,J)+VEH(2,J)                                       EAL02910
   90 CONTINUE                                                         EAL02920
      DO 92 I=1,3                                                      EAL02930
      DO 91 J=1,13                                                     EAL02940
      SEAL(I,14)=SEAL(I,14)+SEAL(I,J)                                  EAL02950
      AXLE(I,14)=AXLE(I,14)+AXLE(I,J)                                  EAL02960
      VEH(I,14)=VEH(I,14)+VEH(I,J)                                     EAL02970
   91 CONTINUE                                                         EAL02980
   92 CONTINUE                                                         EAL02990
      DO 94 I=1,3                                                      EAL03000
      DO 93 J=1,14                                                     EAL03010
      IF (AXLE(I,J).GT.0.) THEN                                        EAL03020
      EALPAXLE(I,J)=SEAL(I,J)/AXLE(I,J)                                EAL03030
      ELSE                                                             EAL03040
      EALPAXLE(I,J)=0.0                                                EAL03050
      ENDIF                                                            EAL03060
      IF (VEH(I,J).GT.0.0) THEN                                        EAL03070
      EALPVEH(I,J)=SEAL(I,J)/VEH(I,J)                                  EAL03080
      ELSE                                                             EAL03090
      EALPVEH(I,J)=0.0                                                 EAL03100
      ENDIF                                                            EAL03110
      IF (VEH(I,J).GT.0.0) THEN                                        EAL03120
      AXLEPVEH(I,J)=AXLE(I,J)/VEH(I,J)                                 EAL03130
      ELSE                                                             EAL03140
```

```
      AXLEPVEH(I,J)=0.0                                                 EAL03150
      ENDIF                                                             EAL03160
   93 CONTINUE                                                          EAL03170
   94 CONTINUE                                                          EAL03180
      WRITE (6,1172) FC,IYR                                             EAL03190
 1172 FORMAT (T5,'FUNCTIONAL CLASS ',I2,5X,'YEAR 19',I2)                EAL03200
C  ********************                                               EAL03210
C             THE NEXT 4 STATEMENTS WERE ADDED BY HFS FOR DEBUGGING    EAL03220
C     072193                                                          EAL03230
C                                                                     EAL03240
C     WRITE (6,1175)                                                  EAL03250
 1175 FORMAT (T5,'NUMBER OF AXLES')                                     EAL03260
      DO 299 J=4,14                                                     EAL03270
      IXLE(1,J)=AXLE(1,J)                                               EAL03280
      IXLE(2,J)=AXLE(2,J)                                               EAL03290
      IXLE(3,J)=AXLE(3,J)                                               EAL03300
  299 WRITE (6,1180) J,IXLE(1,J),IXLE(2,J),IXLE(3,J)                    EAL03310
C **************                                                       EAL03320
      WRITE (6,1173)                                                    EAL03330
 1173 FORMAT (T5,'NUMBER OF VEHICLES WEIGHED')                          EAL03340
      WRITE (6,1174)                                                    EAL03350
 1174 FORMAT (T5,'VEH TYPE',5X,'NON-COAL',5X,'COAL',5X,'ALL')           EAL03360
      DO 95 J=4,14                                                      EAL03370
      IVEH(1,J)=VEH(1,J)                                                EAL03380
      IVEH(2,J)=VEH(2,J)                                                EAL03390
      IVEH(3,J)=VEH(3,J)                                                EAL03400
      WRITE (6,1180) J,IVEH(1,J),IVEH(2,J),IVEH(3,J)                    EAL03410
 1180 FORMAT (T8,I2,T19,I7,T29,I7,T37,I7)                               EAL03420
   95 CONTINUE                                                          EAL03430
      WRITE (6,1200)                                                    EAL03440
 1200 FORMAT (T5,'EALS PER AXLE')                                       EAL03450
      WRITE (6,1174)                                                    EAL03460
      DO 96 J=4,14                                                      EAL03470
      WRITE (6,1210) J,EALPAXLE(1,J),EALPAXLE(2,J),EALPAXLE(3,J)        EAL03480
 1210 FORMAT (T8,I2,T18,F8.5,T27,F8.5,T36,F8.5)                         EAL03490
   96 CONTINUE                                                          EAL03500
      WRITE (6,1300)                                                    EAL03510
 1300 FORMAT (T5,'EALS PER VEHICLE')                                    EAL03520
      WRITE (6,1174)                                                    EAL03530
      DO 97 J=4,14                                                      EAL03540
      WRITE (6,1210) J,EALPVEH(1,J),EALPVEH(2,J),EALPVEH(3,J)           EAL03550
   97 CONTINUE                                                          EAL03560
      WRITE (6,1400)                                                    EAL03570
 1400 FORMAT (T5,'AXLES PER VEHICLE')                                   EAL03580
      WRITE (6,1174)                                                    EAL03590
      DO 98 J=4,14                                                      EAL03600
      WRITE (6,1210) J,AXLEPVEH(1,J),AXLEPVEH(2,J),AXLEPVEH(3,J)        EAL03610
   98 CONTINUE                                                          EAL03620
      WRITE (8,556) NERRORS1                                            EAL03630
      WRITE (8,557) NERRORS2                                            EAL03640
      WRITE (8,558) NERRORS3                                            EAL03650
      WRITE (8,559) NERRORS4                                            EAL03660
      WRITE (8,555) NERRORS5                                            EAL03670
      WRITE (8,560) NGOOD                                               EAL03680
```

```
556 FORMAT (//I6,5X,'EXCESSIVE GROSS WEIGHT(S)')                    EAL03690
557 FORMAT (I6,5X,'NEGLIGIBLE GROSS WEIGHT(S)')                     EAL03700
558 FORMAT (I6,5X,'EXCESSIVE WHEELBASE(S)')                         EAL03710
559 FORMAT (I6,5X,'NEGLIGIBLE WHEELBASE(S)')                        EAL03720
555 FORMAT (I6,5X,'VEHICLE TYPE "921"')                            EAL03730
560 FORMAT (/I6,5X,'SUCCESSFUL ENTRIES')                           EAL03740
    RETURN                                                         EAL03750
    END                                                            EAL03760
```

LOADOMTR SUMMARY PROGRAM

```
$DEBUG
C LOADWTR1.FOR  5/23/96      EALPC\
C  ****************************************************************
C  *  THIS PROGRAM WAS DEVELOPED TO SUMMARIZE WEIGHT DATA IN ORDER
C  *  TO COMBINE THE WEIGHT AND CLASSIFICATION FILES TO FACILITATE
C  *  THE PREDICTION OF TRAFFIC PARAMETERS.
C  *
C  *                                              *
C  ****************************************************************
C
      INTEGER YEAR,CARDC,VTYPE,AXLEW(15),AXLE(15),CARDNO,COMMOD,I,A(50),
     +AW(15,6),YEARC(100),NWEIGH,NCLASS,N,NSTAT,
     +AL(15),NS,WA(15),Q,Q2,Q3,Q4,Q5,NC,M,STAT1(100),Q6,CT,U2,U6,NCLAS1,
     +NWNC(13,6),NWANC(6),NATNC(6,16),NCT(13),NWC(13,6),NWAC(6),CLASS,
     +WNC(13,6,16),WANC(6,16),WC(13,6,16),WAC(6,16),NATC(6,16),C(13),
     +BADSP,BADWT,NOMTCH,NCV
      REAL D(18),WHLBAS,GRWT,ULIMWT,LLIMWT,ULIMSP,LLIMSP
      INTEGER TEST(79)
C     CHARACTER*80 RTMP
      INTEGER FC1(100),FC
      CHARACTER*2 FCX
      CHARACTER*14 IINF
      CHARACTER*12 FN1,FN2
      CHARACTER*3 STAT,STATN
C
C  ************************************************
C  *  INITIALIZE VARIABLES                        *
C  ************************************************
C
      DO 5 I=1,60
      WRITE(*,*)' '
 5    CONTINUE
      OPEN(7,FILE='FILE07', MODE='WRITE')
C     OPEN(8,FILE='FILE08', MODE='WRITE')
C     OPEN(9,FILE='FILE09', MODE='WRITE')
C     OPEN(14,FILE='FILE014', MODE='WRITE')
C
      NOMTCH=0
      NWEIGH=0
      NCLAS1=0
      NCLASS=0
      CT=0
      I=0
      NCV=0.
      NC=0
      BADSP=0
      BADWT=0
      NSTAT=0
C     DATA U2,U6/'0U2','0U6'/
      DO 10 L=1,13
      NCT(L)=0.
```

```
      C(L)=0.
   10 CONTINUE
      DO 20 J=1,6
      NWANC(J)=0.
      NWAC(J)=0.
   20 CONTINUE
      DO 50 L=1,16
      DO 40 K=1,6
      DO 30 J=1,13
      WNC(J,K,L)=0.
      WC(J,K,L)=0.
   30 CONTINUE
   40 CONTINUE
   50 CONTINUE
      DO 70 K=1,6
      DO 60 J=1,13
      NWNC(J,K)=0.
      NWC(J,K)=0.
   60 CONTINUE
   70 CONTINUE
      DO 80 J=1,100
   80 CONTINUE
      DO 100 K=1,16
      DO 90 J=1,6
      WANC(J,K)=0.
      WAC(J,K)=0.
   90 CONTINUE
  100 CONTINUE
C
C  ********************************************************************
C  *  FOLLOWING STATEMENTS   ARE THE READ STATEMENTS THAT ARE FORMAT-  *
C  *  TED TO READ THE NECCESSARY INFORMATION FROM THE LOADOMETER FILE. *
C  ********************************************************************
      WRITE(*,'(A)')' THE INPUT DATA FILE NAME ='
      READ(*,'(A)')IINF
      IIN = 12
      OPEN(IIN,FILE=IINF, STATUS='OLD', MODE='READ')
C
      READ(12,1009)FCX
 1009 FORMAT(T4,A2)
      FN1='LOADFC'//FCX//'.REP'
      FN2='LOADFC'//FCX//'.EAL'
      OPEN(9,FILE=FN1, MODE='WRITE')
      OPEN(14,FILE=FN2, MODE='WRITE')
      REWIND 12
  120 READ(12,1100,END=860) CARDC,FC,STAT,YEAR,VTYPE,COMMOD,(AXLEW(L),
     +L=1,5),(AXLE(L),L=1,4),CARDNO
 1100 FORMAT(I1,T4,I2,T6,A3,T10,I2,T18,I6,T36,I3,T46,9I3,T80,I1)
      IF(CARDC.NE.7) GO TO 120
C  ********************************************************************
C     NEXT 2 STATEMENTS COUNT THE TOTAL NUMBER OF STATIONS
C  ********************************************************************
      IF (STAT.NE.STATN) NSTAT=NSTAT+1
```

```
      STATN=STAT
C  ********************************************************************
C     STATEMENTS 126 AND 127 REJECT FHWA VEHICLE TYPES 999999 IN THE
C     INPUT LOADOMTER DATA. THIS CODE IS ASSIGNED TO A VEHICLE WHICH
C     DOES NOT FIT ANY EXISTING VEHICLE TYPE CONFIGURATION.
C  ********************************************************************
C
      IAT=VTYPE/100000
      IF (IAT.GE.9) GO TO 120
      DO 130 J=5,12
      AXLE(J)=0
  130 CONTINUE
      DO 140 J=6,13
      AXLEW(J)=0
  140 CONTINUE
      IF(CARDNO.EQ.1) READ (12,1150) CARDC,FC,STAT,(AXLEW(L),L=6,13),
     +(AXLE(L),L=5,12),CARDNO
 1150 FORMAT(I1,T4,I2,T6,A3,T29,16I3,T80,I1)
      IF(CARDC.NE.7) GO TO 120
C     IF(CARDNO.EQ.9) WRITE (8,1150) CARDC,FC,STAT,(AXLEW(L),L=6,13),
C    +(AXLE(L),L=5,12),CARDNO
C
C  ********************************************************************
C     NM1=N-1
C     DO 141 J=1,NM1
C     IF (STAT.EQ.STAT1(J)) GO TO 142
C 141 CONTINUE
C     NOMTCH=NOMTCH+1
C     WRITE(7,1108) CARDC,STAT,YEAR,FC,VTYPE,COMMOD,(AXLEW(L),
C    +L=1,5),(AXLE(L),L=1,4),CARDNO,NOMTCH
C1108 FORMAT(I1,T6,A3,T10,I2,T13,I2,T18,I6,T36,I3,T46,9I3,T80,I1,/,
C    +'NO MATCH WITH STATION     #',I5)
C     GO TO 120
C 142 CONTINUE
C  ********************************************************************
C
C
C  ********************************************************************
C  *  INITIALIZE VARIABLES WHICH MUST BE INITIALIZED EACH TIME A NEW  *
C  *  CARD IS READ.                                                   *
C  ********************************************************************
C
      WHLBAS=0
      GRWT=0
      DO 150 L=1,15
      A(L)=0
  150 CONTINUE
      Q2=0
      DO 160 J=1,18
      D(J)=0
  160 CONTINUE
      D(1)=0
      M=1
```

```
      NS=1
      DO 180 K=1,12
      DO 170 J=1,6
      AW(K,J)=0
  170 CONTINUE
  180 CONTINUE
      IA=0
      IB=0
C
C  ******************************************************************
C  *  REJECT DATA FROM 1969 THRU 1972                               *
C  ******************************************************************
C
      IF (YEAR.GE.69.AND.YEAR.LE.72)
     +GO TO 120
C     NM1=N-1
C     DO 190 J=1,NM1
C     IF (STAT.NE.STAT1(J)) GO TO 120
C 190 CONTINUE
C
C  ******************************************************************
C  *  CALCULATE THE DISTANCE FROM EACH AXLE TO THE STEERING AXLE     *
C  ******************************************************************
C
      DO 210 K=2,12
      D(K)=0
      DO 200 J=1,M
      IF (AXLE(J).EQ.0) GO TO 220
      D(K)=D(K)+FLOAT(AXLE(J))/10.
  200 CONTINUE
      M=M+1
  210 CONTINUE
  220 CONTINUE
C
C  ******************************************************************
C  *  CALCULATE THE NUMBER OF AXLES ON THE VEHICLE                  *
C  ******************************************************************
C
      DO 230 J=2,12
      IF(D(J)-D(J-1).GT.0) NS=NS+1
  230 CONTINUE
C
C  ******************************************************************
C  *  CONVERT THE VEHICLE TYPE IN THE LOADOMETER DATA TO A VEHICLE TYPE*
C  *  COMPATIABLE WITH CLASSIFICATION DATA IF THE CONVERSION CANNOT BE *
C  *  MADE IT WILL BE ACCOUNTED FOR IN "NCLASS" AND THE NEXT CARD WILL *
C  *  BE READ.                                                      *
C  ******************************************************************
C
      IA=VTYPE/100000
      IB=VTYPE/10000-IA*10
      GO TO (240,250,260,260,270,270,270,270),IA
      GO TO 280
```

```
 240 I=4
     GO TO 290
 250 IF(IB.LE.1) I=3
     IF(IB.EQ.2) I=5
     IF(IB.EQ.3) I=6
     IF(IB.GE.4) I=7
     GO TO 290
 260 IF(NS.LE.4) I=8
     IF(NS.EQ.5) I=9
     IF(NS.GE.6) I=10
     GO TO 290
 270 IF(NS.LE.5) I=11
     IF(NS.EQ.6) I=12
     IF(NS.GE.7) I=13
     GO TO 290
 280 NCLASS=NCLASS+1
     WRITE(7,1105) CARDC,STAT,YEAR,FC,VTYPE,COMMOD,(AXLEW(L),
    +L=1,5),(AXLE(L),L=1,4),CARDNO,NCLASS,I
1105 FORMAT(I1,T6,A3,T10,I2,T13,I2,T18,I6,T36,I3,T46,9I3,T80,I1,/,
    +'NCLASS',I6,2X,'NO. OF AXLES=',I2)
     GO TO 120
 290 CONTINUE
C  ********************************************************************
C  *  CHECK THE WHEELBASEAND THE GROSS WEIGHT AGAINST PRESELECTED     *
C  *  LIMITS                                                          *
C  *                                                                  *
C  *  A DECISION WAS MADE 10/13/87 TO INCREASE THE UPPER WEIGHT LIMIT  *
C  *  BY 10% TO INCLUDE A GREATER PORTION OF HEAVY COAL TRUCKS.     DHC*
C  *                                                                  *
C  ********************************************************************
     IF (I-5) 291,292,292
 291 ULIMWT=1.1*(15.*(FLOAT(I)-5.)+60.)
     LLIMWT=.5*(FLOAT(I)-5.)+2.
     GO TO 293
 292 ULIMWT=1.1*(15.*(FLOAT(I)-4.)+60.)
     LLIMWT=.5*(FLOAT(I)-4.)+2.
 293 CONTINUE
     IF(ULIMWT.GT.180.) ULIMWT=180.
     IF(LLIMWT.GT.26.) LLIMWT=26.
     DO 300 J=1,NS
     GRWT=GRWT+FLOAT(AXLEW(J))/10.
 300 CONTINUE
     IF(GRWT.LE.ULIMWT.AND.GRWT.GE.LLIMWT) GO TO 310
     BADWT=BADWT+1
     WRITE(7,1106) CARDC,STAT,YEAR,FC,VTYPE,COMMOD,(AXLEW(L),
    +L=1,5),(AXLE(L),L=1,4),CARDNO,BADWT,I,ULIMWT,LLIMWT,GRWT
1106 FORMAT(I1,T6,A3,T10,I2,T13,I2,T18,I6,T36,I3,T46,9I3,T80,I1,/,
    +'BADWT',I6,2X,'NO OF AXLES=',I2,2X,'ULIMWT=',F5.1,2X,
    +'LLIMWT=',F5.1,2X,'GRWT=',F5.1)
     GO TO 120
 310 CONTINUE
     IF (I-5) 311,312,312
 311 ULIMSP=6.*(FLOAT(I)-5.)+40.
```

```
      LLIMSP=1.1538*(FLOAT(I)-5)+2.
      GO TO 313
  312 ULIMSP=6.*(FLOAT(I)-4.)+40.
      LLIMSP=1.1538*(FLOAT(I)-4)+2.
  313 CONTINUE
      IF(ULIMSP.GT.90.) ULIMSP=90.
      WHLBAS=D(NS)
      IF(WHLBAS.LE.ULIMSP.AND.WHLBAS.GE.LLIMSP) GO TO 320
      BADSP=BADSP+1
      WRITE(7,1107) CARDC,STAT,YEAR,FC,VTYPE,COMMOD,(AXLEW(L),
     +L=1,5),(AXLE(L),L=1,4),CARDNO,BADSP,I,ULIMSP,LLIMSP,WHLBAS
 1107 FORMAT(I1,T6,A3,T10,I2,T13,I2,T18,I6,T36,I3,T46,9I3,T80,I1,/,
     +'BADSP',I6,2X,'NO OF AXLES=',I2,2X,'ULIMSP=',F5.1,2X,
     +'LLIMSP=',F5.1,2X,'WHLBAS=',F5.1)
      GO TO 120
  320 CONTINUE
C
C  *********************************************************************
C  *  FOLLOWING  STATEMENTS  WILL  DETECT ERRORS WHEN A VEHICLE HAS    *
C  *  MORE OR LESS AXLES THAN SHOULD TYPICALLY BE ON THE VEHICLE TYPE. *
C  *  IF AN ERROR IS DETECTED IT WILL BE ACCOUNTED FOR IN "NCLAS1"     *
C  *  AND THE NEXT CARD WILL BE READ.                                  *
C  *********************************************************************
C
      IF ((I.EQ.3.OR.I.EQ.5).AND.NS.NE.2) GO TO 330
      IF (I.EQ.6.AND.NS.NE.3) GO TO 330
      IF (I.EQ.7.AND.NS.LT.4) GO TO 330
      GO TO 340
  330 NCLAS1=NCLAS1+1
      WRITE(7,1109) CARDC,STAT,YEAR,FC,VTYPE,COMMOD,(AXLEW(L),
     +L=1,5),(AXLE(L),L=1,4),CARDNO,NCLAS1,I,NS
 1109 FORMAT(I1,T6,A3,T10,I2,T13,I2,T18,I6,T36,I3,T46,9I3,T80,I1,/,
     +'NCLAS1',I6,2X,'REQ. NO OF AXLES=',I2,2X,'NO OF AXLES=',I4)
      GO TO 120
C
C  *********************************************************************
C  *  CALCULATE THE LOAD CATEGORY FOR ALL AXLES TREATED AS SINGLE AXLES*
C  *********************************************************************
C
  340 DO 350 J=1,NS
      AW(J,6)=((AXLEW(J)/25.01)+1)
      IF (AW(J,6).GT.16) AW(J,6)=16
  350 CONTINUE
C
C  *********************************************************************
C  *  FIRST DETERMINE IF THE VEHICLE IS  COAL-HAULING  OR  NON-COAL-   *
C  *  HAULING THEN FOR COAL-HAULING VEHICLES DATA  *
C  *  FROM ALL RURAL STATIONS WILL BE USED AND FOR NON-COAL-HAULING    *
C  *  VEHICLES ONLY DATA FROM MAIN RURAL STATIONS IS TO BE CONSIDERED. *
C  *********************************************************************
C
      IF(YEAR.GE.69.AND.YEAR.LE.72.AND.(COMMOD.LT.110.OR.COMMOD.GT.112))
     +GO TO 370
```

```
      IF(YEAR.GE.69.AND.YEAR.LE.72.AND.COMMOD.GE.110.AND.COMMOD.LE.112)
     +GO TO 440
      IF(YEAR.GE.69.AND.YEAR.LE.72) GO TO 120
C     NM1=N-1
      IF(COMMOD.GE.110.AND.COMMOD.LE.112) GO TO 440
      GO TO 370
  370 DO 390 K=1,16
      DO 380 J=1,NS
      IF(AW(J,6).EQ.K) WNC(I,6,K)=WNC(I,6,K)+1
  380 CONTINUE
  390 CONTINUE
      DO 400 L=1,NS
      NWNC(I,6)=NWNC(I,6)+1
  400 CONTINUE
      IF(I.EQ.3) GO TO 520
      DO 420 K=1,16
      DO 410 L=1,NS
      IF(AW(L,6).EQ.K) WANC(6,K)=WANC(6,K)+1
  410 CONTINUE
  420 CONTINUE
      DO 430 L=1,NS
      NWANC(6)=NWANC(6)+1
  430 CONTINUE
      GO TO 520
  440 DO 460 L=1,NS
      DO 450 K=1,16
      IF(AW(L,6).EQ.K) WC(I,6,K)=WC(I,6,K)+1
  450 CONTINUE
  460 CONTINUE
      DO 470 L=1,NS
      NWC(I,6)=NWC(I,6)+1
  470 CONTINUE
      IF(I.EQ.3) GO TO 520
  480 DO 500 K=1,16
      DO 490 L=1,NS
      IF(AW(L,6).EQ.K) WAC(6,K)=WAC(6,K)+1
  490 CONTINUE
  500 CONTINUE
      DO 510 L=1,NS
      NWAC(6)=NWAC(6)+1
  510 CONTINUE
C
C  *****************************************************************
C  *  REASSIGN AXLELOADS AND CUMULATIVE DISTANCE (1 TO X) TO POSITIONS *
C  *  (4 TO X+3) IN ORDER TO AVOID NEGATIVE SUBSCRIPTS WHEN DETERMINING*
C  *  AXLE TYPES                                                   *
C  *****************************************************************
C
  520 DO 530 J=1,12
      AXLEW(16-J)=AXLEW(13-J)
      AXLE(16-J)=AXLE(13-J)
      D(16-J)=D(13-J)
  530 CONTINUE
```

```
      DO 540 J=1,3
      D(J)=-200
      AXLEW(J)=0
  540 CONTINUE
      Q3=NS+4
      DO 550 J=Q3,15
      D(J)=200
      AXLEW(J)=0
  550 CONTINUE
C
C  *********************************************************************
C  *  ASSIGN AXLE  TYPES (SINGLE,  TANDEM,  ETC.)  TO  A  VEHICLE  IN  *
C  *  ACCORDANCE WITH AXLE SPACINGS.                                   *
C  *********************************************************************
C
      Q5=NS+3
      DO 560J=5,Q5
      A(J)=2
      IF(D(J)-D(J-1).LE.5.OR.D(J+1)-D(J).LE.5) A(J)=3
      IF(D(J)-D(J-2).LE.10.OR.D(J+1)-D(J-1).LE.10.OR.D(J+2)-D(J).LE.
     +10) A(J)=4
      IF(D(J)-D(J-3).LE.15.OR.D(J+1)-D(J-2).LE.15.OR.D(J+2)-D(J-1).LE.
     +15.OR.D(J+3)-D(J).LE.15) A(J)=5
  560 CONTINUE
      NC=1
      A(4)=1
C
C  *********************************************************************
C  *  CALCULATE THE LOADS ON EACH AXLE TYPE.                           *
C  *********************************************************************
C
      WA(4)=AXLEW(4)
      Q=5
      Q4=NS+4
      DO 580 K=5,Q4
      WA(K)=0
      Q2=Q+A(Q)-2
      DO 570 J=Q,Q2
      WA(K)=WA(K)+AXLEW(J)
  570 CONTINUE
      A(K)=A(Q)
      NC=NC+1
      Q=Q+A(Q)-1
      IF(D(Q)-D(Q-1).GT.60) GO TO 590
  580 CONTINUE
  590 CONTINUE
      DO 600 J=Q,15
      WA(J)=0
  600 CONTINUE
C
C  *********************************************************************
C  *  REASSIGN AXLELOADS AND AXLE TYPES TO POSITION (1 TO X) IN ORDER  *
C  *  TO CONTINUE PROCCESSING.                                         *
```

```
C  *********************************************************************
C
      DO 610 J=1,NC
      A(J)=A(J+3)
      WA(J)=WA(J+3)
  610 CONTINUE
C
C  *********************************************************************
C  *  ASSIGN LOAD CATEGORIES TO EACH AXLE TYPE.                        *
C  *********************************************************************
C
      DO 620 J=1,NC
      IF(A(J).EQ.1) AW(J,1)=((WA(J)/15.01)+1)
      IF (AW(J,1).GT.16) AW(J,1)=16
      IF(A(J).EQ.2) AW(J,2)=((WA(J)/25.01)+1)
      IF (AW(J,2).GT.16) AW(J,2)=16
      IF(A(J).EQ.3) AW(J,3)=((WA(J)/50.01)+1)
      IF (AW(J,3).GT.16) AW(J,3)=16
      IF(A(J).EQ.4) AW(J,4)=((WA(J)/75.01)+1)
      IF (AW(J,4).GT.16) AW(J,4)=16
      IF(A(J).EQ.5) AW(J,5)=((WA(J)/100.01)+1)
      IF (AW(J,5).GT.16) AW(J,5)=16
  620 CONTINUE
C
C  *********************************************************************
C  *  FOLLOWING  STATEMENTS  FIRST  DETERMINE IF THE VEHICLE IS COAL-  *
C  *  HAULING OR NON-COAL-HAULING THEN FOR COAL-HAULING VEHICLES DATA  *
C  *  FROM ALL RURAL STATIONS WILL BE USED AND FOR NON-COAL-HAULING    *
C  *  VEHICLES ONLY DATA FROM MAIN RURAL STATIONS IS TO BE CONSIDERED. *
C  *********************************************************************
C
      IF(YEAR.GE.69.AND.YEAR.LE.72.AND.
     +(COMMOD.LT.110.OR.COMMOD.GT.112)) GO TO 640
      IF(YEAR.GE.69.AND.YEAR.LE.72.AND.
     +(COMMOD.GE.110.AND.COMMOD.LE.112))GO TO 750
C     NM1=N-1
C     DO 630 J=1,NM1
C     IF(STAT.EQ.STAT1(J)) GO TO 530
      IF(COMMOD.LT.110.OR.COMMOD.GT.112) GO TO 640
      IF(COMMOD.GE.110.AND.COMMOD.LE.112)GO TO 750
C 530 CONTINUE
C     NOMTCH=NOMTCH+1
C     WRITE(7,1110) CARDC,STAT,YEAR,FC,VTYPE,COMMOD,(AXLEW(L),
C    +L=1,5),(AXLE(L),L=1,4),CARDNO,NOMTCH
C1110 FORMAT(I1,T6,A3,T10,I2,T13,I2,T18,I6,T36,I3,T46,9I3,T80,I1,/,
C    +'NMTCH',I6)
      GO TO 120
  640 DO 670 L=1,NC
      DO 660 K=1,16
      DO 650 J=1,5
      IF(A(L).EQ.J.AND.AW(L,J).EQ.K) WNC(I,J,K)=WNC(I,J,K)+1
  650 CONTINUE
  660 CONTINUE
```

```
  670 CONTINUE
      DO 690 J=1,5
      DO 680 K=1,NC
      IF(A(K).EQ.J) NWNC(I,J)=NWNC(I,J)+1
  680 CONTINUE
  690 CONTINUE
      IF(I.EQ.3) GO TO 120
      DO 720 J=1,5
      DO 710 K=1,16
      DO 700 L=1,NC
      IF(A(L).EQ.J.AND.AW(L,J).EQ.K) WANC(J,K)=WANC(J,K)+1
  700 CONTINUE
  710 CONTINUE
  720 CONTINUE
      DO 740 J=1,5
      DO 730 L=1,NC
      IF(A(L).EQ.J) NWANC(J)=NWANC(J)+1
  730 CONTINUE
  740 CONTINUE
      GO TO 120
  750 DO 780 J=1,5
      DO 770 K=1,16
      DO 760 L=1,NC
      IF(A(L).EQ.J.AND.AW(L,J).EQ.K) WC(I,J,K)=WC(I,J,K)+1
  760 CONTINUE
  770 CONTINUE
  780 CONTINUE
      DO 800 J=1,5
      DO 790 L=1,NC
      IF(A(L).EQ.J) NWC(I,J)=NWC(I,J)+1
  790 CONTINUE
  800 CONTINUE
      C(I)=C(I)+1
      IF(I.EQ.3) GO TO 120
      DO 830 J=1,5
      DO 820 K=1,16
      DO 810 L=1,NS
      IF(A(L).EQ.J.AND.AW(L,J).EQ.K) WAC(J,K)=WAC(J,K)+1.
  810 CONTINUE
  820 CONTINUE
  830 CONTINUE
      DO 850 J=1,5
      DO 840 L=1,NC
      IF(A(L).EQ.J) NWAC(J)=NWAC(J)+1
  840 CONTINUE
  850 CONTINUE
      GO TO 120
  860 CONTINUE
C
C  ****************************************************************
C  *  CALCULATES THE NUMBER OF VEHICLES WHICH WERE PROCCESSED AND    *
C  *  WRITE THE OUTPUT TO DISK.                                     *
C  ****************************************************************
```

```
C
      CLASS = 0
      DO 870 I=3,13
      CLASS = CLASS + NWNC(I,1) + NWC(I,1)
      WRITE(14,1200) YEAR,FC,I,(NWNC(I,J),J=1,6),NSTAT
 1200 FORMAT(I2,T3,'NWNC',T8,I2,T13,I2,T15,6I6,T70,I7)
  870 CONTINUE
      WRITE(14,1250) YEAR,FC,(NWANC(J),J=1,6),NSTAT
 1250 FORMAT(I2,T3,'NWANC',T8,I2,T15,6I6,T70,I7)
      DO 880 I=3,13
      WRITE(14,1300) YEAR,FC,I,(NWC(I,J),J=1,6),NSTAT
 1300 FORMAT(I2,T3,'NWC',T8,I2,T13,I2,6I6,T70,I7)
  880 CONTINUE
      WRITE(14,1350) YEAR,FC,(NWAC(J),J=1,6),NSTAT
 1350 FORMAT(I2,T3,'NWAC',T8,I2,T15,6I6,T70,I7)
      DO 900 I=3,13
      DO 890 J=1,6
      WRITE(14,1400) YEAR,FC,I,J,(WNC(I,J,K),K=1,8),NSTAT
 1400 FORMAT(I2,T3,'WNC',T8,I2,T11,2I2,8I6,T70,I7)
      WRITE(14,1400) YEAR,FC,I,J,(WNC(I,J,K),K=9,16),NSTAT
  890 CONTINUE
  900 CONTINUE
      DO 910 J=1,6
      WRITE(14,1450) YEAR,FC,J,(WANC(J,K),K=1,8),NSTAT
 1450 FORMAT(I2,T3,'WANC',T8,I2,T13,I2,8I6,T70,I4)
      WRITE(14,1450) YEAR,FC,J,(WANC(J,K),K=9,16),NSTAT
  910 CONTINUE
      DO 930 I=3,13
      DO 920 J=1,6
      WRITE(14,1500) YEAR,FC,I,J,(WC(I,J,K),K=1,8),NSTAT
 1500 FORMAT(I2,T3,'WC',T8,I2,T11,2I2,T15,8I6,T70,I4)
      WRITE(14,1500) YEAR,FC,I,J,(WC(I,J,K),K=9,16),NSTAT
  920 CONTINUE
  930 CONTINUE
      DO 940 J=1,6
      WRITE(14,1550) YEAR,FC,J,(WAC(J,K),K=1,8),NSTAT
 1550 FORMAT(I2,T3,'WAC',T8,I2,T13,I2,8I6,T70,I4)
      WRITE(14,1550) YEAR,FC,J,(WAC(J,K),K=9,16),NSTAT
  940 CONTINUE
      WRITE(14,1600) YEAR,FC,(C(I),I=1,11),NSTAT
 1600 FORMAT(I2,T3,'C',T8,I2,T15,11I6,T70,I4)
      WRITE(14,1600) YEAR,FC,(C(I),I=12,13),NSTAT
C
C  ****************************************************************
C  *  OUTPUT WILL BE RECIEVED AND CHECKED FOR ANY GROSS ERRORS IN THE  *
C  *  CALCULATIONS OR AN UNACCEPTABLE AMOUNT OF VEHICLES WHICH WERE    *
C  *  REJECTED.                                                        *
C  ****************************************************************
C
      WRITE(9,1650) YEAR,FC
 1650 FORMAT('1',T21,'STATEWIDE AXLE AND AXLELOAD DISTRIBUTIONS FOR 19',
     +I2,4X,'FUNCTIONAL CLASS',I2)
      WRITE(9,4550)
```

```
      WRITE(9,4550)
      WRITE(9,1700)
 1700 FORMAT(T50,'VARIABLE NAMES AND CODES DEFINED')
      WRITE(9,4550)
      WRITE(9,4550)
      WRITE(9,1750)
 1750 FORMAT(1X,'NWNC  - NUMBER OF AXLES WEIGHED AT ALL MAIN RURAL STATI
     +ONS FOR NON-COAL-HAULING VEHICLES CLASSIFIED BY VEHICLE TYPE AND A
     +XLE TYPE')
      WRITE(9,1800)
 1800 FORMAT(1X,'NWANC - NUMBER OF AXLES WEIGHED AT ALL MAIN RURAL STATI
     +ONS FOR NON-COAL-HAULING VEHICLES OF TYPES 6-13 CLASSIFIED BY AXLE
     + TYPE')
      WRITE(9,1850)
 1850 FORMAT(1X,'NWC   - NUMBER OF AXLES WEIGHED AT ALL RURAL STATIONS F
     +OR COAL-HAULING VEHICLES CLASSIFIED BY VEHICLE TYPE AND AXLE TYPE'
     +)
      WRITE(9,1900)
 1900 FORMAT(1X,'NWAC  - NUMBER OF AXLES WEIGHED AT ALL RURAL STATIONS F
     +OR COAL-HAULING VEHICLES OF TYPES 6-13 CLASSIFIED BY AXLE TYPE')
      WRITE(9,1950)
 1950 FORMAT(1X,'WNC   - NUMBER OF AXLES WEIGHED AT ALL MAIN RURAL STATI
     +ONS FOR NON-COAL-HAULING VEHICLES CLASSIFIED BY VEHICLE TYPE, AXLE
     + TYPE AND')
      WRITE(9,2000)
 2000 FORMAT(9X,'LOAD INTERVAL')
      WRITE(9,2050)
 2050 FORMAT(1X,'WANC  - NUMBER OF AXLES WEIGHED AT ALL MAIN RURAL STATI
     +ONS FOR NON-COAL-HAULING VEHICLES OF TYPES 6-13 CLASSIFIED BY AXLE
     + TYPE AND')
      WRITE(9,2000)
      WRITE(9,2100)
 2100 FORMAT(1X,'WC    - NUMBER OF AXLES WEIGHED AT ALL RURAL STATIONS F
     +OR COAL-HAULING VEHICLES CLASSIFIED BY VEHICLE TYPE, AXLE TYPE AND
     +')
      WRITE(9,2000)
      WRITE(9,2150)
 2150 FORMAT(' ','WAC   - NUMBER OF AXLES WEIGHED AT ALL RURAL STATIONS
     +FOR COAL-HAULING VEHICLES OF TYPES 6-13 CLASSIFIED BY AXLE TYPE AN
     +D')
      WRITE(9,2000)
      WRITE(9,2200)
 2200 FORMAT(1X,'C     - NUMBER OF COAL HAULING VEHICLES WEIGHED AT ALL
     +RURAL STATIONS CLASSIFIED BY VEHICLE TYPE')
      WRITE(9,4550)
      WRITE(9,4550)
      WRITE(9,2250)
 2250 FORMAT(T50,'VEHICLE CODES (VTYPE)')
      WRITE(9,4550)
      WRITE(9,4550)
      WRITE(9,2300)
 2300 FORMAT(2X,'3 - 2 AXLE, 4-TIRE VEHICLES OTHER THAN PASSENGER CARS')
      WRITE(9,2400)
```

```
2400 FORMAT(2X,'4 - ALL BUSSES')
     WRITE(9,2450)
2450 FORMAT(2X,'5 - 2 AXLE, 6-TIRE')
     WRITE(9,2500)
2500 FORMAT(2X,'6 - 3 AXLE SINGLE UNIT')
     WRITE(9,2550)
2550 FORMAT(2X,'7 - 4 OR MORE AXLE SINGLE UNIT')
     WRITE(9,2600)
2600 FORMAT(2X,'8 - 4 OR LESS AXLE SINGLE TRAILER')
     WRITE(9,2650)
2650 FORMAT(2X,'9 - 5 AXLE SINGLE TRAILER')
     WRITE(9,2700)
2700 FORMAT(1X,'10 - 6 OR MORE AXLE SINGLE TRAILER')
     WRITE(9,2750)
2750 FORMAT(1X,'11 - 5 OR LESS AXLE MULTI-TRAILER')
     WRITE(9,2800)
2800 FORMAT(1X,'12 - 6 AXLE MULTIPLE TRAILER')
     WRITE(9,2850)
2850 FORMAT(1X,'13 - 7 OR MORE AXLE MULTIPLE TRAILER')
     WRITE(9,4550)
     WRITE(9,4550)
     WRITE(9,2900)
2900 FORMAT(T50,'AXLE TYPE CODES (AXTYPE)')
     WRITE(9,4550)
     WRITE(9,4550)
     WRITE(9,2950)
2950 FORMAT(1X,'1 - STEERING AXLES')
     WRITE(9,3000)
3000 FORMAT(1X,'2 - OTHER SINGLE AXLES')
     WRITE(9,3050)
3050 FORMAT(1X,'3 - TANDEM AXLES (5 FT. MAXIMUM SPAN)')
     WRITE(9,3100)
3100 FORMAT(1X,'4 - TRIDEM AXLES (10 FT. MAXIMUM SPAN)')
     WRITE(9,3150)
3150 FORMAT(1X,'5 - QUAD AXLES (15 FT. MAXIMUM SPAN)')
     WRITE(9,3200)
3200 FORMAT(1X,'6 - ALL AXLES TREATED AS SINGLES')
     WRITE(9,4550)
     WRITE(9,4550)
     WRITE(9,3250)
3250 FORMAT('0',T36,'AXLE LOAD INTERVALS')
     WRITE(9,3300)
3300 FORMAT('0',T39,'(1000 POUNDS)')
     WRITE(9,3350)
3350 FORMAT(T4,'LOAD',T24,'OTHER',T85,'ALL')
     WRITE(9,3400)
3400 FORMAT(T2,'INTERVAL',T11,'STEERING',T23,'SINGLE',T36,'TANDEM',T50,
    +'TRIDEM',T70,'QUAD',T83,'SINGLE')
     WRITE(9,3450)
3450 FORMAT('0',T5,'1',T11,'0.0- 1.5',T23,'0.0- 2.5',T35,'0.0- 5.0',
    +T49,'0.0-  7.5',T68,'0.0- 10.0',T83,'0.0- 2.5')
     WRITE(9,3500)
3500 FORMAT(T5,'2',T11,'1.6- 3.0',T23,'2.6- 5.0',T35,'5.1-10.0',
```

```
     +T49,'7.6- 15.0',T67,'10.1- 20.0',T83,'2.6- 5.0')
      WRITE(9,3550)
3550 FORMAT(T5,'3',T11,'3.1- 4.5',T23,'5.1- 7.5',T34,'10.1-15.0',
     +T48,'15.1- 22.5',T67,'20.1- 30.0',T83,'5.1- 7.5')
      WRITE(9,3600)
3600 FORMAT(T5,'4',T11,'4.6- 6.0',T23,'7.6-10.0',T34,'15.1-20.0',
     +T48,'22.6- 30.0',T67,'30.1- 40.0',T83,'7.6-10.0')
      WRITE(9,3650)
3650 FORMAT(T5,'5',T11,'6.1- 7.5',T22,'10.1-12.5',T34,'20.1-25.0',
     +T48,'30.1- 37.5',T67,'40.1- 50.0',T82,'10.1-12.5')
      WRITE(9,3700)
3700 FORMAT(T5,'6',T11,'7.6- 9.0',T22,'12.6-15.0',T34,'25.1-30.0',
     +T48,'37.6- 45.0',T67,'50.1- 60.0',T82,'12.6-15.0')
      WRITE(9,3750)
3750 FORMAT(T5,'7',T11,'9.1-10.5',T22,'15.1-17.5',T34,'30.1-35.0',
     +T48,'45.1- 52.5',T67,'60.1- 70.0',T82,'15.1-17.5')
      WRITE(9,3800)
3800 FORMAT(T5,'8',T10,'10.6-12.0',T22,'17.6-20.0',T34,'35.1-40.0',
     +T48,'52.6- 60.0',T67,'70.1- 80.0',T82,'17.6-20.0')
      WRITE(9,3850)
3850 FORMAT(T5,'9',T10,'12.1-13.5',T22,'20.1-22.5',T34,'40.1-45.0',
     +T48,'60.1- 67.5',T67,'80.1- 90.0',T82,'20.1-22.5')
      WRITE(9,3900)
3900 FORMAT(T4,'10',T10,'13.6-15.0',T22,'22.6-25.0',T34,'45.1-50.0',
     +T48,'67.6- 75.0',T67,'90.1-100.0',T82,'22.6-25.0')
      WRITE(9,3950)
3950 FORMAT(T4,'11',T10,'15.1-16.5',T22,'25.1-27.5',T34,'50.1-55.0',
     +T48,'75.1- 82.5',T66,'100.1-110.0',T82,'25.1-27.5')
      WRITE(9,4000)
4000 FORMAT(T4,'12',T10,'16.6-18.0',T22,'27.6-30.0',T34,'55.1-60.0',
     +T48,'82.6- 90.0',T66,'110.1-120.0',T82,'27.6-30.0')
      WRITE(9,4050)
4050 FORMAT(T4,'13',T10,'18.1-19.5',T22,'30.1-32.5',T34,'60.1-65.0',
     +T48,'90.1- 97.5',T66,'120.1-130.0',T82,'30.1-32.5')
      WRITE(9,4100)
4100 FORMAT(T4,'14',T10,'19.6-21.0',T22,'32.6-35.0',T34,'65.1-70.0',
     +T48,'97.6-105.0',T66,'130.1-140.0',T82,'32.6-35.0')
      WRITE(9,4150)
4150 FORMAT(T4,'15',T10,'21.1-22.5',T22,'35.1-37.5',T34,'70.1-75.0',
     +T47,'105.1-112.5',T66,'140.1-150.0',T82,'35.1-37.5')
      WRITE(9,4200)
4200 FORMAT(T4,'16',T10,'OVER 22.5',T22,'OVER 37.5',T34,'OVER 75.0',
     +T48,'OVER 112.5',T67,'OVER 150.0',T82,'OVER 37.5')
      WRITE(9,4550)
      WRITE(9,4550)
      WRITE(9,4250) NOMTCH
4250 FORMAT('0','NUMBER OF VEHICLES REJECTED DUE TO INABILITY TO MATCH
     +STATION WITH STATION ON THE HEADER CARD',I7)
      WRITE(9,4300) BADWT
4300 FORMAT('0','NUMBER OF VEHICLES REJECTED DUE TO UNACCEPTABLE GROSS
     +WEIGHT',34X,I6)
      WRITE(9,4350) BADSP
4350 FORMAT('0','NUMBER OF VEHICLES REJECTED DUE TO UNACCEPTABLE WHEEL
```

```
     +BASE',36X,I6)
C
      WRITE(9,4400) NCLASS
 4400 FORMAT('0','NUMBER OF VEHICLES REJECTED DUE TO UNINTERPRETABLE VEH
     +ICLE TYPE',30X,I7)
      WRITE(9,4450) NCLAS1
 4450 FORMAT('0','NUMBER SINGLE UNIT OF VEHICLES REJECTED DUE TO INAPPRO
     +PRIATE NUMBER OF AXLES',18X,I6)
      WRITE(9,4500) CLASS
 4500 FORMAT('0','NUMBER OF TYPE 3-13 VEHICLES PROCESSED',T92,I6)
      WRITE(9,4550)
 4550 FORMAT('0')
      WRITE(9,4550)
      WRITE(9,4600)
 4600 FORMAT('0',T30,'AXLE TYPE')
      WRITE(9,4650)
 4650 FORMAT(T10,'VTYPE',5X,'1',5X,'2',5X,'3',5X,'4',5X,'5',5X,'6')
      DO 950 I=3,13
      WRITE(9,4700) I,(NWNC(I,J),J=1,6)
 4700 FORMAT(T3,'NWNC',T13,I2,T15,6I6)
  950 CONTINUE
      WRITE(9,4625)
 4625 FORMAT('1',T30,'AXLE TYPE')
      WRITE(9,4850)
      WRITE(9,4750) (NWANC(J),J=1,6)
 4750 FORMAT(T3,'NWANC',T15,6I6)
      WRITE(9,4550)
      WRITE(9,4600)
      WRITE(9,4650)
      DO 960 I=3,13
      WRITE(9,4800) I,(NWC(I,J),J=1,6)
 4800 FORMAT(T3,'NWC',T13,I2,6I6)
  960 CONTINUE
      WRITE(9,4550)
      WRITE(9,4600)
      WRITE(9,4850)
 4850 FORMAT('0',T20,'1',5X,'2',5X,'3',5X,'4',5X,'5',5X,'6')
      WRITE(9,4900) (NWAC(J),J=1,6)
 4900 FORMAT(T3,'NWAC',T15,6I6)
      WRITE(9,4550)
      WRITE(9,4950)
 4950 FORMAT('0',T12,'V',1X,'A')
      WRITE(9,5000)
 5000 FORMAT(T12,'T',1X,'X')
      WRITE(9,5050)
 5050 FORMAT(T12,'Y',1X,'T')
      WRITE(9,5100)
 5100 FORMAT(T12,'P',1X,'Y',T60,'LOAD CATEGORY')
      WRITE(9,5150)
 5150 FORMAT(T12,'E',1X,'P',5X,'1',5X,'2',5X,'3',5X,'4',5X,'5',5X,
     +'6',5X,'7',5X,'8',5X,'9',4X,'10',4X,'11',4X,'12',4X,'13',4X,'14',
     +4X,'15',4X,'16')
      DO 980 I=3,13
```

```
      DO 970 J=1,6
      WRITE(9,5200) I,J,(WNC(I,J,K),K=1,16)
 5200 FORMAT(T3,'WNC',T11,2I2,16I6)
  970 CONTINUE
  980 CONTINUE
      WRITE(9,4550)
      WRITE(9,5250)
 5250 FORMAT(T60,'LOAD CATEGORY')
      WRITE(9,5300)
 5300 FORMAT('0',T9,'AXTYPE',5X,'1',5X,'2',5X,'3',5X,'4',5X,'5',5X,'6'
     +,5X,'7',5X,'8',5X,'9',4X,'10',4X,'11',4X,'12',4X,'13',4X,'14',
     +4X,'15',4X,'16')
      DO 990 J=1,6
      WRITE(9,5350) J,(WANC(J,K),K=1,16)
 5350 FORMAT(T3,'WANC',T13,I2,16I6)
  990 CONTINUE
      WRITE(9,4550)
      WRITE(9,4950)
      WRITE(9,5000)
      WRITE(9,5050)
      WRITE(9,5100)
      WRITE(9,5150)
      DO 1020 I=3,13
      DO 1010 J=1,6
      WRITE(9,5400) I,J,(WC(I,J,K),K=1,16)
 5400 FORMAT(T3,'WC',T11,2I2,T15,16I6)
 1010 CONTINUE
 1020 CONTINUE
      WRITE(9,4550)
      WRITE(9,5250)
      WRITE(9,5300)
      DO 1030 J=1,6
      WRITE(9,5450) J,(WAC(J,K),K=1,16)
 5450 FORMAT(T3,'WAC',T13,I2,16I6)
 1030 CONTINUE
      WRITE(9,4550)
      WRITE(9,5500)
      WRITE(9,5600)
      WRITE(9,5550) (C(I),I=1,13)
 5500 FORMAT('0',T60,'VTYPE')
 5550 FORMAT(T3,'C',T15,19I6)
 5600 FORMAT('0',T20,'1',5X,'2',5X,'3',5X,'4',5X,'5',5X,'6'
     +,5X,'7',5X,'8',5X,'9',4X,'10',4X,'11',4X,'12',4X,'13',4X,'14')
      WRITE(9,5650)
 5650 FORMAT('1')
      WRITE(*,*)"                  ***** JOB COMPLETED *****"
      WRITE(*,5655)FN1,FN2
 5655 FORMAT (  '             OUTPUT : 'A12,' & ',A12)
      STOP
      END
```

B.3 ESAL CALCULATION PROGRAM

AGGCALC PROGRAM

PROCESS OUTPUT FROM LOADOMTR AND CLASS SUMMARY PROGRAMS TO PRODUCE ESAL-RELATED PARAMETERS

```
c $debug
      DOUBLE PRECISION RTE
      CHARACTER FIN*64,FOUT*64,IH1*64
      CHARACTER*3 DD,DD1,STA,STAC,DUM1
      CHARACTER*50 FCLAS,FAXLE,FWGT(3),FMEAN(10),FWGTT,FMEAT,FMNEW
      CHARACTER*80 DUM2
      CHARACTER*100 TMPREC
      INTEGER NWANC(6),NWAC(6),WNC(14,6,16),WC(14,6,16),NWNC(14,6),NWC
     +(14,6),WANC(6,16),WAC(6,16),A,YEAR,N,AADT,AADTC,FRCOAL,GA,
     +NWANCA(6),NWACA(6),WNCA(14,6,16),WCA(14,6,16),NWNCA(14,6),NWCA
     +(14,6),WANCA(6,16),WACA(6,16),VNC(14),VC(14),RECNO,COU,STAA,
     +FEDAID,COAL,NOSEAS,VTYP(14),COALV,NOTRUK,SUMVC2,NCOUNT(11,5,2,4),
     +DVOL(14),YR,P,NWC1(14,1),Z,NYR(15),VC1(14),VC2(14),VC3(14),SUMVC1,
     +DELVC(14),TRKLFT,SUMVC3,SUMVNC,SUMVC,NCONT(11,2,3),IIWSTA(15),
     +NCON(11,2,3,25),NCOUN(11,5,2,4,25)
C
      REAL DA(6,16),AXPER(14,6),AXPERC(14,6),LA,CNTYR(11,5,2,4,25),
     +FNC(14,6,16),FC(14,6,16),FRC(14),TOTAL,CNTCYR(11,2,4,25),
     +FT,FR(14),MP,NOAXLE,MEAN(11,5,2,4),STD(11,5,2,4),COUNT(5,2,4),
     +MEANC(11,2,3),SUM(11,5,2,4),SUMC(11,2,3),CCOUNT(2,3),STDC(11,2,3),
     +SSQ(11,5,2,4),SSQC(11,2,3),MEANY(11,5,2,4,25),MEANCY(11,2,3,25),
     +DIFF(11,5,2,4,25),DIFFC(11,2,3,25),NOAXLC,COUNTC(5,2,4),COUN(5,2
     +,4),AMEAN(11),AMEANC(11),ASUM(11),ACNT(11),ASUMC(11),ACNTC(11)
      REAL EALDNC(15),EALDC(15)
      INTEGER DFNC(12),ihist
C
C
C
C     ****************************************************************
C     * STATEMENTS 25 THRU 54  ARE DATA STATEMENTS TO ASSIGN DAMAGE    *
C     * FACTORS ACCORDING TO LOAD CATEGORY AND AXLE TYPE.              *
C     ****************************************************************
C
      DATA DA(1,1),DA(1,2),DA(1,3),DA(1,4),DA(1,5),DA(1,6),DA(1,7),DA(1,
     +8),DA(1,9),DA(1,10),DA(1,11),DA(1,12),DA(1,13),DA(1,14),DA(1,15),
     +DA(1,16)
     +/.0001,.003,.0138,.0387,.0855,.163,.281,.451,.685,.997,1.40,1.91,
     +2.55,3.33,4.27,5.39/
      DATA DA(2,1),DA(2,2),DA(2,3),DA(2,4),DA(2,5),DA(2,6),DA(2,7),DA(2,
     +8),DA(2,9),DA(2,10),DA(2,11),DA(2,12),DA(2,13),DA(2,14),DA(2,15)
     +,DA(2,16)
     +/.0004,.0027,.012,.0406,.1130,.277,.613,1.25,2.41,4.40,7.68,12.9,
     +21.1,33.5,51.9,78.6/
      DATA DA(3,1),DA(3,2),DA(3,3),DA(3,4),DA(3,5),DA(3,6),DA(3,7),DA(3,
     +8),DA(3,9),DA(3,10),DA(3,11),DA(3,12),DA(3,13),DA(3,14),DA(3,15),
```

```
     +DA(3,16)
     +/.0007,.0029,.0113,.0359,.0966,.231,.502,1.02,1.94,3.52,6.14,10.3,
     +16.8,26.8,41.5,63.0/
      DATA DA(4,1),DA(4,2),DA(4,3),DA(4,4),DA(4,5),DA(4,6),DA(4,7),DA(4,
     +8),DA(4,9),DA(4,10),DA(4,11),DA(4,12),DA(4,13),DA(4,14),DA(4,15),
     +DA(4,16)
     +/.0007,.0029,.0115,.0363,.097,.23,.497,.998,1.89,3.41,5.91,9.89,
     +16.,25.3,39.1,59./
      DATA DA(5,1),DA(5,2),DA(5,3),DA(5,4),DA(5,5),DA(5,6),DA(5,7),DA(5,
     +8),DA(5,9),DA(5,10),DA(5,11),DA(5,12),DA(5,13),DA(5,14),DA(5,15),
     +DA(5,16)
     +/.0006,.0029,.0115,.0366,.098,.232,.502,1.01,1.90,3.43,5.92,9.88,
     +16.,25.2,38.7,58.3/
      DATA DA(6,1),DA(6,2),DA(6,3),DA(6,4),DA(6,5),DA(6,6),DA(6,7),DA(6,
     +8),DA(6,9),DA(6,10),DA(6,11),DA(6,12),DA(6,13),DA(6,14),DA(6,15),
     +DA(6,16)
     +/.05,.12,.28,.66,1.57,3.73,8.88,21.1,50.2,119.,284.,676.,1607.,
     +3822.,9090.,21619./
C
C     ****************************************************************
C     * STATEMENTS 61 THRU 90 ARE DATA STATEMENTS TO ASSIGN AXLELOAD   *
C     * DISTRIBUTION FACTORS TO BUSSES.                                *
C     ****************************************************************
C
C     DATA FNC(4,1,1),FNC(4,1,2),FNC(4,1,3),FNC(4,1,4),FNC(4,1,5),
C    +FNC(4,1,6),FNC(4,1,7),FNC(4,1,8),FNC(4,1,9),FNC(4,1,10),
C    +FNC(4,1,11),FNC(4,1,12),FNC(4,1,13),FNC(4,1,14),FNC(4,1,15),
C    +FNC(4,1,16)/.0046,.0344,.0796,.1403,.1352,.1706,.1859,.1807,.0401
C    +,.0173,.0043,.0012,.0044,.0014,.0,.0/
C     DATA FNC(4,2,1),FNC(4,2,2),FNC(4,2,3),FNC(4,2,4),FNC(4,2,5),
C    +FNC(4,2,6),FNC(4,2,7),FNC(4,2,8),FNC(4,2,9),FNC(4,2,10),
C    +FNC(4,2,11),FNC(4,2,12),FNC(4,2,13),FNC(4,2,14),FNC(4,2,15),
C    +FNC(4,2,16)/.0192,.1181,.1920,.1492,.1092,.1615,.1679,.0762,.0054
C    +,.0011,.0001,.0,.0,.0,.0,.0/
C     DATA FNC(4,6,1),FNC(4,6,2),FNC(4,6,3),FNC(4,6,4),FNC(4,6,5),
C    +FNC(4,6,6),FNC(4,6,7),FNC(4,6,8),FNC(4,6,9),FNC(4,6,10),
C    +FNC(4,6,11),FNC(4,6,12),FNC(4,6,13),FNC(4,6,14),FNC(4,6,15),
C    +FNC(4,6,16)/.0228,.1282,.2086,.2037,.1605,.1194,.1058,.0471,.0032
C    +,.0007,.0,.0,.0,.0,.0,.0/
C     DATA FNC(5,1,1),FNC(5,1,2),FNC(5,1,3),FNC(5,1,4),FNC(5,1,5),
C    +FNC(5,1,6),FNC(5,1,7),FNC(5,1,8),FNC(5,1,9),FNC(5,1,10),
C    +FNC(5,1,11),FNC(5,1,12),FNC(5,1,13),FNC(5,1,14),FNC(5,1,15),
C    +FNC(5,1,16)/.0046,.0344,.0796,.1403,.1352,.1706,.1859,.1807,.0401
C    +,.0173,.0043,.0012,.0044,.0014,.0,.0/
C     DATA FNC(5,2,1),FNC(5,2,2),FNC(5,2,3),FNC(5,2,4),FNC(5,2,5),
C    +FNC(5,2,6),FNC(5,2,7),FNC(5,2,8),FNC(5,2,9),FNC(5,2,10),
C    +FNC(5,2,11),FNC(5,2,12),FNC(5,2,13),FNC(5,2,14),FNC(5,2,15),
C    +FNC(5,2,16)/.0192,.1181,.1920,.1492,.1092,.1615,.1679,.0762,.0054
C    +,.0011,.0001,.0,.0,.0,.0,.0/
C     DATA FNC(5,6,1),FNC(5,6,2),FNC(5,6,3),FNC(5,6,4),FNC(5,6,5),
C    +FNC(5,6,6),FNC(5,6,7),FNC(5,6,8),FNC(5,6,9),FNC(5,6,10),
C    +FNC(5,6,11),FNC(5,6,12),FNC(5,6,13),FNC(5,6,14),FNC(5,6,15),
C    +FNC(5,6,16)/.0228,.1282,.2086,.2037,.1605,.1194,.1058,.0471,.0032
```

```
C    +,.0007,.0,.0,.0,.0,.0,.0/
C
C     ****************************************************************
C     * STATEMENTS 97 THRU 104 ARE DATA STATEMENTS TO ASSIGN THE NUMBER*
C     * OF AXLES OF A GIVEN AXLE TYPE TO BUSSES.                      *
C     ****************************************************************
C
C     DATA AXPER(4,1),AXPER(4,2),AXPER(4,3),AXPER(4,4),AXPER(4,5),AXPER
C    +(4,6)/1.,1.,.0,.0,.0,2./
C     DATA AXPER(5,1),AXPER(5,2),AXPER(5,3),AXPER(5,4),AXPER(5,5),AXPER
C    +(5,6)/1.,1.7,.0,.0,.0,2.7/
C     DATA AXPERC(4,1),AXPERC(4,2),AXPERC(4,3),AXPERC(4,4),AXPERC(4,5),
C    +AXPERC(4,6)/1.,1.,.0,.0,.0,2./
C     DATA AXPERC(5,1),AXPERC(5,2),AXPERC(5,3),AXPERC(5,4),AXPERC(5,5),
C    +AXPERC(5,6)/1.,1.7,.0,.0,.0,2.7/
C
C     ****************************************************************
C     * STATEMENTS 110 THRU 231 INITIALIZE VARIBLES AND ARRAYS.       *
C     ****************************************************************

      DATA DD/'STA'/
      WRITE(*,'(A)')' Specify Input File Name-'
      READ(*,'(A)')FIN
      OPEN(5,FILE=FIN,STATUS='OLD',BLANK='ZERO')
      WRITE(*,'(A)')' Specify Output File Name-'
      READ(*,'(A)')FOUT
      OPEN(6,FILE=FOUT,STATUS='UNKNOWN')
      OPEN(8,FILE='UNIT8D',STATUS='UNKNOWN')
      OPEN(9,FILE='UNIT9D',STATUS='UNKNOWN')
      WRITE(*,'(A)')' INPUT AGGREGATE CLASS FOR THIS RUN'
      READ(*,'(BN,I2)') IINFC
      WRITE(*,'(A)')' INPUT NUMBER OF WEIGHT YEARS FOR THIS RUN'
      READ(*,'(I1)') IINYR
      IF (IINYR.EQ.0) IINYR=10
      WRITE(*,'(A)')' INPUT NUMBER OF HISTORY YEARS FOR THIS RUN'
      READ(*,'(I2)') IHIST
      WRITE(*,'(A)')' INPUT YEAR FOR THIS RUN'
      READ(*,'(I2)') ICYEAR
c     WRITE(*,'(A)')' DO YOU WANT THE INCREMENTS PRINTED 1-YES, 0-NO?'
c     READ(*,'(BZ,I1)',END=1001) INCR
      pyear=icyear
      incr=0
      READ(5,54441)FAXLE
54441 FORMAT(A12)
      READ(5,54441)FCLAS
      DO 55449 I = 1,IINYR
      READ(5,54441)FWGT(I)
55449 CONTINUE
      READ(5,54441)FMNEW
      DO 55469 I = 1,ihist
      READ(5,54441)FMEAN(I)
55469 CONTINUE
1001  CONTINUE
```

```
      OPEN(19,FILE=FAXLE,STATUS='OLD',BLANK='ZERO')
      READ(19,12999) DUM2
      READ(19,12999) DUM2
      DO 12920 I=4,14
12999 FORMAT(A80)
      READ(19,12917,END=12933)EALDNC(I),EALDC(I)
      WRITE(8,12917)EALDNC(I),EALDC(I)
12917 FORMAT(T16,2F10.5)
12920 CONTINUE
12933 CONTINUE
      IF (ICYEAR.EQ.0) ICYEAR=89
C     NNYR=ICYEAR-83
c     NNYR=9
      NNYR=ihist
      IYEAR=0
      NYEAR=0
      ICTCHK=20000
      P=0
      N=0
      N6=0
      M1=0
      IFF=0
      DO 30 I=1,11
      DO 25 J=1,5
      DO 20 K=1,2
      DO 15 L=1,4
      DO 10 M=1,25
      MEANY(I,J,K,L,M)=0.
      NCOUN(I,J,K,L,M)=0.
      DIFF(I,J,K,L,M)=0.
      CNTYR(I,J,K,L,M)=0.
      CNTCYR(I,K,L,M)=0.
   10 CONTINUE
   15 CONTINUE
   20 CONTINUE
   25 CONTINUE
   30 CONTINUE
      DO 50 I=1,11
      DO 45 J=1,2
      DO 40 K=1,3
      DO 35 L=1,25
      MEANCY(I,J,K,L)=0.
      DIFFC(I,J,K,L)=0.
      NCON(I,J,K,L)=0.
   35 CONTINUE
   40 CONTINUE
   45 CONTINUE
   50 CONTINUE
      DO 55 I=1,15
      NYR(I)=0.
   55 CONTINUE
      DO 75 I=1,11
      DO 70 J=1,5
```

```
      DO 65 K=1,2
      DO 60 L=1,4
      COUNT(J,K,L)=0.
      NCOUNT(I,J,K,L)=0.
      COUN(J,K,L)=0.
      COUNTC(J,K,L)=0.
      SUM(I,J,K,L)=0.
      SSQ(I,J,K,L)=0.
      STD(I,J,K,L)=0.
      MEAN(I,J,K,L)=0.
   60 CONTINUE
   65 CONTINUE
   70 CONTINUE
   75 CONTINUE
      DO 90 I=1,11
      DO 85 J=1,2
      DO 80 K=1,3
      CCOUNT(J,K)=0.
      SUMC(I,J,K)=0.
      NCONT(I,J,K)=0.
      SSQC(I,J,K)=0.
      MEANC(I,J,K)=0.
      STDC(I,J,K)=0.
   80 CONTINUE
   85 CONTINUE
   90 CONTINUE
      DO 95 J=1,6
      NWANC(J)=0.
      NWAC(J)=0.
      NWANCA(J)=0.
      NWACA(J)=0.
   95 CONTINUE
      DO 110 L=1,16
      DO 105 K=1,6
      DO 100 J=1,14
      WNC(J,K,L)=0.
      WC(J,K,L)=0.
      FC(J,K,L)=0.
      WNCA(J,K,L)=0.
      WCA(J,K,L)=0.
  100 CONTINUE
  105 CONTINUE
  110 CONTINUE
      DO 125 L=1,16
      DO 120 K=1,6
      DO 115 J=1,3
      FNC(J,K,L)=0.
  115 CONTINUE
  120 CONTINUE
  125 CONTINUE
      DO 140 L=1,16
      DO 135 K=1,6
      DO 130 J=6,14
```

```
      FNC(J,K,L)=0.
  130 CONTINUE
  135 CONTINUE
  140 CONTINUE
      DO 155 L=1,16
      DO 150 K=3,5
      DO 145 J=4,5
      FNC(J,K,L)=0.
  145 CONTINUE
  150 CONTINUE
  155 CONTINUE
      DO 165 K=1,6
      DO 160 J=1,14
      NWC1(J,1)=0
      NWNC(J,K)=0.
      NWC(J,K)=0.
      NWNCA(J,K)=0.
      NWCA(J,K)=0.
  160 CONTINUE
  165 CONTINUE
      DO 175 K=1,16
      DO 170 J=1,6
      WANC(J,K)=0.
      WAC(J,K)=0.
      WANCA(J,K)=0.
      WACA(J,K)=0.
  170 CONTINUE
  175 CONTINUE
      IFNAM=0
  180 CONTINUE
C
C     ****************************************************************
C     * STATEMENTS 237 THRU 307 READ THE AXLELOAD DISTRIBUTIONS FROM   *
C     * A MAGNETIC TAPE PRODUCED BY THE PROGRAM LOADOMTR SUMMARY.      *
C     ****************************************************************
C
      IFNAM=IFNAM+1
      FWGTT=FWGT(IFNAM)
      n=ifnam
      IF(N.GT.IINYR)GO TO190
      OPEN(14,FILE=FWGTT,STATUS='OLD',BLANK='ZERO')
c     N=IYEAR-NYEAR
c     IF(N.GT.IINYR)GO TO190
C     IF(N.GT.8)GO TO190
C      DO 185 I=3,14
C      READ (14,4000) (NWNC(I,J),J=1,6)
C 4000 FORMAT(T15,6I6)
C  185 CONTINUE
C
C     ****************************************************************
C     * STATEMENT 205 READS THE YEAR TO DETERMINE WHETHER TO          *
C     * READ 14 VEHICLE TYPES(1988 & PRIOR YEARS) OR 13 VEHICLE TYPES. *
C     ****************************************************************
```

```
C
      read (14,4444)tmprec
 4444 format(a100)
      read(tmprec,9000)ijyear
 9000 format(i2)
      if(ijyear.ge.96) then
      READ (tmprec,94000) JYEAR,(NWNC(3,J),J=1,6),IIWSTA(N)
      else
      READ (tmprec,4000) JYEAR,(NWNC(3,J),J=1,6),IIWSTA(N)
      endif
      IF (JYEAR.EQ.ICYEAR) IWSTA=IIWSTA(N)
C     WRITE (8,4000) IIWSTA(N),IWSTA
 4000 FORMAT (I2,T15,6I6,T70,I4)
94000 FORMAT (I2,T15,6I7,T75,I4)
C4001 FORMAT (I2,'NWNC 3',T15,6I6,' N= ',I5)
 4001 FORMAT (2I7)
      IY=13
      IF (JYEAR.LE.87) IY=14
      DO 185 I=4,IY
      if(ijyear.ge.96) then
      READ (14,94005) (NWNC(I,J),J=1,6)
      else
      READ (14,4005) (NWNC(I,J),J=1,6)
      endif
C     WRITE (8,4006) I,(NWNC(I,J),J=1,6)
 4005 FORMAT (T15,6I6)
94005 FORMAT (T15,6I7)
C4006 FORMAT (T3,'NWNC',I2,T15,6I6)
  185 CONTINUE
      IF (JYEAR.GT.87) GO TO 190
      DO 186 J=1,6
      NWNC(4,J)=NWNC(4,J)+NWNC(5,J)
  186 CONTINUE
      DO 188 I=5,13
      DO 187 J=1,6
      NWNC(I,J)=NWNC(I+1,J)
  187 CONTINUE
C     WRITE (8,14006)  I,(NWNC(I,J),J=1,6)
C4006 FORMAT (T3,'NWNC-C',I2,T15,6I6)
  188 CONTINUE
  190 CONTINUE
      IF (N.LE.8)GO TO 200
      DO 195 J=1,12
      READ (14,4125) A
CX    WRITE (7,4126) A
  195 CONTINUE
  200 CONTINUE
      if(ijyear.ge.96) then
      READ (14,94025) NYEAR,(NWANC(J),J=1,6)
      else
      READ (14,4025) NYEAR,(NWANC(J),J=1,6)
      endif
C     WRITE (8,4026) NYEAR,(NWANC(J),J=1,6)
```

```
      IF(N.EQ.0) IYEAR=NYEAR
      NYEAR=NYEAR-1
 4025 FORMAT(I2,T15,6I6)
94025 FORMAT(I2,T15,6I7)
C4026 FORMAT(I2,T3,'NWANC',T15,6I6)
C     IF(N.GT.8) GO TO 210
      IF(N.GT.IINYR) GO TO 210
C       DO 205 I=3,14
C       READ (14,4050) (NWC(I,J),J=1,6)
C 4050 FORMAT(T15,6I6)
C  205 CONTINUE
      DO 205 I=3,IY
      if(ijyear.ge.96) then
      READ (14,94050) (NWC(I,J),J=1,6)
      else
      READ (14,4050) (NWC(I,J),J=1,6)
      endif
C     WRITE (8,4051) I,(NWC(I,J),J=1,6)
 4050 FORMAT (T15,6I6)
94050 FORMAT (T15,6I7)
C4051 FORMAT (T4,'NWC',I2,T15,6I6)
  205 CONTINUE
      IF (JYEAR.GT.87) GO TO 209
      DO 206 J=1,6
      NWC(4,J)=NWC(4,J)+NWC(5,J)
  206 CONTINUE
      DO 208 I=5,13
      DO 207 J=1,6
      NWC(I,J)=NWC(I+1,J)
  207 CONTINUE
C     WRITE (8,4059) I,(NWC(I,J),J=1,6)
C4059 FORMAT (T4,'NWC-C',I2,T15,6I6)
  208 CONTINUE
  209 CONTINUE
      IF (N.LE.8)GO TO 220
  210 DO 215 J=1,12
      READ (14,4125) A
CX    WRITE (7,4126) A
 4126 FORMAT (I2,2X,'A')
  215 CONTINUE
  220 CONTINUE
      if(ijyear.ge.96) then
      READ (14,94075) (NWAC(J),J=1,6)
      else
      READ (14,4075) (NWAC(J),J=1,6)
      endif
94075 FORMAT(T15,6I7)
C     WRITE (8,4076) (NWAC(J),J=1,6)
C     IF (N.GT.8)GO TO 236
      IF (N.GT.IINYR)GO TO 236
 4075 FORMAT(T15,6I6)
C4076 FORMAT(T3,'NWAC',T15,6I6)
C       DO 230 I=3,14
```

```
C      DO 225 J=1,6
C      READ (14,4100) (WNC(I,J,K),K=1,8)
C 4100 FORMAT(T15,8I6)
C      READ (14,4100) (WNC(I,J,K),K=9,16)
C  225 CONTINUE
C  230 CONTINUE
       DO 226 I=3,IY
       DO 225 J=1,6
       READ (14,4100) (WNC(I,J,K),K=1,8)
C      WRITE (8,4101) (I,J,(WNC(I,J,K),K=1,8))
 4100 FORMAT (T15,8I6)
C4101 FORMAT (T4,'WNC',I2,I2,T15,8I6)
       READ (14,4100) (WNC(I,J,K),K=9,16)
C      WRITE (8,4101) (I,J,(WNC(I,J,K),K=9,16))
  225 CONTINUE
  226 CONTINUE
       IF (JYEAR.GT.87) GO TO 235
       DO 231 J=1,6
       DO 230 K=1,16
       WNC(4,J,K)=WNC(4,J,K)+WNC(5,J,K)
  230 CONTINUE
  231 CONTINUE
       DO 234 I=5,13
       DO 233 J=1,6
       DO 232 K=1,16
       WNC(I,J,K)=WNC(I+1,J,K)
  232 CONTINUE
  233 CONTINUE
  234 CONTINUE
  235 CONTINUE
       IF (N.LE.8)GO TO 245
  236 CONTINUE
       DO 240 I=1,144
       READ (14,4125) A
CX     WRITE (7,4126) A
  240 CONTINUE
 4125 FORMAT (I2)
  245 CONTINUE
       DO 250 J=1,6
       READ (14,4150) (WANC(J,K),K=1,8)
C      WRITE (8,4151) (J,(WANC(J,K),K=1,8))
 4150 FORMAT(T15,8I6)
 4151 FORMAT(T3,'WANC',I2,T15,8I6)
       READ (14,4150) (WANC(J,K),K=9,16)
C      WRITE (8,4151) (J,(WANC(J,K),K=9,16))
  250 CONTINUE
       IF(N.LE.8)GO TO 260
       DO 255 I=1,144
       READ (14,4125,END=290) A
CX     WRITE (7,4126,END=290) A
  255 CONTINUE
       GO TO 275
  260 CONTINUE
```

```
  256 CONTINUE
      DO 258 I=3,IY
      DO 257 J=1,6
      READ (14,4175) (WC(I,J,K),K=1,8)
C     WRITE (8,4176) (I,J,(WC(I,J,K),K=1,8))
 4175 FORMAT(T15,T15,8I6)
 4176 FORMAT(T5,'WC',I2,I2,T15,T15,8I6)
      READ (14,4175) (WC(I,J,K),K=9,16)
C     WRITE (8,4176) (I,J,(WC(I,J,K),K=9,16))
  257 CONTINUE
  258 CONTINUE
  275 CONTINUE
      IF (JYEAR.GT.87) GO TO 270
      DO 262 J=1,6
      DO 261 K=1,16
      WC(4,J,K)=WC(4,J,K)+WC(5,J,K)
  261 CONTINUE
  262 CONTINUE
      DO 265 I=5,13
      DO 264 J=1,6
      DO 263 K=1,16
      WC(I,J,K)=WC(I+1,J,K)
  263 CONTINUE
  264 CONTINUE
  265 CONTINUE
  270 CONTINUE
C      DO 270 I=3,14
C      DO 265 J=1,6
C      READ (14,4175) (WC(I,J,K),K=1,8)
C 4175 FORMAT(T15,T15,8I6)
C      READ (14,4175) (WC(I,J,K),K=9,16)
C  265 CONTINUE
C  270 CONTINUE
C  275 CONTINUE
      DO 280 J=1,6
      READ (14,4200) (WAC(J,K),K=1,8)
C     WRITE (8,4201) (J,(WAC(J,K),K=1,8))
 4200 FORMAT(T15,8I6)
 4201 FORMAT(T4,'WAC',I1,T15,8I6)
      READ (14,4200) (WAC(J,K),K=9,16)
C     WRITE (8,4201) (J,(WAC(J,K),K=9,16))
  280 CONTINUE
      DO 285 J=1,3
      READ (14,4125,END=290) A
CX    WRITE (7,4126,END=290) A
  285 CONTINUE
  290 CONTINUE
C
C     ****************************************************************
C     * STATEMENTS 316 THRU 463 CALCULATE NUMBERS OF AXLES OF EACH     *
C     * AXLE TYPE FOR EACH VEHICLE WHICH OVER 10 VEHICLES WERE WEIGHED.*
C     * FOR THOSE WHICH LESS THAN 10 VEHICLES WERE WEIGHED PRESELECTED *
C     * DISTRIBUTIONS WILL BE USED.                                    *
```

```
C     ****************************************************************
C
      P=P+1
      IF (P.NE.1) GOTO 430
      DO 425 I=4,13
CCCC
C     WRITE(9,77345)NWNC(I,1),NWC(I,1),I
77345 FORMAT(' NWNC = ',I10, ' NWC = ',2I10)
      DO 420 J=1,6
      IF (NWNC(I,1).LT.10)GO TO 295
      AXPER(I,J)=FLOAT(NWNC(I,J))/FLOAT(NWNC(I,1))
      GO TO 350
  295 CONTINUE
      GO TO (300,300,300,304,305,310,315,320,325,330,335,340,345),I
  300 CONTINUE
  304 AXPER(4,1)=1.
      AXPER(4,2)=0.6396
      AXPER(4,3)=0.4089
      AXPER(4,4)=0.
      AXPER(4,5)=0.
      AXPER(4,6)=2.4574
      GO TO 350
  305 AXPER(5,1)=1.
      AXPER(5,2)=1.
      AXPER(5,3)=0.
      AXPER(5,4)=0.
      AXPER(5,5)=0.
      AXPER(5,6)=2.
      GO TO 350
  310 AXPER(6,1)=1.
      AXPER(6,2)=0.
      AXPER(6,3)=1.
      AXPER(6,4)=0.
      AXPER(6,5)=0.
      AXPER(6,6)=3.
      GO TO 350
  315 AXPER(7,1)=1.
      AXPER(7,2)=0.
      AXPER(7,3)=0.
      AXPER(7,4)=1.
      AXPER(7,5)=0.
      AXPER(7,6)=4.
      GO TO 350
  320 AXPER(8,1)=1.
      AXPER(8,2)=1.212
      AXPER(8,3)=1.789
      AXPER(8,4)=0.
      AXPER(8,5)=0.
      AXPER(8,6)=3.791
      GO TO 350
  325 AXPER(9,1)=1.
      AXPER(9,2)=0.046
      AXPER(9,3)=1.973
```

```
      AXPER(9,4)=0.003
      AXPER(9,5)=0.
      AXPER(9,6)=5.
      GO TO 350
  330 AXPER(10,1)=1.
      AXPER(10,2)=0.008
      AXPER(10,3)=0.951
      AXPER(10,4)=1.017
      AXPER(10,5)=0.030
      AXPER(10,6)=6.080
      GO TO 350
  335 AXPER(11,1)=1.
      AXPER(11,2)=3.980
      AXPER(11,3)=0.010
      AXPER(11,4)=0.
      AXPER(11,5)=0.
      AXPER(11,6)=5.
      GO TO 350
  340 AXPER(12,1)=1.
      AXPER(12,2)=3.0
      AXPER(12,3)=1.0
      AXPER(12,4)=0.
      AXPER(12,5)=0.
      AXPER(12,6)=6.
      GO TO 350
  345 AXPER(13,1)=1.
      AXPER(13,2)=2.0
      AXPER(13,3)=2.0
      AXPER(13,4)=0.
      AXPER(13,5)=0.
      AXPER(13,6)=7.
  350 CONTINUE
      IF (NWC(I,1).GT.10)GO TO 410
      IF (NWNC(I,1).LT.10)GO TO 355
      AXPERC(I,J)=FLOAT(NWNC(I,J))/FLOAT(NWNC(I,1))
      IF(J.EQ.6) THEN
C     WRITE(9,78345)AXPERC(I,J),I,J
C78345 FORMAT(' AXPERC(I,J) = ',F10.3,2I5)
      ENDIF
      GO TO 415
  355 CONTINUE
      GO TO (360,360,360,364,365,370,375,380,385,390,395,400,405),I
  360 CONTINUE
  364 AXPERC(4,1)=0.
      AXPERC(4,2)=0.
      AXPERC(4,3)=0.
      AXPERC(4,4)=0.
      AXPERC(4,5)=0.
      AXPERC(4,6)=0.
      GO TO 415
  365 AXPERC(5,1)=1.
      AXPERC(5,2)=1.
      AXPERC(5,3)=0.
```

```
      AXPERC(5,4)=0.
      AXPERC(5,5)=0.
      AXPERC(5,6)=2.
      GO TO 415
  370 AXPERC(6,1)=1.
      AXPERC(6,2)=0.
      AXPERC(6,3)=1.
      AXPERC(6,4)=0.
      AXPERC(6,5)=0.
      AXPERC(6,6)=3.
      GO TO 415
  375 AXPERC(7,1)=1.
      AXPERC(7,2)=0.
      AXPERC(7,3)=0.
      AXPERC(7,4)=1.
      AXPERC(7,5)=0.
      AXPERC(7,6)=4.
      GO TO 415
  380 AXPERC(8,1)=1.
      AXPERC(8,2)=1.212
      AXPERC(8,3)=0.789
      AXPERC(8,4)=0.
      AXPERC(8,5)=0.
      AXPERC(8,6)=3.791
      GO TO 415
  385 AXPERC(9,1)=1.
      AXPERC(9,2)=0.046
      AXPERC(9,3)=1.973
      AXPERC(9,4)=0.003
      AXPERC(9,5)=0.
      AXPERC(9,6)=5.
      GO TO 415
  390 AXPERC(10,1)=1.
      AXPERC(10,2)=0.008
      AXPERC(10,3)=0.951
      AXPERC(10,4)=1.017
      AXPERC(10,5)=0.030
      AXPERC(10,6)=6.080
      GO TO 415
  395 AXPERC(11,1)=1.
      AXPERC(11,2)=3.980
      AXPERC(11,3)=0.010
      AXPERC(11,4)=0.
      AXPERC(11,5)=0.
      AXPERC(11,6)=5.
      GO TO 415
  400 AXPERC(12,1)=1.
      AXPERC(12,2)=3.0
      AXPERC(12,3)=1.0
      AXPERC(12,4)=0.
      AXPERC(12,5)=0.
      AXPERC(12,6)=6.
      GO TO 415
```

```
  405 AXPERC(13,1)=1.
      AXPERC(13,2)=2.0
      AXPERC(13,3)=2.0
      AXPERC(13,4)=0.
      AXPERC(13,5)=0.
      AXPERC(13,6)=7.
      GO TO 415
  410 AXPERC(I,J)=FLOAT(NWC(I,J))/FLOAT(NWC(I,1))
  415 CONTINUE
  420 CONTINUE
  425 CONTINUE
  430 CONTINUE
C
C     ****************************************************************
C     * STATEMENTS 471 THRU 523 SUM THE AXLELOAD DISTRIBUTIONS OVER    *
C     * PAST YEARS UNTIL 200 OF EACH AXLE TYPE FOR EACH VEHICLE TYPE   *
C     * HAVE BEEN WEIGHED OR EIGHT YEARS OF DATA HAVE BEEN EXHAUSTED.  *
C     ****************************************************************
C
C    12-03-91 AVOID LOOKING BACK NEXT LINE
      IF (N.GT.IINYR) GO TO 520
      DO 465 J=1,13
      DO 460 K=1,6
      IF(NWNCA(J,K).LT.ICTCHK) GO TO 435
C     IF(NWNCA(J,K).LT.200) GO TO 435
      GO TO 460
  435 CONTINUE
C     IF (N.GT.8) GO TO 445
      IF (N.GT.IINYR) GO TO 445
      DO 440 L=1,16
      WNCA(J,K,L)=WNC(J,K,L)+WNCA(J,K,L)
  440 CONTINUE
      NWNCA(J,K)=NWNC(J,K)+NWNCA(J,K)
CCCCCC   041290
C     WRITE(9,41290)NWNCA(J,K),NWNC(J,K),I,J
41290 FORMAT(' NWNCA(J,K)= ',I10,' NWNC(J,K)= ',I10,' J= ',I2,' K= ',I2)
  445 CONTINUE
      DO 450 L=1,16
      WANCA(K,L)=WANC(K,L)+WANCA(K,L)
  450 CONTINUE
      NWANCA(K)=NWANC(K)+NWANCA(K)
  455 CONTINUE
  460 CONTINUE
  465 CONTINUE
C     IF (N.GT.8) GO TO 475
      IF (N.GT.IINYR) GO TO 475
      DO 470 J=1,13
      NWC1(J,1)=NWC1(J,1)+NWC(J,1)
C     WRITE(8,47077)NWC1(J,1),NWC(J,1),J
C7077 FORMAT(' NWC1(J,1)= ',I10,' NWC(J,1)= ',I10,I10)
  470 CONTINUE
  475 CONTINUE
      DO 500 J=1,13
```

```
      DO 495 K=1,6
C     IF(NWCA(J,K).GE.ICTCHK) GO TO 495
      IF(NWCA(J,K).GE.200) GO TO 495
C     IF (N.GT.8) GO TO 485
      IF (N.GT.IINYR) GO TO 485
      DO 480 L=1,16
      WCA(J,K,L)=WC(J,K,L)+WCA(J,K,L)
      IF(J.EQ.5) THEN
C     WRITE(13,75756)JYEAR,WCA(J,K,L),WC(J,K,L),J,K,L
      ENDIF
75756 FORMAT(I4,2X,2I10,1X,3(I2,1X))
  480 CONTINUE
      NWCA(J,K)=NWC(J,K)+NWCA(J,K)
  485 CONTINUE
      NWACA(K)=NWAC(K)+NWACA(K)
      IF(J.EQ.5) THEN
C     WRITE(13,75756)JYEAR,NWACA(K),NWAC(K),J,K
      ENDIF
      DO 490 L=1,16
      WACA(K,L)=WAC(K,L)+WACA(K,L)
  490 CONTINUE
  495 CONTINUE
  500 CONTINUE
      DO 505 I=1,6
      DO 505 J=1,16
      WACA(I,J)=WACA(I,J)/13
      WANCA(I,J)=WANCA(I,J)/13
  505 CONTINUE
      DO 510 I=1,6
      NWACA(I)=NWACA(I)/13
      NWANCA(I)=NWANCA(I)/13
  510 CONTINUE
CC    WRITE(11,41291)N,NYEAR
CC41291 FORMAT(' N = ',I5,' NYEAR = ',I5)
C     IF(N.EQ.10.OR.NYEAR.EQ.79)GO TO 520
C     IF(N.EQ.IINYR.OR.NYEAR.EQ.77)GO TO 520
CCCCCC 11-25-91
C     IF(N.EQ.IINYR.OR.NYEAR.EQ.77..OR.IINYR.EQ.1)GO TO 520
CCCCCC 12-04-91
      IF(N+1.GE.IINYR.OR.NYEAR.EQ.77..OR.IINYR.EQ.1)GO TO 520
      DO 515 I=3,13
      IF(NWNCA(I,1).LT.ICTCHK) GO TO 180
      IF(NWCA(I,1).LT.ICTCHK) GO TO 180
  515 CONTINUE
  520 CONTINUE
CW3   DO 34465 J=1,13
CW3   DO 34460 K=1,6
CW3   WRITE(8,34461)NWNCA(J,K),NWCA(J,K)
C
C     ****************************************************************
C     * STATEMENTS 530 THRU 550 CONVERT AXLELOAD DISTRIBUTIONS IN     *
C     * NUMBERS OF AXLES TO AXLELOAD DISTRIBUTIONS IN FRACTIONS.      *
C     ****************************************************************
```

```
C
      DO 555 I=4,13
      DO 550 J=1,5
CCCCCC   041290
C     WRITE(11,41296)NWNCA(I,J),NWCA(I,J),I,J
41296 FORMAT(' NWNCA(I,J)= ',I10,' NWCA(I,J)= ',I10,' I= ',I2,' J= ',I2)
      DO 545 K=1,16
C     IF(NWNCA(I,J).LT.ICTCHK) GO TO 525
      IF(NWNCA(I,J).LT.200) GO TO 525
      FNC(I,J,K)=FLOAT(WNCA(I,J,K))/FLOAT(NWNCA(I,J))
C     WRITE(13,76534)FNC(I,J,K),WNCA(I,J,K),NWNCA(I,J),WANCA(J,K),NWANCA
C    1(J),I,J,K
      GO TO 530
  525 IF (NWANCA(J).EQ.0) GO TO 530
      FNC(I,J,K)=0.
      FNC(I,J,K)=FLOAT(WANCA(J,K))/FLOAT(NWANCA(J))
C     WRITE(13,76535)FNC(I,J,K),WNCA(I,J,K),NWNCA(I,J),WANCA(J,K),NWANCA
C    1(J),I,J,K
  530 CONTINUE
C     IF(NWCA(I,J).LT.ICTCHK) GO TO 535
      IF(NWCA(I,J).LT.200) GO TO 535
      FC(I,J,K)=FLOAT(WCA(I,J,K))/FLOAT(NWCA(I,J))
C     WRITE(8,76532)FC(I,J,K),WCA(I,J,K),NWCA(I,J),WACA(J,K),NWACA(J),I
C    1,J,K
      GO TO 540
  535 IF (NWACA(J).EQ.0) GO TO 540
      FC(I,J,K)=0.
      FC(I,J,K)=FLOAT(WACA(J,K))/FLOAT(NWACA(J))
C     WRITE(8,76533)FC(I,J,K),WCA(I,J,K),NWCA(I,J),WACA(J,K),NWACA(J),I
C    1,J,K
  540 CONTINUE
  545 CONTINUE
  550 CONTINUE
76534 FORMAT('A-FNC= ',F10.3,4(2X,I10),3I5)
76535 FORMAT('B-FNC= ',F10.3,4(2X,I10),3I5)
76532 FORMAT('A- FC= ',F10.3,4(2X,I10),3I5)
76533 FORMAT('B- FC= ',F10.3,4(2X,I10),3I5)
  555 CONTINUE
  560 CONTINUE
C
C     ****************************************************************
C     * STATEMENTS 557 THRU 583 INITIALIZE VARIABLES WHICH MUST BE     *
C     * INITIALIZED FOR EACH INDIVIDUAL CLASSIFICATION STATION.        *
C     ****************************************************************
C
C     IWSTA=IIWSTA(1)
      CHECK=0
      NOTRUK=0
      NOCTRK=0
      FT=0
      SUMVNC=0
      RNOAXC=0
      EALCT=0
```

```
      EALC=0
      NOAXLC=0
      EALT=0
      EALCTA=0
      EALTAX=0
      TEAL=0
      TEALT=0
      TEALCT=0
      FCT=0
      RNOAX=0
      NOAXLE=0
      DO 565 I=1,14
      FRC(I)=0
      FR(I)=0
      DVOL(I)=0
      VC(I)=0
      VNC(I)=0
      SUMVC=0
  565 CONTINUE
  570 CONTINUE
      OPEN(15,FILE=FCLAS,STATUS='OLD',BLANK='ZERO')
C
C     ****************************************************************
C     * STATEMENTS 590 THRU 601 READ CLASSIFICATION DATA FROM        *
C     * A MAGNETIC TAPE PRODUCED BY THE PROGRAM CLASS SUMMARY.       *
C     ****************************************************************
C
      READ(15,4225,END=780)COU,DD1,STA,RTE,MP,YR,AADT,FEDAID,IFCL
CX    WRITE(8,4225)COU,DD1,STA,RTE,MP,YR,AADT,FEDAID,FC
CX    WRITE(7,4226)COU,DD1,STA,RTE,MP,YR,AADT,FEDAID,FC
C 4225 FORMAT(T4,I3,A3,A3,T16,A8,T27,F6.3,T35,I2,T41,I6,T55,I1,T60,I2)
 4225 FORMAT(T4,I3,A3,A3,T16,A9,T27,F6.3,T35,I2,T41,I6,T55,I1,T60,I2)
 4226 FORMAT('1',T4,I3,A3,A3,T16,A8,T27,F6.3,T35,I2,T41,I6,T55,I1,T60,
     +I2)
      IY=13
CC    WRITE(8,72076)COU,DD1,DD,YR
CC72076 FORMAT(I3,2A3,I2)
C***** ADDED 12-05-94 FOR TWO LINE PC CLASS INPUT
      IF (YR.GE.88) GO TO 587
C***** CHANGE 8-8-90 CORRECT FOR YEAR = 0 ONE LINE
      IF (YR.LT.88) THEN IY=14
      IF(DD1.EQ.DD)GO TO 580
      DO 575 I=1,5
      READ(15,4125) A
CX    WRITE(7,4126) A
  575 CONTINUE
      GO TO 560
  580 DO 585 I=1,4
      READ(15,4125) A
CX    WRITE(7,4126) A
  585 CONTINUE
      IF (YR.GE.88) GO TO 587
      READ(15,4250) RECNO,(VTYP(I),I=1,14),COALV,TOTAL
```

```
C     IF(COU.EQ.56.AND.STA.EQ.'847') THEN
C     WRITE(8,4250) RECNO,(VTYP(I),I=1,14),COALV,TOTAL
C     ENDIF
 4250 FORMAT(I1,T8,I4,I6,I5,2I4,I5,3I4,I5,5I4,T74,F6.0)
      VTYP(4)=VTYP(4)+VTYP(5)
      DO 586 I=5,13
      VTYP(I)=VTYP(I+1)
  586 CONTINUE
      GO TO 588
  587 CONTINUE
      READ(15,4251) RECNO,(VTYP(I),I=1,13),COALV,TOTAL
C     IF(COU.EQ.56.AND.STA.EQ.'847') THEN
C     WRITE(9,4251) RECNO,(VTYP(I),I=1,13),COALV,TOTAL
C     ENDIF
 4251 FORMAT(I1,T8,I4,I6,I5,I4,I5,3I4,I5,5I4,T70,F6.0)
  588 CONTINUE
C
C     ******************************************************************
C     * STATEMENTS 607 THRU 609 DETECT ERRORS IN INPUT.                *
C     ******************************************************************
C
CCCCC
      FEDAID=1
      IF(FEDAID.EQ.8)FEDAID=5
      IF(COU.EQ.0.OR.COU.GT.120.OR.FEDAID.GT.5.OR.AADT.LE.
     +0.OR.FEDAID.EQ.0) GO TO 710
C
C     ******************************************************************
C     * STATEMENTS 615 THRU 672 CALCULATE THE TRAFFIC PARAMETERS       *
C     * NECCESARY TO CALCULATE EAL'S.                                  *
C     ******************************************************************
C
      TOTAL=ABS(TOTAL)
      DO 590 I=4,13
      NOTRUK=NOTRUK+VTYP(I)
      NOCTRK=NOCTRK+NWC1(I,1)
CC      WRITE (8,59099)NOCTRK,NWC1(I,1),I,N
CC59099 FORMAT(' NOCTRK = ',I10,' NWC1(I,1) = ',I10,2I10)
  590 CONTINUE
CCCCC
      IF(COU.EQ.48.AND.STA.EQ.'P18') THEN
      DO 98785 IK=1,14
      DO 98786 IT=1,6
C     WRITE(9,98787)NWC(IK,IT),NWNC(IK,IT),IK,IT
98787 FORMAT(' NWC(IK,IT) = ',I10,' NWNC(IK,IT)= ',I10,I5,I5)
98786 CONTINUE
98785 CONTINUE
      ENDIF
      DO 595 I=1,13
      FR(I)=FLOAT(VTYP(I))/TOTAL
      DVOL(I)=FR(I)*AADT+0.5
  595 CONTINUE
      IF(NOTRUK.EQ.0) GO TO600
```

```
      FCT=FLOAT(COALV)/NOTRUK
  600 CONTINUE
      FT=FLOAT(NOTRUK)/TOTAL
      COALV=(FLOAT(COALV)*FLOAT(AADT)/TOTAL)+.5
      IF(COALV.EQ.0) GO TO 660
      SUMVNT=0
      DO 605 I=1,13
C     IF(COU.EQ.56) THEN
C     WRITE(9,23333)COALV,NWC1(I,1),NOCTRK,I
C3333 FORMAT('COALV= ',3I10,I5)
C     ENDIF
      IF(NOCTRK.EQ.0) THEN
      FRC(I)=0
      ELSE
      FRC(I)=(FLOAT(NWC1(I,1)))/(FLOAT(NOCTRK))
      ENDIF
      VC(I)=FRC(I)*COALV+0.5
      SUMVNT=SUMVNT+VC(I)
C     IF(COU.EQ.56.AND.STA.EQ.'847') THEN
C     WRITE(6,23333)VC(I),FRC(I),COALV,NWC1(I,1),NOCTRK,I
C3333 FORMAT(' VC(I)= ',I10,F10.3,3I10,I5)
C     ENDIF
  605 CONTINUE
      IF(COALV.GT.0.AND.SUMVNT.EQ.0)THEN
       VC(6)=INT(COALV/4)
       VC(7)=INT((COALV-VC(6))/3)
       VC(9)=INT((COALV-VC(7)-VC(6))/2)
       VC(10)=INT(COALV-VC(6)-VC(7)-VC(9))
      ENDIF
      SUMVC1=0
      SUMVC2=0
      SUMVC3=0
      TRKLFT=0
      DO 610 I=1,13
      DELVC(I)=0
  610 CONTINUE
      DO 615 I=5,13
      VC1(I)=MIN0(VC(I),DVOL(I))
      SUMVC1=SUMVC1+VC1(I)
  615 CONTINUE
      CHECK=(COALV-SUMVC1)/FLOAT(COALV)
      IF(CHECK.LE..0001)GO TO 635
      DO 620 I=5,13
      DELVC(I)=(COALV-SUMVC1)*FLOAT(VC(I))/COALV
      VC2(I)=VC1(I)+DELVC(I)
      VC3(I)=MIN0(VC2(I),DVOL(I))
      SUMVC3=SUMVC3+VC3(I)
      IF(COU.EQ.48.AND.STA.EQ.'P18') THEN
C     WRITE(9,62222)CHECK,DELVC(I),COALV,SUMVC1,VC(I),I
62222 FORMAT(F10.4,5I10,' 620 LOOP')
      ENDIF
  620 CONTINUE
      CHECK=(COALV-SUMVC3)/FLOAT(COALV)
```

```
      IF(CHECK.LE..0001)GO TO 645
      DO 625 I=5,13
      TRKLFT=TRKLFT+DVOL(I)-VC3(I)
      IF(COU.EQ.48.AND.STA.EQ.'P18') THEN
C     WRITE(9,62625)DVOL(I),VC3(I),TRKLFT,I
62625 FORMAT(4I10,' 625 LOOP')
      ENDIF
  625 CONTINUE
      DO 630 I=5,13
      DELVC(I)=(COALV-SUMVC3)*FLOAT(DVOL(I)-VC3(I))/TRKLFT
      VC(I)=VC3(I)+DELVC(I)
      IF(COU.EQ.48.AND.STA.EQ.'P18') THEN
C     WRITE(9,62223)CHECK,DELVC(I),COALV,SUMVC3,DVOL(I),VC3(I),TRKLFT,I
62223 FORMAT(F10.4,7I10,' 630 LOOP')
      ENDIF
  630 CONTINUE
      GO TO 655
  635 DO 640 I=5,13
      VC(I)=VC1(I)
  640 CONTINUE
      GO TO 655
  645 DO 650 I=5,13
      VC(I)=VC2(I)
  650 CONTINUE
  655 CONTINUE
      SUMVC1 = 0
      DO 656 I = 1,13
C     IF(COU.EQ.119.AND.STA.EQ.'027') THEN
C     WRITE(9,23735)VC(I),SUMVC1
C     ENDIF
  656 SUMVC1 = SUMVC1 + VC(I)
      IFUDGE = COALV - SUMVC1
      VC(9) = VC(9) + IFUDGE
      IF(VC(9).LT.0) VC(9)=0
23735 FORMAT(' VC(I)= ',I10,' SUMVC1 = ',I10)
  660 CONTINUE
C
C     ****************************************************************
C     * STATEMENTS 679 THRU 715 DISTRIBUTE THE COAL TRUCK VOLUME TO THE*
C     * APPROPRIATE VEHICLE TYPES.                                   *
C     ****************************************************************
C
      DO 665 I=1,13
      VNC(I)=DVOL(I)-VC(I)
      IF(VNC(I).LT.0) VNC(I)=0
C     IF(COU.EQ.119.AND.STA.EQ.'027') THEN
C     WRITE(9,23334)VNC(I),DVOL(I),VC(I),VC1(I),VC2(I),VC3(I),DELVC(I),I
C3334 FORMAT(' VNC(I)= ',I10,6I10,I5)
C     ENDIF
  665 CONTINUE
      DO 670 I=4,13
      SUMVNC=SUMVNC+VNC(I)
      SUMVC=SUMVC+VC(I)
```

```
  670 CONTINUE
CCCC
      TMPE=0
      DO 685 I=4,13
      EALCT=EALCT+VC(I)*EALDC(I)
      EALT=EALT+VNC(I)*EALDNC(I)
C     IF(COU.EQ.119.AND.STA.EQ.'027') THEN
C     WRITE(9,72375)EALCT,VC(I),EALDC(I),I
C     WRITE(11,72375)EALT,VNC(I),EALDNC(I)
C     ENDIF
72375 FORMAT(F10.3,I10,F10.4,I5)
C     DO 680 J=1,5
C     DO 675 K=1,16
C     EALCT=EALCT+VC(I)*FC(I,J,K)*AXPERC(I,J)*DA(J,K)
C     EALT=EALT+VNC(I)*FNC(I,J,K)*AXPER(I,J)*DA(J,K)
C     IF(COU.EQ.41.AND.STA.EQ.'P23'.AND.J.EQ.1.AND.K.EQ.1) THEN
C     IF(COU.EQ.48.AND.STA.EQ.'P18') THEN
C      IF(EALCT.GT.0.AND.EALCT.NE.TMPE) THEN
C     WRITE(9,72075)EALCT,VC(I),FC(I,J,K),AXPERC(I,J),DA(J,K),I,J,K
C     WRITE(11,72075)EALT,VNC(I),FNC(I,J,K),AXPER(I,J),DA(J,K),I,J,K
C72075 FORMAT(F10.3,I10,3F10.3,3I5)
C       TMPE=EALCT
C       ENDIF
C     ENDIF
C 675 CONTINUE
C 680 CONTINUE
  685 CONTINUE
      EALC=(((VNC(2)+VNC(3))*0.005)*365)/1000.
      DO 690 I=4,13
      NOAXLC=NOAXLC+VC(I)*AXPERC(I,6)
      NOAXLE=NOAXLE+VNC(I)*AXPER(I,6)
c     IF(COU.EQ.48.AND.STA.EQ.'P18') THEN
c     IF(noaxlc.EQ.0) THEN
c     WRITE(9,72080)NOAXLC,NOAXLE,VC(I),VNC(I),AXPERC(I,6),AXPER(I,6),I
c72080 FORMAT(' 690 LOOP ',2F10.2,2I10,2F10.2,I5)
c      ENDIF
  690 CONTINUE
      IF(SUMVC.GT.0)GOTO 695
      RNOAXC=0
      EALCTA=0
      GO TO 700
  695 RNOAXC=NOAXLC/SUMVC
      if(noaxlc.eq.0)noaxlc=1
      EALCTA=EALCT/NOAXLC
CCCCC 03-27-90 ADDED NEXT TWO WRITE STATEMENTS
C      WRITE(9,72190)RNOAXC,NOAXLC,SUMVC
C72190 FORMAT(F10.3,F10.2,I10)
C     WRITE(9,72191)EALCTA,EALCT,NOAXLC
72191 FORMAT(' EALCTA=',F10.3,F10.2,F10.2)
  700 CONTINUE
      IF(SUMVNC.GT.0)GO TO 705
      RNOAX=0
      EALTA=0
```

```
      GO TO 710
  705 RNOAX=NOAXLE/SUMVNC
      IF(NOAXLE.EQ.0.OR.EALT.EQ.0) THEN
C       WRITE(9,5175)COU,STA,RTE,MP,FEDAID,AADT,FT,FCT
C       WRITE(9,72074)EALT,NOAXLE,SUMVC,SUMVNC
C72074 FORMAT(' EALT= ',F10.3,' NOAXLE= ',F10.1,' SUMVC= ',I10,' SUMVNC=
C     1',I10)
c     WRITE(8,5175)COU,STA,RTE,MP,FEDAID,AADT,FT,FCT,RNOAX,
c    +RNOAXC,EALTAX,EALCTA,EALC,EALT,EALCT,TEAL
      WRITE(8,5175)COU,STA,RTE,MP,AADT,FT,FCT,RNOAX,
     +RNOAXC,EALTAX,EALCTA,EALC,EALT,EALCT,TEAL
      noaxle=1
      ENDIF
      EALTAX=EALT/NOAXLE
  710 CONTINUE
      EALCT=EALCT*365/1000
      EALT=EALT*365/1000
      TEAL=EALT+EALCT+EALC
C
C     ****************************************************************
C     * STATEMENTS 722 THRU 843 PRINT OUT THE TRAFFIC PARAMETERS FOR   *
C     * EACH CLASSIFICATION STATION.                                   *
C     ****************************************************************
C
      N6=N6+1
      IF(IFF.EQ.1) GO TO 735
CCCC ADDED NEXT LINE TO SUPPRESS PRINTOUT OF VAR. AND CODES
      IF(IFF.EQ.0) GO TO 715
      IF(N6.NE.1) GO TO 715
      WRITE(6,4275)
 4275 FORMAT('1',T51,'VARIABLES AND CODES DEFINED')
      WRITE(6,6175)
      WRITE(6,4300)
 4300 FORMAT(T54,'NON-COAL-HAULING ROADS')
      WRITE(6,4325)
 4325 FORMAT(T40,'COAL TRUCKS COMPRISE LESS THAN 1.0% OF THE TRUCK VOLUM
     +E')
      WRITE(6,6175)
      WRITE(6,4350)
 4350 FORMAT(T50,' FA  - FEDERAL AID CODES')
      WRITE(6,4375)
 4375 FORMAT(T56,' 1 - INTERSTATE')
      WRITE(6,4400)
 4400 FORMAT(T56,' 2 - FEDERAL AID PRIMARY')
      WRITE(6,4425)
 4425 FORMAT(T56,' 3 - FEDERAL AID URBAN')
      WRITE(6,4450)
 4450 FORMAT(T56,' 4 - FEDERAL AID SECONDARY')
      WRITE(6,4475)
 4475 FORMAT(T56,' 5 - NON-FEDERAL AID')
      WRITE(6,6175)
      WRITE(6,4500)
 4500 FORMAT(T50,' GA  - GEOGRAPHIC AREA CODES')
```

```
      WRITE(6,4525)
 4525 FORMAT(T56,' 1 - WEST (HIGHWAY DISTRICTS 1,2)')
      WRITE(6,4550)
 4550 FORMAT(T56,' 2 - SOUTH-CENTRAL (HIGHWAY DISTRICTS 3,4,8)')
      WRITE(6,4575)
 4575 FORMAT(T56,' 3 - NORTH-CENTRAL (HIGHWAY DISTRICTS 5,6,7)')
      WRITE(6,4600)
 4600 FORMAT(T56,' 4 - EAST (HIGHWAY DISTRICTS 9,10,11,12)')
      WRITE(6,6175)
      WRITE(6,4625)
 4625 FORMAT(T50,' VOL - VOLUME CODES')
      WRITE(6,4650)
 4650 FORMAT(T56,' 1 - LESS THAN 5000 AADT')
      WRITE(6,4675)
 4675 FORMAT(T56,' 2 - 5000 OR MORE AADT')
      WRITE(6,6175)
      WRITE(6,4700)
 4700 FORMAT(T56,'COAL-HAULING ROADS')
      WRITE(6,4725)
 4725 FORMAT(T40,'COAL TRUCKS COMPRISE 1.0% OR MORE OF THE TRUCK VOLUME'
     +)
      WRITE(6,6175)
      WRITE(6,4750)
 4750 FORMAT(T50,' CT  - COAL-HAULING ROAD CODES')
      WRITE(6,4775)
 4775 FORMAT(T56,' 1 - COAL TRUCKS COMPRISE 1.0-4.99% OF THE TRUCK VOLUM
     +E')
      WRITE(6,4800)
 4800 FORMAT(T56,' 2 - COAL TRUCKS COMPRISE 5.0-20.00% OF THE TRUCK VOLU
     +ME')
      WRITE(6,4825)
 4825 FORMAT(T56,' 3 - COAL TRUCKS COMPRISE MORE THAN 20.0% OF THE TRUCK
     + VOLUME')
      WRITE(6,6175)
      WRITE(6,4625)
      WRITE(6,4650)
      WRITE(6,4675)
      WRITE(6,6175)
      WRITE(6,4850)
 4850 FORMAT(T48,'INDIVIDUAL CLASSIFICATION STATIONS')
      WRITE(6,6175)
      WRITE(6,4875)
 4875 FORMAT(T50,' DIR')
      WRITE(6,4900)
 4900 FORMAT(T50,' OPR -DIRECTIONAL OPERATION CODES')
      WRITE(6,4925)
 4925 FORMAT(T56,' 1 - ONE-WAY OPERATION')
      WRITE(6,4950)
 4950 FORMAT(T56,' 2 - TWO-WAY OPERATION')
      WRITE(6,6175)
      WRITE(6,4975)
 4975 FORMAT(T50,' FED')
      WRITE(6,5000)
```

```
 5000 FORMAT(T50,' AID - FEDERAL AID CODES')
      WRITE(6,4375)
      WRITE(6,4400)
      WRITE(6,4425)
      WRITE(6,4450)
      WRITE(6,4475)
  715 CONTINUE
CCCCC CHECK
CC    WRITE(8,5175)COU,STA,RTE,MP,FEDAID,AADT,FT,FCT,RNOAX,
CC   +RNOAXC,EALTAX,EALCTA,EALC,EALT,EALCT,TEAL
      M1=M1+1
CCHANGED M1 COUNTER TO 35 FOR WORDPERFECT PRINTOUTS
C     IF(M1.EQ.50)M1=0
      IF(M1.EQ.35)M1=0
      IF(M1.NE.1) GO TO 720
      WRITE(6,5025)
 5025 FORMAT('1',T35,'EAL TRAFFIC PARAMETERS FOR INDIVIDUAL CLASSIFICATI
     +ON STATIONS')
      WRITE(6,5050) YR
      CALL FCHEAD(IINFC,1)
 5050 FORMAT(T63,'19',I2)
c 5051 FORMAT(T61,I4)
      WRITE(6,6175)
      WRITE(6,5075)
C 5075 FORMAT(T43,'FRACT',T50,'AXLES',T56,'AXLES',T62,'EAL"S/',T69,'EAL
C     +'S/',T81,'2-DIRECTION EAL"S IN 1000"S')
C      WRITE (6,5100)
C 5100 FORMAT(T43,'OF TRK',T51,'PER',T57,'PER',T63,'NON',T70,'COAL')
C      WRITE (6,5125)
 5075 FORMAT(T41,'FRACT',T47,'AXLES/TRUCK',T61,'EAL"S/AXLE',
     +T81,'2-DIRECTION EAL"S IN 1000"S')
      WRITE (6,5100)
 5100 FORMAT(T40,'OF TRK',T47,'------------',T60,'------------',T76,
     +'--------------------------------------')
      WRITE (6,5125)
C5125 FORMAT(T20,'MILE',T26,'FED',T36,'TRUCK',T44,
 5125 FORMAT(T21,'MILE',T34,'TRUCK',T42,
     +'WITH',T47,'NORMAL HEAVY',T60,'NORMAL HEAVY',T78,'4-TIRE',
     +T86,'NON-COAL',T98,'COAL')
C5125 FORMAT(T20,'MILE',T36,'TRUCK',T44,
C     +'WITH',T51,'NON',T56,'COAL',T63,'COAL',T70,'TRUCK',T77,'4-TIRE',
C     +T85,'NON-COAL',T98,'COAL')
      WRITE (6,5150)
C5150 FORMAT(T2,'COU',T6,'STA',T11,'ROUTE',T20,'POINT'
C     +,T31,'AADT',T37,'FRACT',T45,'COAL',T52,'COAL',
C     +,T26,'AID',T30,'AADT',T36,'FRACT',T44,'COAL',T51,'COAL',
C     +T56,'TRUCK',T63,'AXLE',T70,'AXLE',T76,'VEHICLES',T86,'TRUCKS',
C     +T97,'TRUCKS',T107,'TOTAL')
 5150 FORMAT(T2,'COU',T6,'STA',T12,'ROUTE',T20,'POINT',T28,'AADT'
     +,T34,'FRACT',T42,'COAL',
     +T54,' COAL',T68,'COAL',T77,'VEHICLES',T87,'TRUCKS',
     +T98,'TRUCKS',T108,'TOTAL')
  720 IF(COU.EQ.0.OR.COU.GT.120.OR.FEDAID.GT.5.OR.AADT.EQ.
```

```
     +0.OR.FEDAID.EQ.0) GO TO 725
C     CALL DEFEAL(IINFC,EALTAX,EALCTA,EALDNC,EALDC)
C     IF(FCT.LE.0.0009)EALCTA=0
C     EALC=(((AADT*(1-FT)*.005)*365)/1000)
C     EALT=(((AADT*(FT*(1-FCT))*RNOAX*EALTAX)*365)/1000)
C     EALCT=(((AADT*(FT*FCT)*RNOAXC*EALCTA)*365)/1000)
C     TEAL=EALC+EALT+EALCT
C     WRITE(6,5175)COU,STA,RTE,MP,FEDAID,AADT,FT,FCT,RNOAX,
C5175 FORMAT(T2,I3,T6,A3,T10,A8,T19,F6.1,T28,I1,T30,I6,
C5175 FORMAT(T2,I3,T6,A3,T10,A9,T19,F6.1,T28,I1,T30,I6,
C5175 FORMAT(T2,I3,T6,A3,T10,A9,T19,F6.1,T30,I6,
C    +T36,F6.3,T43,F6.3,T50,F5.3,T56,F5.3,T62,F6.3,T69,F6.3,T81,F7.0,
C    +T89,F8.0,T100,F8.0,T109,F8.0)
      WRITE(6,5175)COU,STA,RTE,MP,AADT,FT,FCT,RNOAX,
     +RNOAXC,EALTAX,EALCTA,EALC,EALT,EALCT,TEAL
 5175 FORMAT(T2,I3,T6,A3,T10,A9,T19,F6.1,T26,I6,
     +T33,F6.3,T40,F6.3,T47,F5.3,T54,F5.3,T59,F9.3,T69,F8.3,T78,F7.0,
     +T86,F8.0,T96,F8.0,T105,F8.0)
      GO TO 730
  725 WRITE(6,5200)COU,STA,RTE,MP,FEDAID,AADT
C5200 FORMAT(T2,I3,T6,A3,T10,A8,T19,F6.1,T28,I1,T30,I6,
 5200 FORMAT(T2,I3,T6,A3,T10,A9,T19,F6.1,T28,I1,T30,I6,
     +T45,'***************')
      GO TO 560
  730 CONTINUE
  735 CONTINUE
C
C     ****************************************************************
C     * STATEMENTS 852 THRU 888 DETERMINE WHICH CELL OF THE TWO OUTPUT *
C     * MATRICES THE CLASSIFICATION STATION BELONGS.                   *
C     ****************************************************************
C
CCCC CHANGE NEXT 6 LINES TO DIVIDE FRCOAL INTO TWO CATAGORIES
      IF(FCT.LT..03) FRCOAL=4
      IF(FCT.GE..03) FRCOAL=1
C     IF(FCT.LT..01) FRCOAL=4
C     IF(FCT.GE..01.AND.FCT.LT..05) FRCOAL=1
C     IF(FCT.GE..05.AND.FCT.LT..20) FRCOAL=2
C     IF(FCT.GE..20) FRCOAL=3
      FCT=FCT*100
      FT=FT*100
      IF(AADT.LT.5000)AADTC=1
      IF(AADT.GE.5000)AADTC=2
      IF(FEDAID.EQ.8)FEDAID=5
      IF(COU.EQ.4.OR.COU.EQ.17.OR.COU.EQ.18.OR.COU.EQ.20.OR.COU.EQ.24.
     +OR.COU.EQ.28.OR.COU.EQ.30.OR.COU.EQ.38.OR.COU.EQ.42.OR.COU.EQ.46.
     +OR.COU.EQ.51.OR.COU.EQ.53.OR.COU.EQ.54.OR.COU.EQ.70.OR.COU.EQ.72.
     +OR.COU.EQ.73.OR.COU.EQ.75.OR.COU.EQ.79.OR.COU.EQ.89.OR.COU.EQ.92.
     +OR.COU.EQ.111.OR.COU.EQ.113.OR.COU.EQ.117) GA=1
      IF(COU.EQ.1.OR.COU.EQ.2.OR.COU.EQ.5.OR.COU.EQ.14.OR.COU.EQ.16.
     +OR.COU.EQ.23.OR.COU.EQ.27.OR.COU.EQ.29.OR.COU.EQ.31.OR.COU.EQ.43.
     +OR.COU.EQ.44.OR.COU.EQ.47.OR.COU.EQ.50.OR.COU.EQ.62.OR.COU.EQ.69.
     +OR.COU.EQ.71.OR.COU.EQ.74.OR.COU.EQ.78.OR.COU.EQ.82.OR.COU.EQ.85.
```

```
     +OR.COU.EQ.86.OR.COU.EQ.90.OR.COU.EQ.100.OR.COU.EQ.102.OR.COU.EQ.
     +104.OR.COU.EQ.107.OR.COU.EQ.109.OR.COU.EQ.110.OR.COU.EQ.114.OR.
     +COU.EQ.115.OR.COU.EQ.116) GA=2
      IF(COU.EQ.3.OR.COU.EQ.8.OR.COU.EQ.9.OR.COU.EQ.11.OR.COU.EQ.12.
     +OR.COU.EQ.15.OR.COU.EQ.19.OR.COU.EQ.21.OR.COU.EQ.25.OR.COU.EQ.34.
     +OR.COU.EQ.37.OR.COU.EQ.39.OR.COU.EQ.40.OR.COU.EQ.41.OR.COU.EQ.49.
     +OR.COU.EQ.52.OR.COU.EQ.56.OR.COU.EQ.57.OR.COU.EQ.59.OR.COU.EQ.76.
     +OR.COU.EQ.84.OR.COU.EQ.87.OR.COU.EQ.93.OR.COU.EQ.94.OR.COU.EQ.96.
     +OR.COU.EQ.101.OR.COU.EQ.105.OR.COU.EQ.106.OR.COU.EQ.108.OR.
     +COU.EQ.112.OR.COU.EQ.120) GA=3
      IF(COU.EQ.6.OR.COU.EQ.7.OR.COU.EQ.10.OR.COU.EQ.13.OR.COU.EQ.22.
     +OR.COU.EQ.26.OR.COU.EQ.32.OR.COU.EQ.33.OR.COU.EQ.35.OR.COU.EQ.36.
     +OR.COU.EQ.45.OR.COU.EQ.48.OR.COU.EQ.55.OR.COU.EQ.58.OR.COU.EQ.60.
     +OR.COU.EQ.61.OR.COU.EQ.63.OR.COU.EQ.64.OR.COU.EQ.65.OR.COU.EQ.66.
     +OR.COU.EQ.67.OR.COU.EQ.68.OR.COU.EQ.77.OR.COU.EQ.80.OR.COU.EQ.81.
     +OR.COU.EQ.83.OR.COU.EQ.88.OR.COU.EQ.91.OR.COU.EQ.95.OR.
     +COU.EQ.97.OR.COU.EQ.98.OR.COU.EQ.99.OR.COU.EQ.103.OR.
     +COU.EQ.118.OR.COU.EQ.119) GA=4
CCCC  ADDED NEXT THREE LINES
      FEDAID=1
      AADTC=1
      GA=1
      IF (FRCOAL.NE.4) GOTO 760
C
C     ****************************************************************
C     * STATEMENTS 895 THRU 926 CALCULATE THE SUMS AND THE SUM OF THE  *
C     * SQUARES FOR THOSE STATIONS WHICH ARE NON-COAL-HAUL STATONS.    *
C     ****************************************************************
C
      if(aadt.gt.46340)aadt=46340
      DO 755 I=1,5
      DO 750 J=1,2
      DO 745 K=1,4
      IF (FEDAID.NE.I.OR.AADTC.NE.J.OR.GA.NE.K) GO TO 745
      COUN(I,J,K)=COUN(I,J,K)+1
      IF(FCT.LT..00001)GO TO 740
      COUNTC(I,J,K)=COUNTC(I,J,K)+1
  740 CONTINUE
      SUM(1,I,J,K)=SUM(1,I,J,K)+AADT
      SSQ(1,I,J,K)=SSQ(1,I,J,K)+AADT**2
      SUM(2,I,J,K)=SUM(2,I,J,K)+FT
      SSQ(2,I,J,K)=SSQ(2,I,J,K)+FT**2
      SUM(3,I,J,K)=SUM(3,I,J,K)+FCT
      SSQ(3,I,J,K)=SSQ(3,I,J,K)+FCT**2
      SUM(4,I,J,K)=SUM(4,I,J,K)+RNOAX
      SSQ(4,I,J,K)=SSQ(4,I,J,K)+RNOAX**2
      SUM(5,I,J,K)=SUM(5,I,J,K)+RNOAXC
      SSQ(5,I,J,K)=SSQ(5,I,J,K)+RNOAXC**2
      SUM(6,I,J,K)=SUM(6,I,J,K)+EALTAX
      SSQ(6,I,J,K)=SSQ(6,I,J,K)+EALTAX**2
      SUM(7,I,J,K)=SUM(7,I,J,K)+EALCTA
      SSQ(7,I,J,K)=SSQ(7,I,J,K)+EALCTA**2
      SUM(8,I,J,K)=SUM(8,I,J,K)+EALC
```

```
      SSQ(8,I,J,K)=SSQ(8,I,J,K)+EALC**2
      SUM(9,I,J,K)=SUM(9,I,J,K)+EALT
      SSQ(9,I,J,K)=SSQ(9,I,J,K)+EALT**2
      SUM(10,I,J,K)=SUM(10,I,J,K)+EALCT
      SSQ(10,I,J,K)=SSQ(10,I,J,K)+EALCT**2
      SUM(11,I,J,K)=SUM(11,I,J,K)+TEAL
      SSQ(11,I,J,K)=SSQ(11,I,J,K)+TEAL**2
  745 CONTINUE
  750 CONTINUE
  755 CONTINUE
      GO TO 775
  760 CONTINUE
C
C     ****************************************************************
C     * STATEMENTS 936 THRU 966 CALCULATE THE SUMS AND THE SUM OF THE  *
C     * SQUARES FOR THOSE STATIONS WHICH ARE COAL-HAUL STATONS.       *
C     ****************************************************************
C
CCCC ADDED NEXT LINE
      AADTC=1
      DO 770 I=1,2
      DO 765 J=1,3
      IF (AADTC.NE.I.OR.FRCOAL.NE.J) GO TO 765
      CCOUNT(I,J)=CCOUNT(I,J)+1
      SUMC(1,I,J)=SUMC(1,I,J)+AADT
      SSQC(1,I,J)=SSQC(1,I,J)+AADT**2
      SUMC(2,I,J)=SUMC(2,I,J)+FT
      SSQC(2,I,J)=SSQC(2,I,J)+FT**2
      SUMC(3,I,J)=SUMC(3,I,J)+FCT
      SSQC(3,I,J)=SSQC(3,I,J)+FCT**2
      SUMC(4,I,J)=SUMC(4,I,J)+RNOAX
      SSQC(4,I,J)=SSQC(4,I,J)+RNOAX**2
      SUMC(5,I,J)=SUMC(5,I,J)+RNOAXC
      SSQC(5,I,J)=SSQC(5,I,J)+RNOAXC**2
      SUMC(6,I,J)=SUMC(6,I,J)+EALTAX
      SSQC(6,I,J)=SSQC(6,I,J)+EALTAX**2
      SUMC(7,I,J)=SUMC(7,I,J)+EALCTA
      SSQC(7,I,J)=SSQC(7,I,J)+EALCTA**2
      SUMC(8,I,J)=SUMC(8,I,J)+EALC
      SSQC(8,I,J)=SSQC(8,I,J)+EALC**2
      SUMC(9,I,J)=SUMC(9,I,J)+EALT
      SSQC(9,I,J)=SSQC(9,I,J)+EALT**2
      SUMC(10,I,J)=SUMC(10,I,J)+EALCT
      SSQC(10,I,J)=SSQC(10,I,J)+EALCT**2
      SUMC(11,I,J)=SUMC(11,I,J)+TEAL
      SSQC(11,I,J)=SSQC(11,I,J)+TEAL**2
C     WRITE(9,12181) SUMC(7,I,J),I,J
C12181 FORMAT(' SUMC(7,I,J)  ',F10.4,2I5)
  765 CONTINUE
  770 CONTINUE
  775 CONTINUE
      GO TO 560
  780 CONTINUE
```

```
      IF (YR.EQ.0) YR = ICYEAR
C
C     ****************************************************************
C     * STATEMENTS 973 THRU 993  CALCULATE THE MEANS AND STANDARD DEV- *
C     * IATIONS FOR EACH CELL IN THE NON-COAL-HAUL MATRIX.            *
C     ****************************************************************
C
      DO 815 I=1,11
      DO 810 J=1,5
      DO 805 K=1,2
      DO 800 L=1,4
      IF(I.EQ.5.OR.I.EQ.7)GOTO 785
      COUNT(J,K,L)=COUN(J,K,L)
      GO TO 790
  785 COUNT(J,K,L)=COUNTC(J,K,L)
  790 IF (COUNT(J,K,L).EQ.0.0)GO TO 800
      IF (COUNT(J,K,L).EQ.1.0)GO TO 795
      MEAN(I,J,K,L)=SUM(I,J,K,L)/COUNT(J,K,L)
      STD(I,J,K,L)=(COUNT(J,K,L)*SSQ(I,J,K,L)-SUM(I,J,K,L)**2)/
     +             (COUNT(J,K,L)*(COUNT(J,K,L)-1.))
      IF(STD(I,J,K,L).LT.0.001.AND.STD(I,J,K,L).GT.-.001)STD(I,J,K,L)=0.
      STD(I,J,K,L)=SQRT(ABS(STD(I,J,K,L)))
      GO TO 800
  795 MEAN(I,J,K,L)=SUM(I,J,K,L)/COUNT(J,K,L)
  800 CONTINUE
  805 CONTINUE
  810 CONTINUE
  815 CONTINUE
C
C     ****************************************************************
C     * STATEMENTS 1000 THRU 1014 CALCULATE THE MEANS AND STANDARD DEV-*
C     * IATIONS FOR EACH CELL IN THE COAL-HAUL MATRIX.                *
C     ****************************************************************
C
      DO 835 I=1,11
      DO 830 J=1,2
      DO 825 K=1,3
C     WRITE(9,33533)CCOUNT(J,K),FRCOAL
C3533 FORMAT(' CCOUNT = ',F10.3,' FRCOAL =',I10)
      IF (CCOUNT(J,K).EQ.0.0)GO TO 825
      IF (CCOUNT(J,K).EQ.1.0)GO TO 820
      MEANC(I,J,K)=SUMC(I,J,K)/CCOUNT(J,K)
      STDC(I,J,K)=(CCOUNT(J,K)*SSQC(I,J,K)-SUMC(I,J,K)**2)/
     +             (CCOUNT(J,K)*(CCOUNT(J,K)-1.))
      IF(STDC(I,J,K).LT.0.001.AND.STDC(I,J,K).GT.-0.001)STDC(I,J,K)=0.
      STDC(I,J,K)=SQRT(ABS(STDC(I,J,K)))
      GO TO 825
  820 MEANC(I,J,K)=SUMC(I,J,K)/CCOUNT(J,K)
      IF (I.EQ.7) THEN
C     WRITE(9,72089) MEANC(I,J,K),SUMC(I,J,K),CCOUNT(J,K)
C2089 FORMAT(' MEANC = ',F10.4,' SUMC = ',F10.3,'CCOUNT = ',F10.3)
      ENDIF
  825 CONTINUE
```

```
  830 CONTINUE
  835 CONTINUE
C
C     ****************************************************************
C     * STATEMENTS 1022 THRU 1046 WRITE THE MEANS,NUMBER OF STATIONS   *
C     * AND STANDARD DEVIATIONS OF THE TRAFFIC PARAMETERS FOR THE      *
C     * CURRENT YEAR ON TAPE.                                          *
C     ****************************************************************
C
CCCC CHANGED LOOP COUNTER 860 5 TO 1, 855 2 TO 1, 850 4 TO 1   3-16-90
      OPEN(16,FILE=FMNEW,STATUS='UNKNOWN')
      DO 875 I=1,11
      DO 860 J=1,1
      DO 855 K=1,1
      DO 850 L=1,1
      IF (I.EQ.5.OR.I.EQ.7.OR.YR.LT.80) GOTO 840
      COUNT(J,K,L)=COUN(J,K,L)
      GO TO 845
  840 COUNT(J,K,L)=COUNTC(J,K,L)
  845 WRITE (16,5225) YR,I,J,K,L,COUNT(J,K,L),MEAN(I,J,K,L),STD(I,J,K,L)
     1,IWSTA
 5225 FORMAT(I2,4I1,F4.0,2F15.3,T46,I5)
  850 CONTINUE
  855 CONTINUE
  860 CONTINUE
CCCC CHANGED LOOP COUNTER  870 2 TO 1, 865 3 TO 1   3-16-90
      DO 870 J=1,1
      DO 865 K=1,1
      WRITE (16,5250) YR,I,J,K,CCOUNT(J,K),MEANC(I,J,K),STDC(I,J,K)
     1,IWSTA
 5250 FORMAT(I2,3I1,1X,F4.0,2F15.3,T46,I5)
  865 CONTINUE
  870 CONTINUE
  875 CONTINUE
      END FILE 16
      IF(YR.EQ.79) GO TO 1550
      Z=0
  880 CONTINUE
      Z=Z+1
      FMEAT=FMEAN(Z)
      OPEN(16,FILE=FMEAT,STATUS='OLD',BLANK='ZERO')
C
C     ****************************************************************
C     * STATEMENTS 1053 THRU 1072 READ THE MEANS OF THE TRAFFIC PARA-  *
C     * METERS FOR 15 PREVIOUS YEARS.                                  *
C     ****************************************************************
C
CCCC CHANGED LOOP COUNTER 895 5 TO 1, 890 2 TO 1, 885 4 TO 1   3-16-90
      DO 910 I=1,11
      DO 895 J=1,1
      DO 890 K=1,1
      DO 885 L=1,1
      READ (16,5275) NYR(Z),CNTYR(I,J,K,L,Z),MEANY(I,J,K,L,Z),IIWSTT
```

```
      IF(IIWSTA(Z).EQ.0)IIWSTA(Z)=IIWSTT
 5275 FORMAT(I2,T7,F4.0,T11,F15.3,T46,I5)
  885 CONTINUE
  890 CONTINUE
  895 CONTINUE
CCCC CHANGED LOOP COUNTER 905 2 TO 1, 900 3 TO 1   3-16-90
      DO 905 J=1,1
      DO 900 K=1,1
      READ (16,5300) CNTCYR(I,J,K,Z),MEANCY(I,J,K,Z)
 5300 FORMAT(T7,F4.0,T11,F15.3)
  900 CONTINUE
  905 CONTINUE
  910 CONTINUE
C     ENDFILE 16
      READ(16,913,END=914)DUM1
  913 FORMAT(A3)
  914 CONTINUE
CCCC CHANGED 2-22-90 TO LIMIT FOR FC DATA
C     IF(NYR(Z).EQ.69.OR.Z.EQ.15) GO TO 915
      IF(NYR(Z).EQ.79.OR.Z.EQ.NNYR) GO TO 915
      GO TO880
  915 CONTINUE
C
C     ****************************************************************
C     * STATEMENTS 1080 THRU 1104 CALCULATE THE DIFFERENCES IN THE     *
C     * TRAFFIC PARAMETERS FROM EACH OF THE 15 PREVIOUS YEARS TO THE   *
C     * CURRENT YEAR.                                                  *
C     ****************************************************************
C
      DO 965 I=1,11
      DO 940 J=1,5
      DO 935 K=1,2
      DO 930 L=1,4
      DO 925 M=1,15
      IF(COUNT(J,K,L).EQ.0.0.OR.CNTYR(I,J,K,L,M).EQ.0.0)GO TO 920
      DIFF(I,J,K,L,M)=MEAN(I,J,K,L)-MEANY(I,J,K,L,M)
      GO TO 925
  920 DIFF(I,J,K,L,M)=1000000000000.
  925 CONTINUE
  930 CONTINUE
  935 CONTINUE
  940 CONTINUE
      DO 960 J=1,2
      DO 955 K=1,3
      DO 950 L=1,15
      IF(CCOUNT(J,K).EQ.0.0.OR.CNTCYR(I,J,K,L).EQ.0.0)GO TO 945
      DIFFC(I,J,K,L)=MEANC(I,J,K)-MEANCY(I,J,K,L)
      GO TO 950
  945 DIFFC(I,J,K,L)=1000000000000.
  950 CONTINUE
  955 CONTINUE
  960 CONTINUE
  965 CONTINUE
```

```
      IF(YR.GE.89) THEN
      DO 72090 IA=1,11
      ASUM(IA)=SUM(IA,1,1,1)+MEANY(IA,1,1,1,1)*CNTYR(IA,1,1,1,1)+MEANY
     1(IA,1,1,1,2)*CNTYR(IA,1,1,1,2)
C     WRITE(8,72015)SUM(IA,1,1,1),MEANY(IA,1,1,1,1),CNTYR(IA,1,1,1,1),
C    1MEANY(IA,1,1,1,2),CNTYR(IA,1,1,1,2),COUNT(1,1,1),IA
72015 FORMAT(6F10.3,I5)
C     ACNT(IA)=COUNT(1,1,1)+CNTYR(IA,1,1,1,1)+CNTYR(IA,1,1,1,2)
      ACNT(IA)=COUNT(1,1,1)
       IF(MEANY(IA,1,1,1,1).GT.0) THEN
      ACNT(IA)=ACNT(IA)+CNTYR(IA,1,1,1,1)
       ENDIF
       IF(MEANY(IA,1,1,1,2).GT.0) THEN
      ACNT(IA)=ACNT(IA)+CNTYR(IA,1,1,1,2)
       ENDIF
      IF(ACNT(IA).GT.0) THEN
      AMEAN(IA)=ASUM(IA)/ACNT(IA)
      ENDIF
      ASUMC(IA)=SUMC(IA,1,1)+MEANCY(IA,1,1,1)*CNTCYR(IA,1,1,1)+MEANCY
     1(IA,1,1,2)*CNTCYR(IA,1,1,2)
c     WRITE(8,72015)SUMC(IA,1,1),MEANCY(IA,1,1,1),CNTCYR(IA,1,1,1),
c    1MEANCY(IA,1,1,2),CNTCYR(IA,1,1,2),CCOUNT(1,1),IA
C     ACNTC(IA)=CCOUNT(1,1)+CNTCYR(IA,1,1,1)+CNTCYR(IA,1,1,2)
      ACNTC(IA)=CCOUNT(1,1)
       IF(MEANCY(IA,1,1,1).GT.0) THEN
      ACNTC(IA)=ACNTC(IA)+CNTCYR(IA,1,1,1)
       ENDIF
       IF(MEANCY(IA,1,1,2).GT.0) THEN
      ACNTC(IA)=ACNTC(IA)+CNTCYR(IA,1,1,2)
       ENDIF
      IF(ACNTC(IA).GT.0) THEN
      AMEANC(IA)=ASUMC(IA)/ACNTC(IA)
      ENDIF
c     WRITE(8,71114)ASUM(IA),ACNT(IA),ASUMC(IA),ACNTC(IA)
c71114 FORMAT(' ASUM = ', F10.3,'  ACNT =  ',F10.3,' ASUMC = ',F10.3,
c    1' ACNTC = ',F10.3)
c     WRITE(8,71112)AMEAN(IA),AMEANC(IA),IA
c71112 FORMAT(' AMEAN = ', F10.3,'  AMEANC =  ',F10.3,I5)
72090 CONTINUE
      ENDIF
C     IF(IFF.EQ.1)GO TO 1250
C1250 CONTINUE
C
C     ****************************************************************
C     * STATEMENTS 1111 THRU 1550 PRINT THE MATRICES OF MEANS AND IN-  *
C     * CREMENTS.                                                      *
C     ****************************************************************
C
CCCC  ADDED NEXT LINE TO SUPRESS PRINTOUT
      IF(IFF.EQ.0)GO TO 42651
CCCC CHANGED LOOP COUNTER 1075 5 TO 1, 1070 2 TO 1, 1065 4 TO 1 3-16-90
      DO 1115 I=1,11
      NO=0
```

```
      DO 1075 J=1,1
      DO 1070 K=1,1
      DO 1065 L=1,1
      NO=NO+1
      IF(NO.NE.1) GO TO 1030
      GO TO (970,975,980,985,990,995,1000,1005,1010,1015,1020),I
  970 WRITE(6,5325)
 5325 FORMAT('1',T51,'ANNUAL AVERAGE DAILY TRAFFIC')
      GO TO 1025
  975 WRITE(6,5350)
 5350 FORMAT('1',T58,'PERCENT TRUCKS')
      GO TO 1025
  980 WRITE(6,5375)
 5375 FORMAT('1',T50,'PERCENT OF TRUCKS HAULING COAL')
      GO TO 1025
  985 WRITE(6,5400)
 5400 FORMAT('1',T49,'AXLES PER TRUCK (NON-COAL-HAULING)')
      GO TO 1025
  990 WRITE(6,5425)
 5425 FORMAT('1',T50,'AXLES PER TRUCK (COAL-HAULING)')
      GO TO 1025
  995 WRITE(6,5450)
 5450 FORMAT('1',T45,'EAL"S PER TRUCK AXLE (NON-COAL-HAULING)')
      GO TO 1025
 1000 WRITE(6,5475)
 5475 FORMAT('1',T47,'EAL"S PER TRUCK AXLE (COAL-HAULING)')
      GO TO 1025
 1005 WRITE(6,5500)
 5500 FORMAT('1',T40,'2-DIRECTION EAL"S DUE TO 4-TIRED VEHICLES IN 1000
     +"S')
      GO TO 1025
 1010 WRITE(6,5525)
 5525 FORMAT('1',T35,'2-DIRECTION EAL"S DUE TO NON-COAL-HAULING VEHICLE
     +S IN 1000"S')
      GO TO 1025
 1015 WRITE(6,5550)
 5550 FORMAT('1',T38,'2-DIRECTION EAL"S DUE TO COAL-HAULING VEHICLES IN
     + 1000"S')
      GO TO 1025
 1020 WRITE(6,5575)
 5575 FORMAT('1',T54,'TOTAL 2-DIRECTION EAL"S')
 1025 CONTINUE
      WRITE(6,6175)
      WRITE(6,5600)
 5600 FORMAT(T56,'NON-COAL-HAULING ROADS')
      WRITE(6,5625)
 5625 FORMAT(T4,'LOCAL',T20,'STD')
      WRITE(6,5650)
 5650 FORMAT(' ','CONDITION',T13,'NO OF',' DEV',T59,'AVERAGE VALUE')
      WRITE(6,5675) YR,YR,(NYR(N),N=1,NNYR)
 5675 FORMAT(' FA',T5,'VOL',T10,'GA',T13,'STAS.',I5,T29,I2,14I7)
 1030 CONTINUE
      IF(I.EQ.5.OR.I.EQ.7)GOTO 1035
```

```
      COUNT(J,K,L)=COUN(J,K,L)
      GO TO 1040
 1035 COUNT(J,K,L)=COUNTC(J,K,L)
 1040 IF(MEAN(I,J,K,L).EQ.0.AND.COUNT(J,K,L).EQ.0.0)MEAN(I,J,K,L)=10000
     +000000.
      DO 1045 M=1,14
      IF(MEANY(I,J,K,L,M).EQ.0.AND.CNTYR(I,J,K,L,M).EQ.0.)MEANY(I,J,K,L,
     +M)=100000000.
 1045 CONTINUE
      GO TO (1050,1055,1055,1055,1055,1055,1055,1050,1050,1050,1050),I
 1050 WRITE (6,5700)J,K,L,COUNT(J,K,L),STD(I,J,K,L),MEAN(I,J,K,L),(MEANY
     +(I,J,K,L,M),M=1,NNYR)
 5700 FORMAT(T3,I1,T7,I1,T11,I1,T14,F4.0,F8.0,15F7.0)
      GO TO 1060
 1055 WRITE (6,5725)J,K,L,COUNT(J,K,L),STD(I,J,K,L),MEAN(I,J,K,L),(MEANY
     +(I,J,K,L,M),M=1,NNYR)
 5725 FORMAT(T3,I1,T7,I1,T11,I1,T14,F4.0,F8.3,15F7.3)
 1060 CONTINUE
 1065 CONTINUE
 1070 CONTINUE
 1075 CONTINUE
CCCC CHANGED LOOP COUNTER 1110 2 TO 1, 1105 3 TO 1,   3-16-90
      NO1=0
      DO 1110 J=1,1
      DO 1105 K=1,1
      NO1=NO1+1
      IF (NO1.NE.1) GO TO 1080
      WRITE(6,6175)
      WRITE(6,5750)
 5750 FORMAT(T57,'COAL-HAULING ROADS')
      WRITE(6,5625)
      WRITE(6,5650)
      WRITE(6,5775) YR,YR,(NYR(N),N=1,NNYR)
 5775 FORMAT(T5,'VOL',T9,'CT',T13,'STAS.',I5,T29,I2,14I7)
 1080 CONTINUE
      IF(MEANC(I,J,K).EQ.0.AND.CCOUNT(J,K).EQ.0.)MEANC(I,J,K)=100000000.
      DO 1085 L=1,14
      IF(MEANCY(I,J,K,L).EQ.0.AND.CNTCYR(I,J,K,L).EQ.0.)MEANCY(I,J,K,L)=
     +1000000000.
 1085 CONTINUE
      GO TO (1090,1095,1095,1095,1095,1095,1095,1090,1090,1090,1090),I
 1090 WRITE (6,5800)J,K,CCOUNT(J,K),STDC(I,J,K),MEANC(I,J,K),(
     +MEANCY(I,J,K,L),L=1,NNYR)
 5800 FORMAT(T7,I1,T10,I1,T16,F3.0,F8.0,15F7.0)
      GO TO 1100
 1095 WRITE (6,5825)J,K,CCOUNT(J,K),STDC(I,J,K),MEANC(I,J,K),(MEANCY
     +(I,J,K,L),L=1,NNYR)
 5825 FORMAT(T7,I1,T10,I1,T16,F3.0,F8.3,15F7.3)
 1100 CONTINUE
 1105 CONTINUE
 1110 CONTINUE
 1115 CONTINUE
42651 CONTINUE
```

```
CC      DO 71111 IA=1,11
CC    AMEAN(IA)=(MEAN(IA,1,1,1)+MEANY(IA,1,1,1,1)+MEANY(IA,1,1,1,2))/3
CC    AMEANC(IA)=(MEANC(IA,1,1)+MEANCY(IA,1,1,1)+MEANCY(IA,1,1,2))/3
CC      WRITE(8,71112)AMEAN(IA),AMEANC(IA),IA
CC71112 FORMAT(' AMEAN = ', F10.3,'  AMEANC =  ',F10.3,I5)
CC71111 CONTINUE
      IFCLAS=IFCL
      IF(IFCLAS.EQ.0) IFCLAS=IINFC
      IF (YR.LT.89) THEN
C     WRITE(6,21450)IFCLAS
C1450 FORMAT('1',T35,'SUMMARY OF AVERAGE VALUES FOR FUNCTIONAL CLASS  ',
C    +I2)
c     CALL FCHEAD(IFCLAS,2)
      CALL FCHEAD(IINFC,2)
      WRITE(6,6175)
      WRITE(6,25600)
      WRITE(6,25603)
      WRITE(6,25602)
25600 FORMAT(T60,'UNCLASSIFIED ROADS')
25603 FORMAT(T45,'(ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS')
25602 FORMAT(T46,'THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)')
CCCC ALL LOCATIONS
C     WRITE(6,5325)
      WRITE(6,25625)
25625 FORMAT(T20,'STD')
      WRITE(6,25650)
25650 FORMAT(' ',T13,'NO OF',' DEV',T59,'AADT')
      WRITE(6,25675) YR,YR,(NYR(N),N=1,NNYR)
25675 FORMAT(T13,'STAS.',I5,T29,I2,14I7)
      WRITE(6,21451)COUNT(1,1,1),STD(1,1,1,1),MEAN(1,1,1,1),(MEANY
     +(1,1,1,1,M),M=1,NNYR)
21451 FORMAT(T14,F4.0,F8.0,15F7.0)
      WRITE(6,25350)
25350 FORMAT(' ',T48,'PERCENT TRUCKS')
      WRITE(6,21452)COUNT(1,1,1),STD(2,1,1,1),MEAN(2,1,1,1),(MEANY
     +(2,1,1,1,M),M=1,NNYR)
C5725 FORMAT(T3,I1,T7,I1,T11,I1,T14,F4.0,F8.3,15F7.3)
21452 FORMAT(T14,F4.0,F8.3,15F7.3)
      WRITE(6,25500)
25500 FORMAT(' ',T40,'2-DIRECTION EAL"S DUE TO 4-TIRED VEHICLES IN 1000
     +"S')
      WRITE(6,21451)COUNT(1,1,1),STD(8,1,1,1),MEAN(8,1,1,1),(MEANY
     +(8,1,1,1,M),M=1,NNYR)
      WRITE(6,25401)
25401 FORMAT(' ',T43,'AXLES PER TRUCK (UNCLASSIFIED)')
      WRITE(6,21452)COUNT(1,1,1),STD(4,1,1,1),MEAN(4,1,1,1),(MEANY
     +(4,1,1,1,M),M=1,NNYR)
      WRITE(6,25451)
25451 FORMAT(' ',T40,'EAL"S PER TRUCK AXLE (UNCLASSIFIED)')
      WRITE(6,21452)COUNT(1,1,1),STD(6,1,1,1),MEAN(6,1,1,1),(MEANY
     +(6,1,1,1,M),M=1,NNYR)
      WRITE(6,25526)
25526 FORMAT(' ',T35,'2-DIRECTION EAL"S DUE TO UNCLASSIFIED VEHICLES IN
```

```
     +1000"S')
      WRITE(6,21451)COUNT(1,1,1),STD(9,1,1,1),MEAN(9,1,1,1),(MEANY
     +(9,1,1,1,M),M=1,NNYR)
CCCC SECOND LEVEL  HEAVYWEIGHT
      WRITE(6,6175)
      WRITE(6,6175)
      WRITE(6,25601)
25601 FORMAT(T56,'CLASSIFIED (HEAVY/COAL) ROADS')
      WRITE(6,25625)
      WRITE(6,25650)
      WRITE(6,25675) YR,YR,(NYR(N),N=1,NNYR)
      WRITE (6,25800)CCOUNT(1,1),STDC(1,1,1),MEANC(1,1,1),(MEANCY
     +(1,1,1,L),L=1,NNYR)
      WRITE(6,25350)
      WRITE (6,25825)CCOUNT(1,1),STDC(2,1,1),MEANC(2,1,1),(MEANCY
     +(2,1,1,L),L=1,NNYR)
      WRITE(6,25500)
      WRITE (6,25800)CCOUNT(1,1),STDC(8,1,1),MEANC(8,1,1),(MEANCY
     +(8,1,1,L),L=1,NNYR)
      WRITE(6,25375)
25375 FORMAT(' ',T42,'PERCENT OF TRUCKS (HEAVY/COAL)')
      WRITE (6,25825)CCOUNT(1,1),STDC(3,1,1),MEANC(3,1,1),(MEANCY
     +(3,1,1,L),L=1,NNYR)
      WRITE(6,25400)
25400 FORMAT(' ',T45,'AXLES PER TRUCK (NORMAL)')
      WRITE (6,25825)CCOUNT(1,1),STDC(4,1,1),MEANC(4,1,1),(MEANCY
     +(4,1,1,L),L=1,NNYR)
      WRITE(6,25425)
25425 FORMAT(' ',T44,'AXLES PER TRUCK (HEAVY/COAL)')
      WRITE (6,25825)CCOUNT(1,1),STDC(5,1,1),MEANC(5,1,1),(MEANCY
     +(5,1,1,L),L=1,NNYR)
      WRITE(6,25450)
25450 FORMAT(' ',T43,'EAL"S PER TRUCK AXLE (NORMAL)')
      WRITE (6,25825)CCOUNT(1,1),STDC(6,1,1),MEANC(6,1,1),(MEANCY
     +(6,1,1,L),L=1,NNYR)
      WRITE(6,25475)
25475 FORMAT(' ',T41,'EAL"S PER TRUCK AXLE (HEAVY/COAL)')
      WRITE (6,25825)CCOUNT(1,1),STDC(7,1,1),MEANC(7,1,1),(MEANCY
     +(7,1,1,L),L=1,NNYR)
      WRITE(6,25525)
25525 FORMAT(' ',T35,'2-DIRECTION EAL"S DUE TO NORMAL VEHICLES IN 1000
     +"S')
      WRITE (6,25800)CCOUNT(1,1),STDC(9,1,1),MEANC(9,1,1),(MEANCY
     +(9,1,1,L),L=1,NNYR)
      WRITE(6,25550)
25550 FORMAT(' ',T38,'2-DIRECTION EAL"S DUE TO (HEAVY/COAL) VEHICLES IN
     + 1000"S')
      WRITE (6,25800)CCOUNT(1,1),STDC(10,1,1),MEANC(10,1,1),(MEANCY
     +(10,1,1,L),L=1,NNYR)
      ELSE
      CALL FCHEAD(IINFC,2)
C     WRITE(6,21450)IFCLAS
CCCC ALL LOCATIONS
```

```
      WRITE(6,6175)
 6175 FORMAT(' ')
      WRITE(6,6175)
      WRITE(6,35650)
35650 FORMAT(' ',T28,'       3 YR')
C5650 FORMAT(' ',T13,'NO OF  3 YR',T59,'AADT')
      WRITE(6,35675) YR,(NYR(N),N=1,NNYR)
      WRITE(6,6175)
      WRITE(6,61759)IWSTA,(IIWSTA(M),M=1,2)
c      WRITE(6,61759)IWSTA,(IIWSTA(M),M=1,nnyr)
      WRITE(6,6175)
      WRITE(6,25600)
      WRITE(6,25603)
      WRITE(6,25602)
      WRITE(6,6175)
61759 FORMAT(' NUMBER OF WEIGH STA.',T40,12I7)
35675 FORMAT(' YEAR',T28,'        AVG ',2X,I5,14I7)
      WRITE(6,38952)INT(ACNT(1)),INT(COUNT(1,1,1)),(INT(CNTYR
     +(1,1,1,1,M)),M=1,NNYR)
31752 FORMAT(' NUMBER OF STATIONS',T33,12I7)
38952 FORMAT(' NO OF CLASSIFICATION STA.',T33,12I7)
      WRITE(6,6175)
      WRITE(6,31451)INT(AMEAN(1)),INT(MEAN(1,1,1,1)),(INT(MEANY
     +(1,1,1,1,M)),M=1,NNYR)
      WRITE(6,6175)
31451 FORMAT(' AADT',T33,15I7)
C31451 FORMAT(T18,15F7.0)
C      WRITE(6,25350)
      WRITE(6,31452)AMEAN(2),MEAN(2,1,1,1),(MEANY
     +(2,1,1,1,M),M=1,NNYR)
      WRITE(6,6175)
31452 FORMAT(' PERCENT TRUCKS',T33,15F7.3)
C     WRITE(6,25500)
C     WRITE(6,31451)AMEAN(8),MEAN(8,1,1,1),(MEANY
C    +(8,1,1,1,M),M=1,NNYR)
C     WRITE(6,25401)
      WRITE(6,31552)AMEAN(4),MEAN(4,1,1,1),(MEANY
     +(4,1,1,1,M),M=1,NNYR)
      WRITE(6,6175)
31552 FORMAT(' AXLES PER TRUCK',T33,15F7.3)
C     WRITE(6,25451)
      WRITE(6,31652)AMEAN(6),MEAN(6,1,1,1),(MEANY
     +(6,1,1,1,M),M=1,NNYR)
      WRITE(6,6175)
31652 FORMAT(' EAL"S PER TRUCK AXLE ',T33,15F7.3)
C     WRITE(6,25526)
C     WRITE(6,31451)ACNT(1),AMEAN(9),MEAN(9,1,1,1),(MEANY
C    +(9,1,1,1,M),M=1,NNYR)
CCCC SECOND LEVEL  HEAVYWEIGHT
C     WRITE(6,6175)
      WRITE(6,6175)
      WRITE(6,25611)
      WRITE(6,25612)
```

```
      WRITE(6,25613)
25611 FORMAT(T63,'CLASSIFIED ROADS')
25612 FORMAT(T54,'(MANUAL LOCATION WITH 3% OR MORE OF ')
25613 FORMAT(T55,'TRUCKS CLASSIFIED AS HEAVY/COAL) ')
C     WRITE(6,35650)
C     WRITE(6,35675) YR,(NYR(N),N=1,NNYR)
      WRITE(6,6175)
C     WRITE(6,35751)
      WRITE(6,38952)INT(ACNTC(1)),INT(CCOUNT(1,1)),(INT(CNTCYR
     +(1,1,1,M)),M=1,NNYR)
      WRITE(6,6175)
      WRITE(6,6175)
C     WRITE(6,35750)
      WRITE (6,31451)INT(AMEANC(1)),INT(MEANC(1,1,1)),(INT(MEANCY
     +(1,1,1,L)),L=1,NNYR)
      WRITE(6,6175)
      WRITE(6,6175)
C     WRITE(6,25350)
      WRITE (6,31452)AMEANC(2),MEANC(2,1,1),(MEANCY
     +(2,1,1,L),L=1,NNYR)
      WRITE(6,6175)
      WRITE(6,6175)
C     WRITE(6,25500)
C     WRITE (6,35800)AMEANC(8),MEANC(8,1,1),(MEANCY
C    +(8,1,1,L),L=1,NNYR)
C     WRITE(6,25375)
      WRITE (6,35829)AMEANC(3),MEANC(3,1,1),(MEANCY
     +(3,1,1,L),L=1,NNYR)
C     WRITE(6,6175)
      WRITE(6,35830)
C     WRITE(6,25400)
      WRITE (6,35825)AMEANC(4),MEANC(4,1,1),(MEANCY
     +(4,1,1,L),L=1,NNYR)
      WRITE(6,35827)
C     WRITE(6,6175)
C     WRITE(6,25425)
      WRITE (6,35825)AMEANC(5),MEANC(5,1,1),(MEANCY
     +(5,1,1,L),L=1,NNYR)
      WRITE(6,35828)
C     WRITE(6,6175)
C     WRITE(6,25450)
      WRITE (6,35826)AMEANC(6),MEANC(6,1,1),(MEANCY
     +(6,1,1,L),L=1,NNYR)
      WRITE(6,35827)
C     WRITE(6,6175)
C     WRITE(6,25475)
      WRITE (6,35826)AMEANC(7),MEANC(7,1,1),(MEANCY
     +(7,1,1,L),L=1,NNYR)
      WRITE(6,35828)
C     WRITE(6,6175)
C     WRITE(6,25525)
C     WRITE (6,35800)AMEANC(9),MEANC(9,1,1),(MEANCY
C    +(9,1,1,L),L=1,NNYR)
```

```
C     WRITE(6,25550)
C     WRITE (6,35800)AMEANC(10),MEANC(10,1,1),(MEANCY
C    +(10,1,1,L),L=1,NNYR)
      WRITE(6,6175)
CC    WRITE(6,42490)YR,YR,YR-1,YR-2,IFCLAS
CC490 FORMAT(T15,' AVERAGE VALUES FOR 19',I2,' REFLECT A AVERAGE OF STAT
CC   1IONS',/,T16,'COLLECTED IN 19',I2,', 19',I2,', AND 19',I2,' FOR FUN
CC   2CTIONAL CLASS ',I2)
42490 FORMAT(T15,' AVERAGE VALUES AND NUMBER OF STATIONS FOR 19',I2,/,T1
     15,' REFLECT THE STATIONS COLLECTED IN 19',I2,', 19',I2,', AND 19',
     2I2,/,T15,' FOR FUNCTIONAL CLASS ',I2)
      ENDIF
25800 FORMAT(T15,F3.0,F8.0,15F7.0)
25825 FORMAT(T15,F3.0,F8.3,15F7.3)
35800 FORMAT(T18,15F7.0)
35825 FORMAT(' AXLES PER TRUCK',T33,15F7.3)
35826 FORMAT(' EAL"S PER TRUCK AXLE',T33,15F7.3)
35827 FORMAT(' NORMAL',/)
35828 FORMAT(' HEAVY/COAL',/)
35829 FORMAT(' PERCENT OF TRUCKS',T33,15F7.3)
35830 FORMAT(' CLASSIFIED AS HEAVY/COAL',/)
      IF(INCR.EQ.0)GO TO 1550
 1550 CONTINUE
      STOP
      END
      SUBROUTINE FCHEAD(IFCNUM,IPTT)
      INTEGER FC,IPT
      FC=IFCNUM
      IPT=IPTT
C     WRITE(8,34567) FC,IPT
C4567 FORMAT('IFC = ',I2,I15)
      IF(IPT.EQ.2)WRITE (6,1220)
 1220 FORMAT(T45,' SUMMARY OF AVERAGE VALUES FOR')
      IF (FC.EQ.1) THEN
      WRITE(6,1221)
 1221 FORMAT(T35,' AGGREGATE CLASS I -- RURAL INTERSTATE')
      ELSEIF (FC.EQ.2) THEN
      WRITE(6,1222)
 1222 FORMAT(T35,' AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/')
      WRITE(6,12221)
12221 FORMAT(T35,'                    RURAL MINOR ARTERIAL')
      ELSEIF (FC.EQ.3) THEN
      WRITE(6,1223)
 1223 FORMAT(T35,' AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/')
      WRITE(6,12231)
12231 FORMAT(T35,'                     RURAL MINOR COLLECTOR')
      ELSEIF (FC.EQ.4) THEN
      WRITE(6,1224)
 1224 FORMAT(T35,' AGGREGATE CLASS IV -- URBAN INTERSTATE')
      ELSEIF (FC.EQ.5) THEN
      WRITE(6,1225)
 1225 FORMAT(T35,' AGGREGATE CLASS V -- URBAN OTHER FREEWAY AND EXPRESSW
     1AYS/')
```

```
      WRITE(6,12251)
12251 FORMAT(T35,'                  URBAN OTHER PRINCIPAL ARTERIAL')
      ELSEIF (FC.EQ.6) THEN
      WRITE(6,1226)
 1226 FORMAT(T35,' AGGREGATE CLASS VI --  URBAN MINOR ARTERIAL/')
      WRITE(6,12261)
12261 FORMAT(T35,'                   URBAN COLLECTOR')
      END IF
C     WRITE (6,1228)
 1228 FORMAT('  ')
      RETURN
      END
```

SMOOTH.FOR

```
$DEBUG
C  ***************************************************************
C  *
C  *  PROGRAM NAME : SMOOTH
C  *
C  *
C  *  PROGRAM  FUNCTION :  THIS  PROGRAM  USES  THE  OUTPUT  FILE
C  *      FROM THE EALCAL PROGRAM.  IF AT LEAST FOUR DATA POINTS
C  *      ARE FOUND IN  A TEN YEAR PERIOD,  A STRAIGHT  LINE LEAST
C  *      SQUARES FIT IS COMPUTED  FOR THE POINTS AND NEW VALUES
C  *      ARE COMPUTED FOR EVERY YEAR UP TO TWO YEARS BEYOND THE
C  *      EARLIEST OR LATEST DATA POINT.
C  *
C  *      REGRESSION ANALYSES IS CONDUCTED TO PRODUCE SMOOTHED
C  *      VALUES FOR EACH PARAMETER OF INTEREST TO ELIMINATE
C  *      INTER-ANNUAL VARIABILITY THAT RESULTS FROM SITE-SPECIFIC
C  *      INFLUENCES.
C  *
C  ***************************************************************
C
      CHARACTER FIN*64,FOUT*64
      CHARACTER*12 FCLAS,FAXLE,FWGT(3),FMEAN(10),FWGTT,FMEAT,FMNEW
       REAL*8 RWT
       REAL AADT(11,6,2,4,20),OUT(20),CNTS(11,6,2,4,20),MINVAL
       REAL MEAN(11),MEANY(11,25),MEANC(11),MEANCY(11,25),WT
       INTEGER YR,CH,FA,VOL,GA,CT,MINYR,MAXYR,YRS,LO,HI,YEAR,YRHD(20)
       INTEGER CWSTA(20),WSTA
       INTEGER CCNTU(20),CCNTC(20)
       MINYR = 1000
       YRS = 6
      WRITE(*,'(A)')' INPUT AGGREGATE CLASS FOR THIS RUN'
      READ(*,'(BN,I2)') IFUNCL
      WRITE(*,'(A)')' Specify Input File Name-'
      READ(*,'(A)')FIN
      OPEN(5,FILE=FIN,STATUS='OLD',BLANK='ZERO')
      WRITE(*,'(A)')' Specify Output File Name-'
      READ(*,'(A)')FOUT
      WRITE(*,'(A)')' INPUT NUMBER OF YEARS FOR THIS RUN'
      READ(*,'(I2)') IINYR
      YRS=IINYR
      OPEN(6,FILE=FOUT,STATUS='UNKNOWN')
      OPEN(8,FILE='CHECK.LST',STATUS='UNKNOWN')
      DO 55469 I = 1,YRS
      READ(5,54441)FMEAN(I)
54441 FORMAT(A12)
55469 CONTINUE
c     YRS=3
      OLDPRT=1
C
C  ***************************************************************
C  *   STATEMENTS 29 THRU 45 READ THE YEAR,  FEDERAL AID,  VOLUME   *
```

```
C  *   CLASSIFICATION CODE, THE  GEOGRAPHIC AREA  CODE, THE COUNT   *
C  *   FOR THE NUMBER OF STATIONS IN EACH OF THE CHART CATEGORIES   *
C  *   AND THE AVERAGE VALUE FOR EACH CHART AND CATEGORY.          *
C  ****************************************************************
C
 100   DO 200 YR = 1,YRS
      FMEAT=FMEAN(YR)
      OPEN(15,FILE=FMEAT,STATUS='OLD',BLANK='ZERO')
         DO 170 CH = 1,11
C           DO 150 I = 1,40
             READ(15,5000) YEAR,FA,VOL,GA,CNT,ADT,WSTA
C             WRITE(8,5000) YEAR,FA,VOL,GA,CNT,ADT,WSTA
             AADT(CH,FA,VOL,GA,YR) = ADT
             CNTS(CH,FA,VOL,GA,YR) = CNT
             IF(CH.EQ.1)CCNTU(YR) = INT(CNT)
             IF(CH.EQ.1)CWSTA(YR) = WSTA
C150         CONTINUE
C           DO 170 I = 1,6
             READ(15,5010) VOL,CT,CNT,ADT
             AADT(CH,6,VOL,CT,YR) = ADT
             CNTS(CH,6,VOL,CT,YR) = CNT
             IF(CH.EQ.1)CCNTC(YR) = INT(CNT)
 170     CONTINUE
         IF (YEAR.LT.MINYR) MINYR = YEAR
         YRHD(YRS-YR+1) = YEAR
C     WRITE(8,12245)YRHD(YRS-YR+1),YRS,YR
C 245 FORMAT(3I10)
         READ(15,5000,END=200)
 200   CONTINUE
      MAXYR = MINYR + 9
C
C  ****************************************************************
C  *   STATEMENTS 54 THRU 55  MAKE SURE THAT ANY  ESTIMATED VALUE   *
C  *   DOES NOT FALL BELOW AN ALLOWABLE MINIMUM.   STATEMENTS  56   *
C  *   THRU 86 PRINT THE CHART HEADERS.                            *
C  ****************************************************************
C
      DO 800 CH = 1,11
      MINVAL=0.
      IF(OLDPRT.EQ.0) THEN
      IF (CH.EQ.1)CALL FCHEAD(IFUNCL)
C      IF (CH.EQ.1)WRITE(8,23456)IFUNCL
C3456  FORMAT(' IFUNCL = ',I2)
      IF (CH.EQ.4.OR.CH.EQ.5) MINVAL = 2.
      IF (CH.EQ. 1) WRITE(6,7000)
 7000  FORMAT(' ',T44,'      ANNUAL AVERAGE DAILY TRAFFIC   ')
      IF (CH.EQ. 2) WRITE(6,7100)
 7100  FORMAT(' ',T44,'             PERCENT TRUCKS          ')
      IF (CH.EQ. 3) WRITE(6,7200)
 7200  FORMAT(' ',T44,'       PERCENT TRUCKS HAULING COAL   ')
     IF (CH.EQ. 4) WRITE(6,7300)
 7300 FORMAT(' ',T44,'    AXLES PER TRUCK (NON-COAL-HAULING)')
     IF (CH.EQ. 5) WRITE(6,7400)
```

```
 7400 FORMAT(' ',T44,'      AXLES PER TRUCK (COAL-HAULING)  ')
      IF (CH.EQ. 6) WRITE(6,7500)
 7500 FORMAT(' ',T44,'   EAL"S PER TRUCK AXLE (NON-COAL-HAULING) ')
      IF (CH.EQ. 7) WRITE(6,7600)
 7600 FORMAT(' ',T44,'    EAL"S PER TRUCK AXLE (COAL-HAULING)   ')
      IF (CH.EQ. 8) WRITE(6,7700)
 7700 FORMAT(' ',T36,'    2-DIRECTIONAL EAL"S IN 1000"S DUE TO ',
     * '4-TIRED VEHICLES ')
      IF (CH.EQ. 9) WRITE(6,7800)
 7800 FORMAT(' ',T35,'2-DIRECTIONAL EAL"S IN 1000"S DUE TO NON-COAL',
     * '-HAULING VEHICLES')
      IF (CH.EQ.10) WRITE(6,7900)
 7900 FORMAT(' ',T35,' 2-DIRECTIONAL EAL"S IN 1000"S DUE TO COAL-',
     * 'HAULING VEHICLES  ')
      IF (CH.EQ.11) WRITE(6,8000)
 8000 FORMAT(' ',T44,'    TOTAL 2-DIRECTIONAL EAL"S IN 1000"S ')
      IF (CH.EQ.12) WRITE(6,8100)
 8100 FORMAT(' ',T44,'           STATIONS PER CATEGORY ')
C        WRITE(6,6010)
         IF (CH.EQ.1) THEN
         WRITE(6,6020)
         WRITE(6,6030)
         WRITE(6,6040)  (YRHD(YR),YR=1,YRS)
         ENDIF
      ENDIF
C
C  ****************************************************************
C  *   STATEMENTS 93 THRU 141 COMPUTE  LEAST SQUARES FITS FOR THE   *
C  *   NON-COAL-HAULING ROADS, AND PRINT THEM OUT IN CHART FORM.    *
C  ****************************************************************
C
C**** CHANGED FA,VOL,GA LOOP COUNTERS TO 1
         DO 400 FA = 1,1
            DO 300 VOL = 1,1
              DO 300 GA = 1,1
                CO   = 0
                SW   = 0.
                SWX  = 0.
                SWY  = 0.
                SWXX = 0.
                SWXY = 0.
                A    = 0.
                B    = 0.
                LO   = 20
                HI   = 1
                DO 250 YR = 1,YRS
                  WTC= CNTS(CH,FA,VOL,GA,YR) * YR
                  CYR=YR
                  RWT = 1.0+ ((CYR-1)/9)
                  RWT = 1.0
                  Y  = AADT(CH,FA,VOL,GA,YR)
                  OUT(YR) = Y
                  IF (OUT(YR).EQ.0.AND.WTC.EQ.0) OUT(YR) = 11111111
```

```
                IF (WTC.EQ.0.) GOTO 250
                  IF (YR.LT.LO) LO = YR
                  IF (YR.GT.HI) HI = YR
                  CO   = CO + 1
                  SW   = SW +RWT
                  SWX  = SWX +RWT * YR
                  SWY  = SWY +RWT * Y
                  SWXX = SWXX +RWT * YR * YR
                  SWXY = SWXY +RWT * YR * Y
       WRITE(8,42652)SW,SWX,SWY,SWXX,SWXY,RWT,YR,CO,CH
42652 FORMAT(6F10.3,I5,F10.3,I5)
 250              CONTINUE
C                IF (CO.LT.4) GOTO 290
                 IF (CO.EQ.0) GOTO 290
                  B=(SWXY-(SWX*SWY)/SW)/(SWXX-(SWX*SWX)/SW)
                  A=(SWY-B*SWX)/SW
C     IF(CH.EQ.1) THEN
C       WRITE(8,42651)SW,SWX,SWY,SWXX,SWXY,B,A
C2651 FORMAT(7F10.3)
C     ENDIF
                DO 260 YR = 1,YRS
                  OUT(YR) = A + B * YR
                  IF (OUT(YR).LT.MINVAL) OUT(YR) = MINVAL
                  IF (YR.LT.LO-2) OUT(YR) = 11111111
                  IF (YR.GT.HI+2) OUT(YR) = 11111111
 260              CONTINUE
                HI = YRS
 290            IF (OUT(HI).EQ.11111111.AND.HI.GT.1) HI = HI - 1
               IF (OUT(HI).EQ.11111111.AND.HI.GT.1) GOTO 290
C      WRITE(8,43333)B,OUT(HI)
                IF (OUT(HI).NE.0) B = B * 100. / OUT(HI)
                IF (OUT(HI).EQ.11111111) B = 0
C      WRITE(8,43333)B,OUT(HI)
43333 FORMAT(' B= ',F10.4,' OUT(HI)= ',F10.3)
     IF (OLDPRT.EQ.0) THEN
                IF (CH.EQ.1.OR.CH.GE.8)
    *            WRITE(6,6100) B,
    *                    (OUT(YRS-YR+1),YR=1,YRS)
                IF (CH.NE.1.AND.CH.LT.8)
    *            WRITE(6,6110) B,
    *                    (OUT(YRS-YR+1),YR=1,YRS)
     ENDIF
     MEAN(CH)=B
       DO 78910 IIT=1,YRS
        MEANY(CH,IIT)=OUT(YRS-IIT+1)
       IF(CH.EQ.4.AND.MEANY(CH,IIT).LT.2.0) THEN
        IF(MEANY(CH,IIT).NE.0) THEN
        MEANY(CH,IIT)=2.0
        ENDIF
       ENDIF
C       WRITE(9,72074)MEANY(CH,IIT),CH,IIT
C2074 FORMAT(F10.3,2I5)
78910 CONTINUE
```

```
C     IF (CH.EQ.1.OR.CH.GE.8) THEN
C     WRITE(9,6100)MEAN(CH),(MEANY(CH,IIT),IIT=1,YRS)
C     ELSE
C     WRITE(9,6110)MEAN(CH),(MEANY(CH,IIT),IIT=1,YRS)
C     ENDIF
 300         CONTINUE
C
C  ***************************************************************
C  *   STATEMENTS 149 THRU 203 FINDS THE AVERAGES FOR EACH OF THE   *
C  *   FEDERAL AID CATEGORIES,  COMPUTES A  LEAST SQUARES FIT AND   *
C  *   THEN PRINTS THE COMPUTED VALUES IN CHART FORM.               *
C  ***************************************************************
C
           CO   = 0
           SW   = 0.
           SWX  = 0.
           SWY  = 0.
           SWXX = 0.
           SWXY = 0.
           A    = 0.
           B    = 0.
           LO   = 20
           HI   = 1
C***** CHANGED LOOP 350 COUNTERS TO 1
           DO 350 VOL = 1,1
             DO 350 GA = 1,1
               DO 350 YR = 1,YRS
                 WTC= CNTS(CH,FA,VOL,GA,YR) * YR
                 CYR=YR
                 WT = 1+((CYR-1)/9)
                 WT = 1
                 Y  = AADT(CH,FA,VOL,GA,YR)
                 OUT(YR) = Y
                 IF (OUT(YR).EQ.0.AND.WTC.EQ.0) OUT(YR) = 11111111
                 IF (WTC.EQ.0.) GOTO 350
                   IF (YR.LT.LO) LO = YR
                   IF (YR.GT.HI) HI = YR
                   CO   = CO + 1
                   SW   = SW + WT
                   SWX  = SWX + WT * YR
                   SWY  = SWY + WT * Y
                   SWXX = SWXX + WT * YR * YR
                   SWXY = SWXY + WT * YR * Y
 350         CONTINUE
C           IF (CO.LT.4) GOTO 370
           IF (CO.EQ.0) GOTO 370
           IF ((SWXX-(SWX*SWX)/SW).EQ.0) GOTO 370
             B=(SWXY-(SWX*SWY)/SW)/(SWXX-(SWX*SWX)/SW)
             A=(SWY-B*SWX)/SW
             DO 360 YR = 1,YRS
               OUT(YR) = A + B * YR
               IF (OUT(YR).LT.MINVAL) OUT(YR) = MINVAL
               IF (YR.LT.LO-2) OUT(YR) = 11111111
```

```
              IF (YR.GT.HI+2) OUT(YR) = 11111111
 360          CONTINUE
            HI = YRS
           GOTO 390
 370        DO 380 YR = 1 , YRS
            OUT(YR) = 11111111
 380        CONTINUE
 390        IF (OUT(HI).EQ.11111111.AND.HI.GT.1) HI = HI - 1
           IF (OUT(HI).EQ.11111111.AND.HI.GT.1) GOTO 390
           IF (OUT(HI).NE.0) B = B * 100. / OUT(HI)
           IF (OUT(HI).EQ.11111111) B = 0
C           IF (CH.EQ.1.OR.CH.GE.8)
C    *       WRITE(6,6200) FA,B,(OUT(YRS-YR+1),YR=1,YRS)
C           IF (CH.NE.1.AND.CH.LT.8)
C    *       WRITE(6,6210) FA,B,(OUT(YRS-YR+1),YR=1,YRS)
 400     CONTINUE
 500     CONTINUE
C500      WRITE(6,6015)
     IF(CH.EQ.21) THEN
        WRITE(6,6020)
        WRITE(6,6030)
        WRITE(6,6050)  (YRHD(YR),YR=1,YRS)
     ENDIF
C
C  ****************************************************************
C  *   STATEMENTS 210 THRU  258  COMPUTE  LEAST  SQUARES FITS FOR   *
C  *   THE COAL-HAULING ROADS, AND PRINT THEM OUT IN CHART FORM.    *
C  ****************************************************************
C
C****** CHANGED COUNTERS TO 1
        DO 600 VOL = 1,1
          DO 600 CT = 1,1
            CO   = 0
            SW   = 0.
            SWX  = 0.
            SWY  = 0.
            SWXX = 0.
            SWXY = 0.
            A    = 0.
            B    = 0.
            LO   = 20
            HI   = 1
            DO 550 YR = 1,YRS
              WTC= CNTS(CH,6,VOL,CT,YR) * YR
                 CYR=YR
                 WT = 1+((CYR-1)/9)
                 WT = 1
              Y  = AADT(CH,6,VOL,CT,YR)
              OUT(YR) = Y
              IF (OUT(YR).EQ.0.AND.WTC.EQ.0) OUT(YR) = 11111111
              IF (WTC.EQ.0.) GOTO 550
                IF (YR.LT.LO) LO = YR
                IF (YR.GT.HI) HI = YR
```

```
                CO   = CO + 1
                SW   = SW + WT
                SWX  = SWX + WT * YR
                SWY  = SWY + WT * Y
                SWXX = SWXX + WT * YR * YR
                SWXY = SWXY + WT * YR * Y
 550          CONTINUE
C             IF (CO.LT.4) GOTO 590
              IF (CO.EQ.0) GOTO 590
              IF ((SWXX-(SWX*SWX)/SW).EQ.0) GOTO 590
                B=(SWXY-(SWX*SWY)/SW)/(SWXX-(SWX*SWX)/SW)
                A=(SWY-B*SWX)/SW
                DO 560 YR = 1,YRS
                  OUT(YR) = A + B * YR
                  IF (OUT(YR).LT.MINVAL) OUT(YR) = MINVAL
                  IF (YR.LT.LO-2) OUT(YR) = 11111111
                  IF (YR.GT.HI+2) OUT(YR) = 11111111
 560              CONTINUE
                HI = YRS
 590            IF (OUT(HI).EQ.11111111.AND.HI.GT.1) HI = HI - 1
              IF (OUT(HI).EQ.11111111.AND.HI.GT.1) GOTO 590
              IF (OUT(HI).NE.0) B = B * 100. / OUT(HI)
              IF (OUT(HI).EQ.11111111) B = 0
       IF (OLDPRT.EQ.0) THEN
              IF (CH.EQ.1.OR.CH.GE.8)
     *         WRITE(6,6400) B,
     *                  (OUT(YRS-YR+1),YR=1,YRS)
              IF (CH.NE.1.AND.CH.LT.8)
     *         WRITE(6,6410) B,
     *                  (OUT(YRS-YR+1),YR=1,YRS)
        ENDIF
       MEANC(CH)=B
         DO 78911 IIT=1,YRS
          MEANCY(CH,IIT)=OUT(YRS-IIT+1)
       IF(CH.EQ.4.AND.(MEANCY(CH,IIT).GT.0.AND.MEANCY(CH,IIT).LT.2))THEN
          MEANCY(CH,IIT)=2.0
          ENDIF
       IF(CH.EQ.5.AND.(MEANCY(CH,IIT).GT.0.AND.MEANCY(CH,IIT).LT.2))THEN
          MEANCY(CH,IIT)=2.0
          ENDIF
C         WRITE(9,72074)MEANCY(CH,IIT),CH,IIT
C2074 FORMAT(F10.3,2I5)
78911 CONTINUE
C     IF (CH.EQ.1.OR.CH.GE.8) THEN
C     WRITE(9,6400)MEANC(CH),(MEANCY(CH,IIT),IIT=1,YRS)
C     ELSE
C     WRITE(9,6410)MEANC(CH),(MEANCY(CH,IIT),IIT=1,YRS)
C     ENDIF
C              WRITE(6,26400)
 600       CONTINUE
 800  CONTINUE
C     WRITE(6,6000)
C
```

```
C  ***************************************************************
C  *   STATEMENTS 266 THRU 280 ARE FORMAT STATEMENTS.            *
C  ***************************************************************
C
 5000 FORMAT(I2,1X,3I1,F4.0,F25.3,T46,I5)
 5010 FORMAT(3X,2I1,F5.0,F25.3)
 6000 FORMAT('1',T35,A62)
 6010 FORMAT(T55,'NON-COAL-HAULING ROADS')
 6015 FORMAT(' ',T55,'  COAL-HAULING ROADS')
 6020 FORMAT('                ANNUAL ')
 6030 FORMAT('                CHANGE ')
 6040 FORMAT('                ( % )  ',20I7)
 6050 FORMAT('                ( % )  ',20I7)
C 6100 FORMAT(I7,2I4,F9.3,3X,20F7.0)
C 6110 FORMAT(I7,2I4,F9.3,3X,20F7.3)
 6100 FORMAT(T15,F9.3,3X,20F7.0)
 6110 FORMAT(T15,F9.3,3X,20F7.3)
 6200 FORMAT(I7,'       ',F9.3,3X,20F7.0)
 6210 FORMAT(I7,' AVERAGE',F9.3,3X,20F7.3)
C 6400 FORMAT(I11,I4,F9.3,3X,20F7.0)
C 6410 FORMAT(I11,I4,F9.3,3X,20F7.3)
 6400 FORMAT(T15,F9.3,3X,20F7.0)
 6410 FORMAT(T15,F9.3,3X,20F7.3)
26400 FORMAT(' ')
      CALL PRINTN (MEAN,MEANY,MEANC,MEANCY,YRS,IFUNCL,YRHD,CCNTU,CCNTC
     1,CWSTA)
       STOP
       END
      SUBROUTINE FCHEAD(IFCNUM)
      INTEGER FC
      FC=IFCNUM
C     WRITE(8,34567)IFC
C4567 FORMAT('IFC = ',I2)
      IF (FC.EQ.1) THEN
      WRITE(6,1221)
 1221 FORMAT(T35,' AGGREGATE CLASS I -- RURAL INTERSTATE')
      ELSEIF (FC.EQ.2) THEN
      WRITE(6,1222)
 1222 FORMAT(T35,' AGGREGATE CLASS II -- RURAL PRINCIPAL ARTERIAL/')
      WRITE(6,12221)
12221 FORMAT(T35,'                   RURAL MINOR ARTERIAL')
      ELSEIF (FC.EQ.3) THEN
      WRITE(6,1223)
 1223 FORMAT(T35,' AGGREGATE CLASS III -- RURAL MAJOR COLLECTOR/')
      WRITE(6,12231)
12231 FORMAT(T35,'                    RURAL MINOR COLLECTOR')
      ELSEIF (FC.EQ.4) THEN
      WRITE(6,1224)
 1224 FORMAT(T35,' AGGREGATE CLASS IV -- URBAN INTERSTATE')
      ELSEIF (FC.EQ.5) THEN
      WRITE(6,1225)
 1225 FORMAT(T35,' AGGREGATE CLASS V -- URBAN OTHER FREEWAY AND EXPRESSW
     1AYS/')
```

```
      WRITE(6,12251)
12251 FORMAT(T35,'                  URBAN OTHER PRINCIPAL ARTERIAL')
      ELSEIF (FC.EQ.6) THEN
      WRITE(6,1226)
 1226 FORMAT(T35,' AGGREGATE CLASS VI --  URBAN MINOR ARTERIAL/')
      WRITE(6,12261)
12261 FORMAT(T35,'                   URBAN COLLECTOR')
      END IF
      WRITE (6,1220)
 1220 FORMAT(T50,'AVERAGE VALUES (SMOOTHED)')
C     WRITE (6,1228)
 1228 FORMAT('  ')
      RETURN
      END
      SUBROUTINE PRINTN (MEAN,MEANY,MEANC,MEANCY,YRS,IFUNCL,YRHD,CCNTU,
     1CCNTC,CWSTA)
       REAL MEAN(11),MEANY(11,25),MEANC(11),MEANCY(11,25)
      INTEGER CCNTU(20),CCNTC(20)
      INTEGER YRS,YRHD(20),CWSTA(20)
      CHARACTER*1 QM,QMM(2)
      DATA QMM/' ','?'/
      CALL FCHEAD(IFUNCL)
25600 FORMAT(T60,'UNCLASSIFIED ROADS')
25603 FORMAT(T45,'(ALL AVC LOCATIONS AND MANUAL LOCATIONS WITH LESS')
25602 FORMAT(T46,'THAN 3% OF TRUCKS CLASSIFIED AS HEAVY/COAL)')
         WRITE(6,6021)
         WRITE(6,6030)
         WRITE(6,6040)  (YRHD(YR),YR=1,YRS)
      WRITE(6,6175)
      WRITE(6,39752) (CWSTA(M),M=YRS,9,-1)
      WRITE(6,6175)
      WRITE(6,25600)
      WRITE(6,25603)
      WRITE(6,25602)
 6020 FORMAT('                ANNUAL',T56,'UNCLASSIFIED ROADS')
 6021 FORMAT(T32,'ANNUAL')
 6030 FORMAT(T32,'CHANGE ')
 6040 FORMAT(' YEAR',T32,'( % )',T43,15I7)
      WRITE(6,6175)
C     WRITE(6,25600)
C5600 FORMAT(T56,'UNCLASSIFIED ROADS')
c     WRITE(6,7000)
c7000  FORMAT(' ',T44,'      ANNUAL AVERAGE DAILY TRAFFIC   ')
CCCC ALL LOCATIONS
      WRITE(6,31752) (CCNTU(M),M=YRS,1,-1)
      WRITE(6,6175)
31752 FORMAT(' NO. OF CLASSIFICATION STA.',T43,15I7)
39752 FORMAT(' NO. OF WEIGH STA.',T43,15I7)
      QM=QMM(1)
      IF(MEAN(1).GT. 5.0 .OR. MEAN(1) .LT. 0) QM=QMM(2)
      WRITE(6,31451)QM,MEAN(1),QM,(INT(MEANY(1,M)),M=1,YRS)
      WRITE(6,6175)
31451 FORMAT(' AADT',T30,A1,F7.3,1X,A1,T43,14I7)
```

```
21451 FORMAT(T17,F9.3,3X,20F7.0)
c     WRITE(6,25350)
c5350 FORMAT(' ',T58,'PERCENT TRUCKS')
      QM=QMM(1)
      IF(MEAN(2).GT. 5.0 .OR. MEAN(2) .LT. 0) QM=QMM(2)
      WRITE(6,31452)QM,MEAN(2),QM,(MEANY(2,M),M=1,YRS)
      WRITE(6,6175)
31452 FORMAT(' PERCENT TRUCKS',T30,A1,F7.3,1X,A1,T43,12F7.3)
21452 FORMAT(T17,F9.3,3X,20F7.3)
      QM=QMM(1)
      IF(MEAN(4).GT. 5.0 .OR. MEAN(4) .LT. 0) QM=QMM(2)
      WRITE(6,31552)QM,MEAN(4),QM,(MEANY(4,M),M=1,YRS)
      WRITE(6,6175)
31552 FORMAT(' AXLES PER TRUCK',T30,A1,F7.3,1X,A1,T43,12F7.3)
      QM=QMM(1)
      IF(MEAN(6).GT. 5.0 .OR. MEAN(6) .LT. 0) QM=QMM(2)
      WRITE(6,31652)QM,MEAN(6),QM,(MEANY(6,M),M=1,YRS)
      WRITE(6,6175)
31652 FORMAT(' EAL"S PER TRUCK AXLE ',T30,A1,F7.3,1X,A1,T43,12F7.3)
CCCC SECOND LEVEL  HEAVYWEIGHT
      WRITE(6,6175)
C     WRITE(6,25601)
C5601 FORMAT(T56,'CLASSIFIED (HEAVY/COAL) ROADS')
C     WRITE(6,25625)
C     WRITE(6,25650)
C     WRITE(6,25675) YR,YR,(NYR(N),N=1,NNYR)
      WRITE(6,25611)
      WRITE(6,25612)
      WRITE(6,25613)
25611 FORMAT(T63,'CLASSIFIED ROADS')
25612 FORMAT(T54,'(MANUAL LOCATION WITH 3% OR MORE OF ')
25613 FORMAT(T55,'TRUCKS CLASSIFIED AS HEAVY/COAL) ')
C         WRITE(6,6021)
C         WRITE(6,6030)
C         WRITE(6,6040)  (YRHD(YR),YR=1,YRS)
      WRITE(6,6175)
      WRITE(6,31752) (CCNTC(M),M=YRS,1,-1)
      WRITE(6,6175)
      WRITE(6,6175)
c6021 FORMAT(T20,'ANNUAL',T50,'CLASSIFIED (HEAVY/COAL) ROADS')
c     WRITE(6,7000)
      QM=QMM(1)
      IF(MEANC(1).GT. 5.0 .OR. MEANC(1) .LT. 0) QM=QMM(2)
      WRITE (6,31451)QM,MEANC(1),QM,(INT(MEANCY(1,L)),L=1,YRS)
      WRITE(6,6175)
      WRITE(6,6175)
c     WRITE(6,25350)
      QM=QMM(1)
      IF(MEANC(2).GT. 5.0 .OR. MEANC(2) .LT. 0) QM=QMM(2)
      WRITE (6,31452)QM,MEANC(2),QM,(MEANCY(2,L),L=1,YRS)
      WRITE(6,6175)
      WRITE(6,6175)
c     WRITE(6,25500)
```

```
c     WRITE (6,25800)MEANC(8),(MEANCY(8,L),L=1,YRS)
c     WRITE(6,25375)
c5375 FORMAT(' ',T50,'PERCENT OF TRUCKS (HEAVY/COAL)')
      QM=QMM(1)
      IF(MEANC(3).GT. 5.0 .OR. MEANC(3) .LT. 0) QM=QMM(2)
      WRITE (6,35829)QM,MEANC(3),QM,(MEANCY(3,L),L=1,YRS)
      WRITE(6,35830)
      WRITE(6,6175)
c5400 FORMAT(' ',T49,'AXLES PER TRUCK (NORMAL)')
      QM=QMM(1)
      IF(MEANC(4).GT. 5.0 .OR. MEANC(4) .LT. 0) QM=QMM(2)
      WRITE (6,35825)QM,MEANC(4),QM,(MEANCY(4,L),L=1,YRS)
      WRITE(6,35827)
      WRITE(6,6175)
c     WRITE(6,25425)
c5425 FORMAT(' ',T50,'AXLES PER TRUCK (HEAVY/COAL)')
      QM=QMM(1)
      IF(MEANC(5).GT. 5.0 .OR. MEANC(5) .LT. 0) QM=QMM(2)
      WRITE (6,35825)QM,MEANC(5),QM,(MEANCY(5,L),L=1,YRS)
      WRITE(6,35828)
      WRITE(6,6175)
c     WRITE(6,25450)
c5450 FORMAT(' ',T45,'EAL"S PER TRUCK AXLE (NORMAL)')
      QM=QMM(1)
      IF(MEANC(6).GT. 5.0 .OR. MEANC(6) .LT. 0) QM=QMM(2)
      WRITE (6,35826)QM,MEANC(6),QM,(MEANCY(6,L),L=1,YRS)
      WRITE(6,35827)
      WRITE(6,6175)
c     WRITE(6,25475)
c5475 FORMAT(' ',T47,'EAL"S PER TRUCK AXLE (HEAVY/COAL)')
      QM=QMM(1)
      IF(MEANC(7).GT. 5.0 .OR. MEANC(7) .LT. 0) QM=QMM(2)
      WRITE (6,35826)QM,MEANC(7),QM,(MEANCY(7,L),L=1,YRS)
      WRITE(6,35828)
      WRITE(6,6175)
c     WRITE(6,25525)
c5525 FORMAT(' ',T35,'2-DIRECTION EAL"S DUE TO NORMAL VEHICLES IN 1000
c    +"S')
c     WRITE (6,25800)MEANC(9),(MEANCY(9,L),L=1,YRS)
c     WRITE(6,25550)
c5550 FORMAT(' ',T38,'2-DIRECTION EAL"S DUE TO (HEAVY/COAL) VEHICLES IN
c    + 1000"S')
c     WRITE (6,25800)MEANC(10),(MEANCY(10,L),L=1,YRS)
c5800 FORMAT(T17,F9.3,3X,20F7.0)
25825 FORMAT(T17,F9.3,3X,20F7.3)
35800 FORMAT(T15,F3.0,15F7.0)
35825 FORMAT(' AXLES PER TRUCK',T30,A1,F7.3,1X,A1,T43,12F7.3)
35826 FORMAT(' EAL"S PER TRUCK AXLE',T30,A1,F7.3,1X,A1,T43,12F7.3)
35827 FORMAT(' NORMAL')
35828 FORMAT(' HEAVY/COAL')
35927 FORMAT(' NORMAL',T35,'CAUTION')
35928 FORMAT(' HEAVY/COAL',T35,'CAUTION')
35829 FORMAT(' PERCENT OF TRUCKS',T30,A1,F7.3,1X,A1,T43,12F7.3)
```

```
35830 FORMAT(' CLASSIFIED AS HEAVY/COAL')
35930 FORMAT(' CLASSIFIED AS HEAVY/COAL',T35,'CAUTION')
36650 FORMAT(T35,'CAUTION')
36651 FORMAT(' CLASSIFICATION')
36652 FORMAT(' WEIGH')
C     WRITE(6,6175)
C     WRITE(6,6175)
C     WRITE(6,6175)
C     WRITE(6,6175)
 6175 FORMAT(' ')
      RETURN
      END
```

For more information or a complete publication list, contact us at:

KENTUCKY TRANSPORTATION CENTER

176 Raymond Building University
of Kentucky Lexington, Kentucky
40506-0281

(859) 257-4513
(859) 257-1815 (FAX)
1-800-432-0719
www.ktc.uky.edu
ktc@engr.uky.edu