

**CONFIGURACIÓN DE AYUDAS PARA APLICACIÓN DE METODOLOGÍA DE
ANÁLISIS AASHTO-2002 A LAS CONDICIONES DE LA CIUDAD DE
BUCARAMANGA**

**LAURA ANDREA VARGAS CARVAJAL
ANDRÉS FELIPE GÓMEZ CHAPARRO**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS
ESCUELA DE INGENIERÍA CIVIL
BUCARAMANGA**

2011

**CONFIGURACIÓN DE AYUDAS PARA APLICACIÓN DE METODOLOGÍA DE
ANÁLISIS AASHTO-2002 A LAS CONDICIONES DE LA CIUDAD DE
BUCARAMANGA**

**LAURA ANDREA VARGAS CARVAJAL
ANDRÉS FELIPE GÓMEZ CHAPARRO**

**Trabajo realizado en la modalidad de investigación
Para optar al título de Ingeniero Civil**

**Director
Eduardo Alberto Castañeda Pinzón
Ingeniero Civil, M.Sc., Ph.D.**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS
ESCUELA DE INGENIERÍA CIVIL
BUCARAMANGA**

2011

A Dios primeramente, por la fortaleza para afrontar las dificultades.
A nuestros padres, que nos alentaron para llegar a esta meta.
A Ana María y Harvin, que nos soportaron y animaron durante este tiempo.
**A los grandes amigos que conocimos en la universidad, sin los cuales no estaríamos
finalizando este pequeño escalón de nuestras vidas.**

AGRADECIMIENTOS

Agradecemos al profesor Eduardo Catañeda, por confiar en nosotros y dirigirnos durante este periodo de culminación académica.

También agradecemos a la Subdirección de Apoyo Técnico y Dirección Territorial Santander del Instituto Nacional de Vías (INVIAS), TECNOPAVIMENTOS S.A., Direcciones e Inspecciones de Tránsito del Área Metropolitana de Bucaramanga y CORASFALTOS, fuentes principales de información para la realización de este proyecto.

TABLA DE CONTENIDO

INTRODUCCIÓN	22
OBJETIVOS	24
OBJETIVO GENERAL.....	24
OBJETIVOS ESPECÍFICOS	24
1. METODOLOGÍA AASHTO 2002	25
1.1 ANTECEDENTES	25
1.2 GUÍA DE DISEÑO AASHTO 2002	28
1.2.1 Metodología De Diseño	29
1.2.2 Niveles Jerárquicos de las Entradas de Diseño.....	32
1.2.3 Parámetros de Análisis	33
1.2.3.1 Caracterización de Materiales	33
1.2.3.2 Caracterización del Tráfico.....	39
1.2.3.3 Caracterización del Clima.....	42
2. RECOPIACIÓN, PROCESAMIENTO Y ADAPTACIÓN DE LOS REQUERIMIENTOS DE LA GUIA AASHTO 2002 PARA EL TRÁFICO	45
2.1 FACTOR DE DISTRIBUCIÓN DIRECCIONAL.....	45
2.2 FACTOR DE DISTRIBUCIÓN DE CARRIL	45
2.3 VELOCIDAD DE OPERACIÓN	46
2.4 TIPOLOGÍA DE VEHÍCULOS DE CARGA.....	48
2.5 DIMENSIONES DE LOS VEHÍCULOS DE CARGA	50
2.6 PESO BRUTO VEHICULAR DE VEHÍCULOS DE CARGA Y PESO MÁXIMO POR EJE	50

2.7 REGLAMENTACIÓN DEL TRÁNSITO DE TRANSPORTE DE CARGA EN LA CIUDAD DE BUCARAMANGA	53
2.8 VOLÚMENES DE TRÁNSITO Y COMPOSICIÓN VEHICULAR	58
2.8.1 Estaciones de Conteo-Vías de Santander	58
2.8.2 Tránsito Promedio Diario Anual (TPDA)	63
2.8.3 Tránsito Promedio Diario Anual de Tráfico Pesado (TPDAT)	64
2.8.4 Distribución Vehicular	64
2.8.5 Variación horaria.....	70
2.9 ESPECTRO DE CARGA	75
2.10 ENTRADAS GENERALES DE TRÁFICO	80
2.10.1 Ubicación de la rueda	80
2.10.2 Desviación de ubicación de rueda	80
2.10.3 Ancho de calzada	81
2.10.4 Número de ejes por clase de camión.....	82
2.10.5 Configuración del eje	82
2.10.5.1 Ancho del eje.....	82
2.10.5.2 Distancia mínima de gemelado	83
2.10.5.3 Espaciamiento entre ejes	84
2.10.5.4 Presión de inflado.....	85
2.10.6 Batalla.....	85
3. RECOPIACIÓN, PROCESAMIENTO Y ADAPTACIÓN DE LOS REQUERIMIENTOS DE LA GUIA AASHTO 2002 PARA LOS MATERIALES	86
3.1 CAPA ASFÁLTICA	86
3.1.1 MEZCLA ASFÁLTICA EN CALIENTE	86
3.1.1.1 Agregados pétreos y llenante mineral	86

3.1.1.2 Material Bituminoso.....	87
3.1.1.3 Diseño de la mezcla	89
3.2 PAVIMENTO DE CONCRETO HIDRÁULICO	90
3.2.1 Cemento	90
3.2.2 Agregado fino.....	92
3.2.3 Agregado grueso	93
3.2.4 Métodos de Curado	94
3.2.5 Módulo de rotura.....	94
3.2.6 Módulo de Elasticidad.....	95
3.3 CLASIFICACIÓN DE SUELOS	95
3.3.1 Sistema de clasificación de suelos según la AASHTO	96
3.3.2 Sistema unificado de clasificación de suelos	98
3.4 BASE GRANULAR	102
3.5 SUBBASE GRANULAR.....	104
3.6 SUBRASANTE	106
4. MANUAL DEL SOFTWARE SOPORTE PARA EL DESARROLLO DE LA METODOLOGÍA AASHTO 2002.....	109
4.1 CÓMO USAR ESTE MANUAL	109
4.2 INSTALACIÓN DEL SOFTWARE	109
4.3 INTRODUCCIÓN.....	113
4.4 INFORMACIÓN GENERAL.....	114
4.5 TRÁNSITO	115
4.5.1 Información general	116
4.5.2 Entradas generales.....	117

4.5.3 Volúmenes de tránsito	117
4.5.4 Espectro de carga.....	119
4.5.5 Factores de Crecimiento.....	120
4.5.6 Resumen de Tránsito.....	120
4.6 ESTRUCTURA.....	123
4.6.1 Capa asfáltica	123
4.6.1.1 Mezcla asfáltica.....	124
4.6.1.2 Ligante	125
4.6.2 Losa de concreto	126
4.6.3 Base,Subbase y Subrasante	126
5. CONCLUSIONES Y RECOMENDACIONES.....	129
BIBLIOGRAFÍA.....	131
ANEXOS	133

LISTA DE TABLAS

Tabla 1.	Factor de distribución direccional	45
Tabla 2.	Porcentaje de vehículos para el carril de diseño	46
Tabla 3.	Velocidad de diseño de los tramos homogéneos en función de la categoría de la carretera y el tipo de terreno	47
Tabla 4.	Velocidades de operación promedio para diferentes condiciones de tráfico	47
Tabla 5.	Tipología para vehículos de transporte de carga	49
Tabla 6.	Dimensiones máximas de los vehículos de carga.....	50
Tabla 7.	Peso máximo por eje.....	51
Tabla 8.	Peso máximo bruto vehicular	52
Tabla 9.	Estaciones de conteo con TPDs de 0-500	59
Tabla 10.	Estaciones de conteo con TPDs de 501-1000	60
Tabla 11.	Estaciones de conteo con TPDs de 1001-2500	60
Tabla 12.	Estaciones de conteo con TPDs de 2501-5000	60
Tabla 13.	Estaciones de conteo con TPDs de 5001-10000	61
Tabla 14.	Estaciones de conteo con TPDs>10000	62
Tabla 15.	TPDA, TPDAT y distribución vehicular para estaciones con TPDS de 0-500	66
Tabla 16.	TPDA, TPDAT y distribución vehicular para estaciones con TPDS de 501-1000	66
Tabla 17.	TPDA, TPDAT y distribución vehicular para estaciones con TPDS de 1001-2500	67
Tabla 18.	TPDA, TPDAT y distribución vehicular para estaciones con TPDS de 2501-5000	67

Tabla 19.	TPDA, TPDAT y distribución vehicular para estaciones con TPDS de 5001-10000	69
Tabla 20.	TPDA, TPDAT y distribución vehicular para estaciones con TPDS mayor a 10000	70
Tabla 21.	Variación horaria para estaciones con TPDS de 0-500	71
Tabla 22.	Variación horaria para estaciones con TPDS de 501-1000	71
Tabla 23.	Variación horaria para estaciones con TPDS de 1001-2500	72
Tabla 24.	Variación horaria para estaciones con TPDS de 2501-5000	72
Tabla 25.	Variación horaria para estaciones con TPDS de 5001-10000	73
Tabla 26.	Variación horaria para estaciones con TPDS de 5001-10000	74
Tabla 27.	Número de ejes sencillos, tandem y tridem por cada mil camiones para vías con diferentes TPDs	75
Tabla 28.	Distribución porcentual para el eje sencillo	76
Tabla 29.	Distribución porcentual para el eje tandem	77
Tabla 30.	Distribución porcentual par el eje tridem	78
Tabla 31.	Ubicación de la rueda en función del ancho de la calzada	80
Tabla 32.	Ancho de calzada (m)	81
Tabla 33.	Número de ejes por clase de camión	82
Tabla 34.	Ancho promedio del eje (m)	83
Tabla 35.	Distancia mínima de gemelado (mm)	84
Tabla 36.	Franjas granulométricas para mezclas asfálticas en caliente	86
Tabla 37.	Especificaciones del cemento asfáltico	87
Tabla 38.	Especificaciones de cementos asfálticos modificados con polímeros	88
Tabla 39.	Criterios de diseño de la mezcla asfáltica en caliente por el método Marshall	89

Tabla 40.	Requisitos físicos del cemento Portland.....	91
Tabla 41.	Granulometría para el agregado fino para pavimentos de concreto hidráulico	93
Tabla 42.	Granulometrías para el agregado grueso para pavimentos de concreto hidráulico	93
Tabla 43.	Resistencia a flexión del concreto	94
Tabla 44.	Correlación entre la resistencia a la compresión y el módulo de elasticidad	95
Tabla 45.	Límites de tamaño de partículas	95
Tabla 46.	Sistema de clasificación de suelos según la AASHTO.....	97
Tabla 47.	Símbolos usados en el sistema Unificado de clasificación de suelos	98
Tabla 48.	Clasificación unificada de suelos.....	100
Tabla 49.	Requisitos de los agregados para bases granulares.....	102
Tabla 50.	Granulometría para base granular	103
Tabla 51.	Requisitos de los agregados para subbases granulares	104
Tabla 52.	Granulometría para subbase granular.....	105
Tabla 53.	Categorías de subrasante	108

LISTA DE FIGURAS

Figura 1.	Características de las secciones de prueba en los circuitos	26
Figura 2.	Diseño Mecanístico-Empírico.....	29
Figura 3.	Procedimiento de análisis ME-PDG	31
Figura 4.	Restricciones de carga en el Área Metropolitana de Bucaramanga...	57
Figura 5.	Estaciones de conteo INVIAS-Santander.....	58
Figura 6.	Ubicación de la rueda.....	80
Figura 7.	Ancho del eje.....	82
Figura 8.	Distancia de gemelado.....	83
Figura 9.	Espaciamiento entre ejes	84
Figura 10.	Presión de inflado [kPa]	85
Figura 11.	Batalla.....	85
Figura 12.	Carta de plasticidad de Casagrande	99
Figura 13.	Correlación AASHTO. Clasificación del suelo vs CBR y MR	107
Figura 14.	Microsoft.NET Framework 4 ClientProfile	109
Figura 15.	Asistente de instalación-Paso1	110
Figura 16.	Asistente de instalación-Paso 2.....	111
Figura 17.	Asistente de instalación-Paso 3.....	111
Figura 18.	Asistente de instalación-Paso 4.....	112
Figura 19.	Introducción	113
Figura 20.	Información general del proyecto.....	114
Figura 21.	Tránsito.....	115
Figura 22.	Información general	116
Figura 23.	Entradas generales de tránsito	117

Figura 24.	Volúmenes de tránsito	118
Figura 25.	Espectro de carga	119
Figura 26.	Factores de crecimiento.....	120
Figura 27.	Resumen de tránsito	121
Figura 28.	Resumen de tránsito – Excel	121
Figura 29.	Resumen de tránsito-Proyección	122
Figura 30.	Capas de la estructura de pavimento.....	123
Figura 31.	Propiedades de la mezcla asfáltica.....	124
Figura 32.	Propiedades del ligante.....	125
Figura 33.	Propiedades de la losa de concreto	126
Figura 34.	Propiedades de la base, subbase y subrasante	127
Figura 35.	Ensayo triaxial.....	128

LISTA DE ANEXOS

Anexo A.	Serie histórica y composición del tránsito promedio diario semanal-Santander	133
Anexo B.	Espectro de carga	135
Anexo B.1	Espectro de carga-eje simple	135
Anexo B.2	Espectro de carga-eje tandem.....	136
Anexo B.3	Espectro de carga-eje tandem.....	137
Anexo C.	Código fuente	139
Anexo C.1	Información general	139
Anexo C.1.1	Formularios	139
Anexo C.1.2	Clases	147
Anexo C.1.3	Módulos	150
Anexo C.2	Información de tráfico	172
Anexo C.2.1	Formularios	172
Anexo C.2.2	Clases	239
Anexo C.3	Información estructural del pavimento	246
Anexo C.3.1	Formularios	246
Anexo C.3.2	Clases	261

GLOSARIO

PESO BRUTO VEHICULAR (PBV): peso de un vehículo en condiciones de marcha más el máximo de carga que puede transportar.

EICM: Enhanced Integrated climate Model (modelo climático integrado reforzado)

ESAL: Equivalent single axle load (eje equivalentes sencillo de carga)

HMA: Hot mix asphalt (mezcla asfáltica en caliente)

JPCP: Jointed plain concrete pavement (Pavimento rígido)

LEF: Load equivalent factor (factor de equivalencia de carga)

LTPP: Long term pavement performance (funcionamiento de pavimento a largo plazo)

PCC: Portland cement concrete

RESUMEN

TITULO: CONFIGURACIÓN DE AYUDAS PARA APLICACIÓN DE METODOLOGÍA DE ANÁLISIS AASHTO-2002 A LAS CONDICIONES DE LA CIUDAD DE BUCARAMANGA*

AUTORES: LAURA ANDREA VARGAS CARVAJAL**
ANDRÉS FELIPE GÓMEZ CHAPARRO**

PALABRAS CLAVES: AASHTO 2002, Mecánico-Empírico, Tránsito, Materiales.

CONTENIDO:

La metodología AASHTO 2002 desarrollada en respuesta a las limitaciones que presentaban las metodologías basadas en métodos empíricos, relaciona principios mecánicos y empíricos en aras de reducir la incertidumbre en el análisis de un pavimento y tiene como fundamento principal el comportamiento de la estructura a partir de criterios de funcionamiento, considerando factores locales tales como las condiciones del tráfico, los efectos climáticos, las características del suelo de fundación o subrasante, así como de los materiales que componen las capas de la estructura en tres niveles jerárquicos que dependen de la fuente de la información y su confiabilidad. La información de primer nivel se obtiene directamente del tramo a evaluar y la de tercer nivel tomando valores por defecto para el análisis.

Esta investigación tiene como propósito adaptar los parámetros de entrada referentes al tráfico y materiales para el desarrollo de la metodología AASHTO 2002 a las condiciones de las vías colombianas, a partir de una interfaz gráfica que compila toda la información requerida para caracterizar el tránsito, tal como la distribución horaria, distribución vehicular, espectros de carga entre otros, y aquella referente a las propiedades de las diferentes capas que componen la estructura de pavimento como por ejemplo el tipo de material, granulometría y límites de Atterberg, en los tres niveles jerárquicos, el primero basado en ensayos y conteos realizados en diferentes vías del departamento de Santander, el segundo a partir de correlaciones y el tercero basado en normativa y series históricas.

* Proyecto de Grado

** Facultad de Ingenierías Físico mecánicas. Escuela de Ingeniería Civil. Director: Ing. Eduardo Alberto Castañeda Pinzón.

ABSTRACT

TITLE: SUPPORTING MATERIAL FOR THE APPLICATION OF AASHTO 2002 ANALYSIS METHODOLOGY TO THE CONDITIONS OF THE CITY OF BUCARAMANGA*

AUTHORS: LAURA ANDREA VARGAS CARVAJAL**
ANDRÉS FELIPE GÓMEZ CHAPARRO**

KEYWORDS: AASHTO 2002, Mechanistic-Empirical, Traffic, Materials.

CONTENT:

AASHTO 2002 methodology was developed to answer at the limitations that methodologies based on empirical methods had, it relates mechanistic and empirical principles in order to reduce uncertainty in pavement analysis and has as main basis the behavior of the structure from performance criteria, taking into account local factors such as traffic conditions, climatic effects, foundation or subgrade characteristics as well as characteristics of the materials of the different layers of the pavement structure in three hierarchical levels based on the source and the accuracy of information. The first level is obtained directly from the road that is going to be analyzed and the third level uses default values for the analysis.

The purpose of this Research is to adapt the inputs related to traffic and materials that are needed to develop AASHTO 2002 methodology to the Colombian roads conditions, with a graphical user interface that compiles all the required information for traffic characterization such as hourly distribution, vehicle class distribution, axle load distribution and others, and information about the properties of the different layers of the pavement structure such as type of material, particle size analysis and Atterberg limits, in the three hierarchical levels, the first of them based on test and counts done in different roads of the department, the second level through correlations and the third level based on regulations and time series.

* Undergraduate Research Project

** Physical Mechanical Engineering Faculty. Civil Engineering School. Director: Ing. Eduardo Alberto Castañeda Pinzón.

INTRODUCCIÓN

El diseño de estructuras de pavimento ha evolucionado a través del tiempo, y en especial en las últimas décadas, pasando de tener una serie de gráficas y correlaciones para el diseño empírico de las capas, a contar con herramientas de análisis que tienen en cuenta las condiciones particulares de cada proyecto, haciendo uso de métodos de elementos finitos y teorías de capa elástica para análisis mecánicos más detallados.

Hoy en día aún existe un grado de incertidumbre considerable cuando se trata de predecir el comportamiento de un pavimento a través del tiempo. Por lo tanto, es necesario utilizar todas las herramientas disponibles, como los métodos mecánicos que son únicamente para análisis de diseños realizados generalmente con métodos empíricos, como el método AASHTO-93, para incrementar el grado de certeza en la predicción de la vida útil del pavimento. La American Association of State Highway and Transportation Officials, mejor conocida como AASHTO, desarrolló un software en el año 2002 para predecir la evolución de los tipos más relevantes de fallas en el periodo de diseño, con base en datos de tráfico, clima y propiedades de los materiales de la estructura variables a través del tiempo, para posteriormente comparar el comportamiento del daño de la estructura con límites de servicio que establece el diseñador, bien sea para una vía nueva o para reparar una vía existente.

El software de análisis AASHTO-2002 tiene parámetros por defecto que, según la guía de diseño y el manual del programa, pueden ser utilizados para realizar análisis sin ningún problema, pues son producto de estudios del comportamiento de pavimento a largo plazo, realizados previamente, junto con análisis de tráfico en las condiciones estadounidenses, datos de estaciones hidroclimáticas en ese país, y estudios de propiedades de algunos materiales granulares utilizados en la construcción de vías en Norteamérica. Todos estos parámetros difieren en gran

manera con las condiciones de tráfico pesado en Colombia, clima y propiedades de algunos materiales utilizados en la construcción o restauración de vías. Por esta razón se hace necesario determinar los parámetros de tráfico, clima y propiedades que puedan ser utilizados en análisis mecánicos del diseño de pavimentos para diferentes tipos de vía, ajustados a condiciones locales, además de definir las características técnicas de distancia entre ejes, distancia entre llantas de ejes de rueda doble, número de ejes por clase de vehículo, entre otros, de los vehículos de tráfico pesado que se encuentran en las carreteras colombianas, y particularmente en la región santandereana.

La determinación de los parámetros de análisis descritos anteriormente se realizará con base en datos históricos de tráfico y su distribución, pesajes y características técnicas de los vehículos de tráfico pesado, así como series históricas del parque automotor, para realizar una predicción ajustada a las condiciones locales para futuros diseños. Además, todos los casos analizados se organizarán de manera que los diseñadores de pavimentos tengan una base de datos, que podrán utilizar con certeza de su veracidad y fidelidad, a través de una interfaz gráfica amigable y completa, también ajustada a factores incidentes en Colombia, teniendo en cuenta las clases de vehículos existentes en las vías nacionales y los rangos de carga por eje del tráfico pesado permitidas y reales.

OBJETIVOS

OBJETIVO GENERAL

- Desarrollar un sistema de ayudas para la aplicación de la metodología de análisis AASHTO 2002 a las condiciones de vías nacionales.

OBJETIVOS ESPECÍFICOS

- Recopilar información referente al tráfico existente en la malla vial de Colombia y propiedades de los materiales utilizados en la construcción del pavimento en la ciudad.
- Procesar la información recopilada de acuerdo a los parámetros necesarios para la aplicación de la metodología AASHTO 2002.
- Adaptar dicha información a los parámetros requeridos para la aplicación de la metodología de análisis AASHTO 2002, de acuerdo a las condiciones de tráfico y propiedades de la estructura de pavimento de la región.
- Crear una interfaz gráfica de usuario que realice la predicción del tráfico futuro para los diferentes ejemplos de vías analizados de la ciudad de Bucaramanga, utilizando factores de crecimiento determinados también a partir de series históricas locales.

1. METODOLOGÍA AASHTO 2002

1.1 ANTECEDENTES

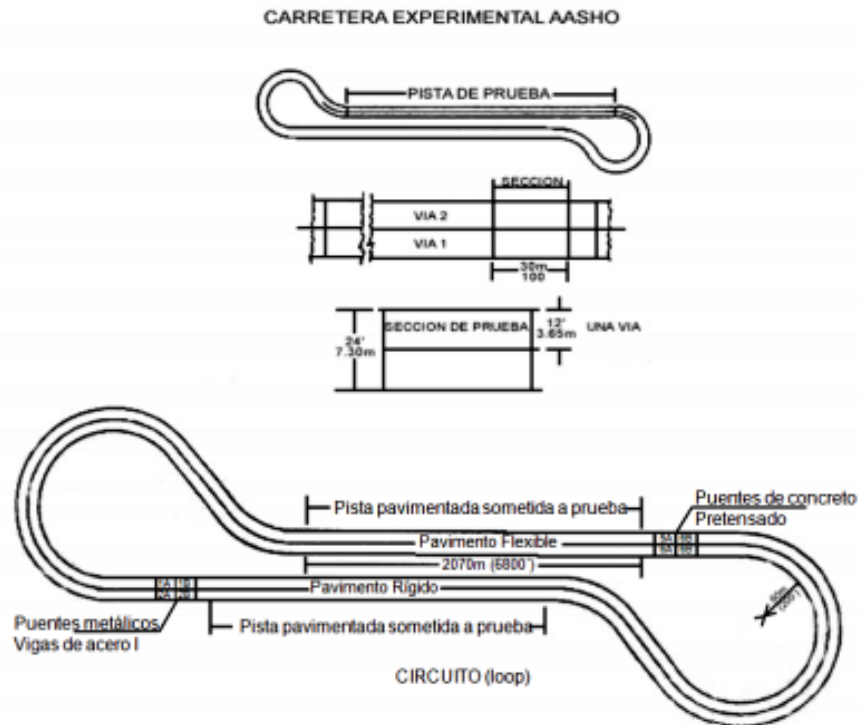
El experimento de carreteras AASHO (American Association of State Highway Officials) realizado durante el periodo de 1958 a 1960 cerca de Ottawa, Illinois, fue la primera fuente de información de datos experimentales relativos al daño que producen los vehículos en las carreteras.

El objetivo principal de este experimento consistía en estudiar el rendimiento de la estructura de pavimento de espesor conocido bajo cargas móviles de magnitud y frecuencia conocida y bajo el efecto del medio ambiente, el cual se llevó a cabo a partir de 6 circuitos de prueba de dos carriles que tenían la mitad del tramo en pavimento de concreto y la otra en pavimento flexible, las secciones de prueba de los pavimentos estaban localizadas en la rectas tangentes de cada circuito, con una longitud aproximada de 2.070 m, subdivididas en sectores de 30 m de largo. La Figura 1 muestra la planta típica de uno de estos circuitos. El Circuito 1 se destinó a medir el efecto del clima y algunas cargas estáticas en los pavimentos, el Circuito 2 se sometió a cargas livianas, y los Circuitos 3, 4, 5 y 6 se sometieron a diversas cargas pesadas.¹

La información de campo obtenida de este experimento fue la base para desarrollar la guía de diseño de pavimentos, cuya primera edición, basada en modelos empíricos, salió en 1961 como “Guía AASHO para Diseño de Pavimentos Rígido y Flexibles”. Posteriormente, en el año 1972, se incorporó el “tránsito” como responsabilidad de la organización, por lo cual se comienza a conocer como AASHTO (*American Association of State Highway and Transportation Officials*), además se realizó una revisión y se publicó el “Procedimiento Provisional para el

¹ CORREDOR, Gustavo. Experimento Vial de la AASHO y las Guías de Diseño AASHTO. Universidad Nacional de Ingeniería. Facultad de Tecnología de la Construcción. 4 p.

Figura 1. Características de las secciones de prueba en los circuitos



Fuente: Experimento Vial de la AASHO y las Guías de Diseño AASHTO

Diseño de Pavimentos AASHTO-72". Más adelante, en el año 1986 se publicó una revisión de la guía bajo el nombre de "Manual de Diseño de Estructuras de Pavimentos AASHTO '86", la cual cambió sustancialmente la metodología al utilizar el procedimiento de teoría elástica multicapa; y en 1993 se realizó una nueva revisión la cual hoy en día se conoce como Método AASHTO-93.

Esta nueva actualización, basada en parámetros tales como: capacidad de servicio, cargas equivalentes sobre ejes sencillos de 18.000 lbs, usando factores de equivalencia y convirtiendo diferentes tipos de ejes a ejes equivalentes (ESALs) y módulos resilientes para la determinación de los coeficientes estructurales, permitió la posibilidad de estimar el comportamiento de los pavimentos con un

enfoque más racional de conformidad con los principios fundamentales de la teoría elástica, permitiendo tomar en consideración los efectos de las características ambientales tales como humedad y temperatura, el efecto de la combinación de la carga en los bordes y un posible cambio en el tráfico a lo largo del periodo de diseño con la incorporación de un factor de confiabilidad, entre otras.

Sin embargo, a pesar de la simplicidad y facilidad en el uso de esta metodología de diseño AASHTO, tiene una serie de limitaciones en su formulación, debido a que esta se realizó con base en información empírica muy limitada, en una sola localidad y con niveles de tráfico menores si son comparados con los que existen hoy en día, estas razones afectan la validez de las relaciones cuando son utilizadas en condiciones diferentes a las consideradas en el desarrollo del modelo, es decir cuando las características del terreno, materiales utilizados para la estructura y ambiente de exposición difiere de las estudiadas con motivo de la prueba empírica AASHTO Road Test.

Las teorías mecanísticas caracterizan de manera más realista los pavimentos en servicio y mejoran la confiabilidad de los diseños, teniendo en cuenta el estado de esfuerzos y deformaciones de la estructura de pavimento a causa de las cargas impuestas de tráfico, el estado de los materiales que conforman las capas de la estructura, y en consecuencia la respuesta de esta a variaciones en la temperatura y a condiciones ambientales a las que esté expuesta la estructura. Sin embargo, debido a los vacíos que aún existen en el conocimiento, los métodos de diseño mecanísticos tienen que apoyarse en relaciones empíricas.

Por esta razón, se desarrolló una nueva Guía para el Diseño de Estructuras de Pavimentos Nuevos y Rehabilitadas, con métodos basados en principios Mecanísticos-Empíricos (M-E), al igual que el software necesario para su aplicación, que finalmente fue adoptado y distribuido por la AASHTO.

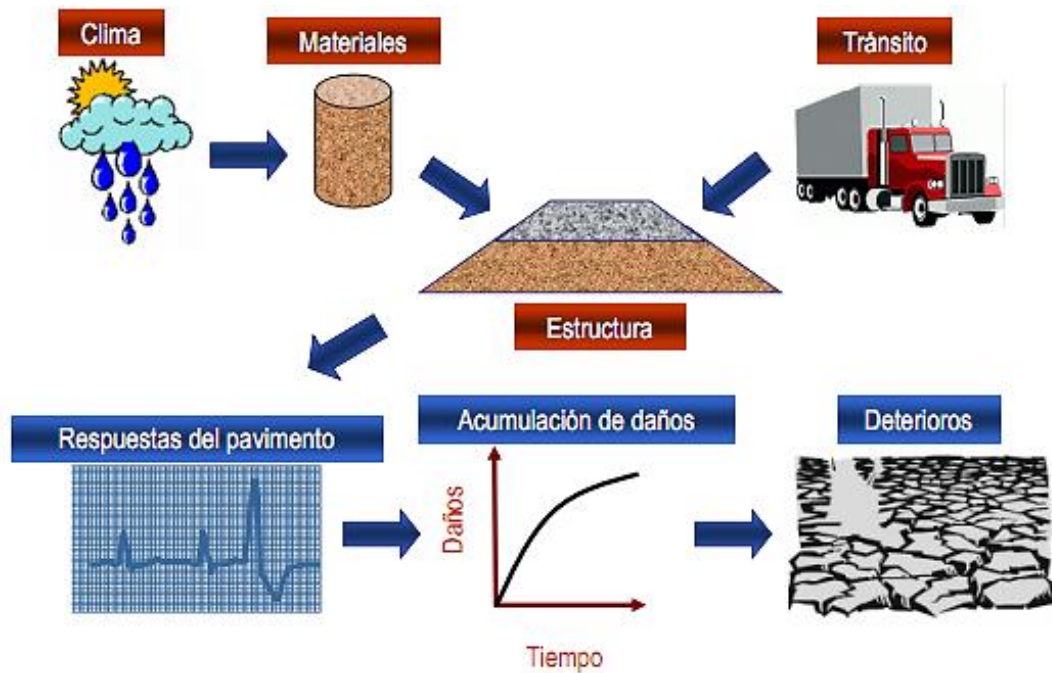
En el año 2002, se publicitó el lanzamiento de la metodología de diseño AASHTO del mismo año, que posteriormente sería parte de un proyecto mayor financiado por la AASHTO y con cooperación de la FHWA, cambiando su nombre en el año 2004 para convertirse en el proyecto NCHRP 1-37A. El principal objetivo de esta metodología es proporcionar una herramienta para el diseño de estructuras de pavimentos nuevos y rehabilitados utilizando principios teóricos-empíricos.

1.2 GUÍA DE DISEÑO AASHTO 2002

La Guía de Diseño de Pavimentos Mecánica-Empírica , M-E PDG por sus siglas en inglés (*Mechanistic-Empirical Pavement Design Guide*), desarrollada a través de un programa de investigación de la *National Cooperative Highway* (NCHRP Project 1-37^a) está basada en el funcionamiento de la estructura de pavimento, considerando en primera instancia condiciones locales tales como tráfico, clima, características del suelo de subrasante , condiciones del pavimento existente para rehabilitación y condiciones de construcción de la estructura, como características de los materiales de las capas que conforman la estructura de pavimento; evaluando los parámetros mencionados anteriormente y a partir de criterios de funcionamiento, el procedimiento de diseño predice y revisa el desempeño del pavimento, teniendo en cuenta la severidad de diferentes tipos de falla y la regularidad de la superficie de rodadura.

Esta guía de diseño, así como la documentación relacionada y el software desarrollado en el proyecto NCHRP 1-37A, están disponibles en línea en <http://www.trb.org/mepdg/>.

Figura 2. Diseño Mecanístico-Empírico



Fuente: Diseño de Pavimentos ME-PDG.

1.2.1 Metodología De Diseño

El diseño se lleva a cabo en tres etapas principales enunciadas a continuación: La etapa 1 consiste en el desarrollo y entrada de los datos requeridos para la evaluación de la estructura, entre estos se encuentran datos concernientes para la consideración de los efectos climáticos considerados a partir de perfiles de temperatura y humedad en toda la profundidad de la estructura de pavimento, para así estimar su incidencia en las propiedades de los materiales de las capas durante la vida de diseño del pavimento; características de los materiales que componen las capas de la estructura tales como, valores de resistencia, módulos dinámicos y propiedades de los agregados; valores de tráfico los cuales se consideran a partir de distribuciones de carga por tipo de eje (eje sencillo, tándem, tridem y cuádruple) en lugar de ESALs (parámetro utilizado en anteriores

versiones del método AASHTO), clasificación por tipo de vehículo, distribución mensual y horaria, entre otros.

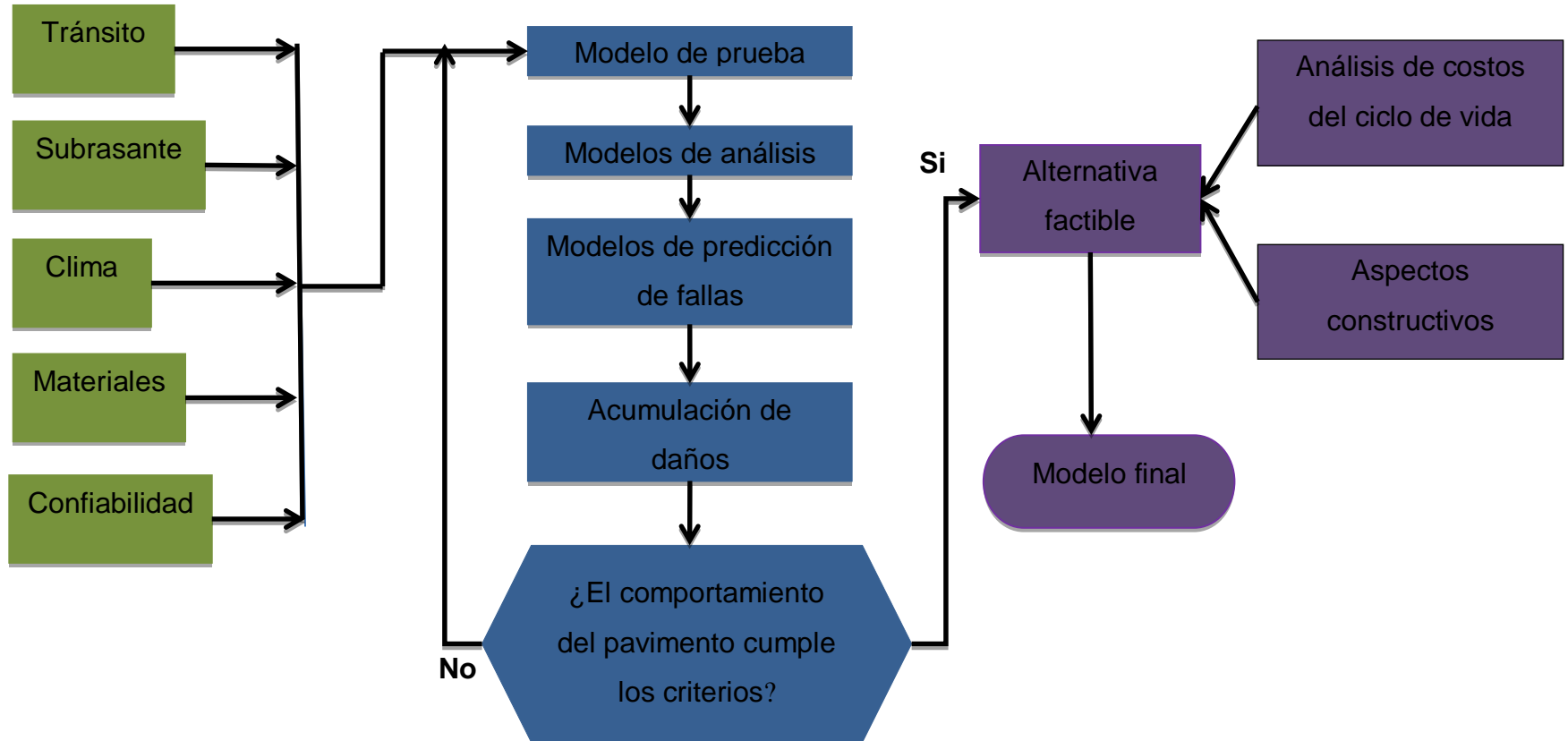
Un paso importante en este proceso es el análisis de la fundación o subrasante, el cual consiste en determinar la rigidez del suelo de fundación, así como los esfuerzos y fallas generadas por la imposición de las cargas de tráfico y su incidencia en el comportamiento de la estructura.

La etapa 2 del procedimiento de diseño consiste en un análisis de desempeño de la estructura de pavimento, el cual es un proceso iterativo que inicia con la selección de un diseño de prueba. El diseño de prueba requiere estimaciones iniciales de espesor de las capas de la estructura del pavimento, propiedades geométricas, regularidad superficial inicial, características de los materiales del pavimento, entre otros. La sección de prueba es analizada incrementalmente en el tiempo a partir de la respuesta de la estructura y los modelos de falla, obteniendo como resultado el daño acumulado, la cantidad de fallas esperada y suavidad a través del tiempo. Si el diseño de prueba no coincide con los criterios de funcionamiento previamente establecidos se realizan modificaciones y se efectúa una vez más el análisis hasta que el resultado obtenido sea satisfactorio.

Por último, la etapa 3 del procedimiento incluye aquellas actividades requeridas para evaluar alternativas estructuralmente viables, tales como el análisis ingenieril y análisis de costos del ciclo de vida de la estructura.²

²NATIONAL COOPERATIVE HIGHWAY RESEARCH PROGRAM. Background, Scope and Overview. En: Guide for Mechanistic-Empirical Design of New and Rehabilitated Pavement Structures. Part 1: Introduction. Illinois, 2004.p. 1.1.11.

Figura 3. Procedimiento de análisis ME-PDG



Fuente: Guide for Mechanistic-Empirical Design of New and Rehabilitated Pavement Structures

1.2.2 Niveles Jerárquicos de las Entradas de Diseño

De acuerdo a la disponibilidad de la información requerida en el análisis y a la confiabilidad que se pretenda obtener, la guía dispone de tres niveles jerárquicos de los datos de entrada. Estos son:

- **Nivel 1:** Los valores de entrada de este tipo proveen el mayor nivel de precisión, por lo cual tendrán el menor margen de error o incertidumbre. Requieren ensayos de campo y laboratorio tales como: ensayos de módulo dinámico de la mezcla asfáltica, datos del espectro de carga por tipo de eje en el sitio específico, ensayo de deflexión no destructivo, entre otros; y además requieren más recursos y tiempo que otros niveles jerárquicos.
- **Nivel 2:** Las entradas de nivel 2 proveen un nivel medio de precisión y pueden ser utilizadas cuando los recursos o el equipo para los ensayos requeridos para los datos de Nivel 1 no están disponibles. Los valores de este nivel jerárquico pueden ser estimados mediante correlaciones o derivados de una base de datos limitada de ensayos, ejemplos de estas entradas incluyen el módulo resiliente de la subrasante o materiales de base estimados del CBR.
- **Nivel 3:** Las entradas de este tipo proveen el menor nivel de precisión y provienen de promedios regionales o nacionales. Se utilizan cuando hay poco conocimiento en las características del pavimento tanto pasadas como futuras.³

³Ibíd.p. 1.1.15.

1.2.3 Parámetros de Análisis

El análisis del funcionamiento del pavimento se basa en la consideración del rendimiento y comportamiento de la estructura frente a las cargas impuestas, propiedades de sus materiales y factores externos como el clima. La guía AASHTO 2002 evalúa el funcionamiento de la estructura, el cual está relacionado con su condición física, a partir de parámetros específicos de acuerdo al tipo de pavimento que se esté analizando. Para pavimentos flexibles se toman en cuenta condiciones de falla tales como: agrietamiento por fatiga y deformaciones, y para pavimentos rígido fallas en juntas y grietas en las losas.

Además, el parámetro que indica el comportamiento funcional del pavimento es la regularidad superficial de este, medida a partir del IRI (International Roughness Index). El enfoque que utiliza la guía es considerar como parte del criterio de funcionamiento del pavimento su suavidad inicial al tiempo de ser construido (IRI inicial) y predecir cambios en este parámetro a través del tiempo como una función de los esfuerzos, fallas y condiciones del pavimento (IRI final).

1.2.3.1 Caracterización de Materiales⁴

Para el análisis mecanístico-empírico de la estructura de pavimento se requieren ciertas propiedades de los materiales que la componen para evaluar su respuesta a partir de la predicción de estados de esfuerzos y deformaciones sujetas a la aplicación de una carga externa de tráfico; determinar fallas o parámetros de suavidad los cuales se ven influenciados por propiedades específicas de las capas del pavimento y como complemento al modelo climático para así considerar factores internos que influyan en el funcionamiento de la estructura a lo largo de

⁴NATIONAL COOPERATIVE HIGHWAY RESEARCH PROGRAM. Background, Scope and Overview. En: Guide for Mechanistic-Empirical Design of New and Rehabilitated Pavement Structures. Part 2: Design Inputs. Chapter 2: Material Characterization. Illinois, 2004.

su vida de diseño a partir de perfiles de temperatura y humedad a través de la sección transversal de la estructura.

La guía de diseño AASHTO 2002 trabaja con las siguientes propiedades de materiales de acuerdo a la capa de la estructura que compongan:

➤ **Caracterización de mezclas asfálticas en caliente**

- **Variación del módulo dinámico con respecto al tiempo y temperatura:** las propiedades del módulo del asfalto sufren variaciones por temperatura, frecuencia de carga, tiempo, y por las características de la mezcla tales como la rigidez del ligante, granulometría, contenido de ligante y porcentaje de vacíos. Esta relación puede ser determinada a partir de una curva maestra construida a una temperatura de referencia en donde datos de varias temperaturas son desplazados con respecto al tiempo hasta que estas curvas converjan en una única función.
- **Viscosidad del ligante:** la viscosidad del ligante varía con la temperatura lo cual se denomina susceptibilidad térmica, además, la penetración también está ligada a la viscosidad del material.
- **Coefficiente de Poisson:** el coeficiente de Poisson es un parámetro que indica las deformaciones relativas en sentido transversal que sufre el material y las deformaciones relativas en dirección de la fuerza aplicada sobre el mismo, para materiales bituminosos usualmente varía entre 0.15 y 0.50 y es una función de la temperatura.
- **Resistencia a tensión:** es una medida que permite conocer las características del material cuando se somete a esfuerzos de tracción a partir de curvas esfuerzo-deformación.

- ***Esfuerzo de fluencia:*** este parámetro indica el punto a partir del cual el material se deforma plásticamente.
 - ***Coefficiente de expansión térmica:*** esta medida refleja la magnitud a la cual el material se expande bajo calentamiento.
 - ***Coefficiente de absorción de onda corta:*** el coeficiente de absorción de onda corta de una capa depende de su composición, color y textura. Este parámetro cuantifica la cantidad de energía solar disponible que es absorbida por la superficie del pavimento
 - ***Conductividad térmica:*** se refiere a la cantidad de calor que fluye normalmente a través de una superficie unitaria por unidad de tiempo y por unidad de gradiente de temperatura.
 - ***Capacidad calórica:*** se refiere a la cantidad de energía calórica Q necesaria para cambiar la temperatura de una unidad de masa en un grado.
- **Caracterización de pavimentos de concreto**
- ***Módulo de elasticidad con respecto al tiempo:*** el módulo de elasticidad define la relación entre el esfuerzo (a compresión) y la deformación del material en el rango elástico de acuerdo a curvas esfuerzo-deformación durante el periodo de diseño, este módulo depende de características de la mezcla tales como relación agua-cemento y porcentaje de pasta y agregado.
 - ***Coefficiente de Poisson:*** los valores asumidos para el diseño de pavimentos de concreto se encuentran entre 0.15 y 0.18.

- **Resistencia a tensión o Módulo de Rotura (MR):** se define como el máximo esfuerzo de tensión hasta la rotura en la parte inferior de una viga de concreto simplemente apoyada durante un ensayo de tracción con cargas en los puntos tercios. Como todas las medidas de resistencia del pavimento de concreto, el MR está fuertemente influenciado por los parámetros de diseño de la mezcla que incluyen:
 - Tipo de cemento.
 - Contenido de cemento.
 - Presencia y tipo de aditivos.
 - Relación agua-cemento.
 - Propiedades del agregado (tipo de agregado, máximo tamaño de partículas, granulometría, forma y textura de partículas)

- **Resistencia a la tracción indirecta:** se refiere al máximo esfuerzo para el cual una probeta cilíndrica se rompe como resultado de la fuerza de tracción ortogonal resultante, si esta es sometida a una fuerza de compresión aplicada en una banda estrecha en toda su longitud.

- **Resistencia a compresión:** se define como el esfuerzo máximo que puede soportar el material bajo una carga de aplastamiento.

- **Peso unitario:** el peso unitario se define como el peso de la unidad de volumen del concreto

- **Coefficiente de expansión térmica:** se define como el cambio en unidad de longitud por el cambio en un grado de temperatura.

- **Esfuerzo último de retracción:** se refiere a la deformación por contracción que producirá el esfuerzo desarrollado por la exposición prolongada a condiciones de humedad, las cuales por definición son 40% de humedad

relativa. La contracción es considerada de dos maneras: permanente y transitoria.

- ***Coefficiente de absorción de onda corta, conductividad térmica y capacidad calórica.***

➤ **Caracterización de Base, Subbase y Subrasante**

- ***Módulo resiliente ajustado por temporadas:*** el módulo resiliente puede ser medido directamente en ensayos de laboratorio o a través de correlaciones a partir de otras propiedades de resistencia del material como el CBR. Este parámetro indica la capacidad de soporte de las capas de base, subbase y subrasante a partir de la cantidad de deformación recuperable para un esfuerzo determinado por medio de ensayos triaxiales dinámicos en muestras sometidas a cargas repetidas. La variación del módulo resiliente en temporadas del año donde se presenten diferentes condiciones de clima expresa variaciones en la capacidad de soporte de las capas que componen la estructura, por lo cual es de suma importancia su determinación.
- ***Coefficiente de Poisson:*** el coeficiente de Poisson es un parámetro requerido para la determinación de la respuesta estructural del pavimento, debido a que su efecto es insignificante su valor usualmente es asumido.

Propiedades requeridas para el modelo climático EICM (*Enhanced Integrated Climatic Model*):

- ***Índice de plasticidad (IP):*** es la diferencia entre el límite líquido y el límite plástico del suelo, e indica la magnitud del rango del contenido de humedad sobre el cual el suelo está en condición plástica.

- **Granulometría y tamaño efectivo de granos:** determina la distribución del tamaño de partículas.
- **Densidad máxima seca y contenido de humedad óptimo:** la densidad máxima seca es la mayor densidad posible en el terreno, el cual tiene una humedad adecuada (humedad óptima) en el momento de la compactación. Estos dos parámetros se pueden determinar a partir del ensayo de Proctor.
- **Gravedad específica:** se define como la relación de la masa de una unidad de volumen de un material a una temperatura determinada con la masa del mismo volumen de agua destilada a la misma temperatura.
- **Conductividad hidráulica saturada:** expresa la capacidad de un medio poroso para transmitir agua y depende principalmente de la estructura del suelo, de su contenido de humedad y de la temperatura del agua.
- **Porcentaje del grado de saturación:** es la proporción del contenido de vacíos que están ocupados por agua de una capa de base, subbase o subrasante.
- **Coefficiente de presión lateral (K_0):** expresa la relación entre la presión lateral de tierra y la presión vertical de tierra.

1.2.3.2 Caracterización del Tráfico⁵

La información relacionada con el tráfico es un parámetro importante ya que se necesita para estimar las cargas soportadas por el pavimento y la frecuencia de aplicación de estas durante la vida de diseño de la estructura. Para el procedimiento de esta guía de diseño los datos de tráfico requeridos son los mismos sin importar el tipo de pavimento a considerar (flexible o rígido).

Para el diseño estructural del pavimento se requieren cuatro (4) tipos de datos de tráfico, los cuales son enunciados a continuación:

➤ **Volumen de Tráfico - Información del Año Base**

El año base es definido como el primer año de funcionamiento del segmento de vía bajo diseño desde el momento en que es abierto al tráfico. Es requerida la siguiente información del año base:

- ***Promedio diario anual de camiones en ambos sentidos (Two-way Annual Average Daily Truck Traffic [Two-way AADT])***: es el volumen total del tráfico de camiones que pasan por un punto o segmento de la vía a ser diseñada en las dos direcciones en un periodo de 24 horas.
- ***Número de carriles en la dirección de diseño***: determinado por las especificaciones de diseño.
- ***Factor de distribución direccional (Directional Distribution Factor [DDF])***: es el porcentaje de camiones en la dirección de diseño, el cual es

⁵NATIONAL COOPERATIVE HIGHWAY RESEARCH PROGRAM. Background, Scope and Overview. En: Guide for Mechanistic-Empirical Design of New and Rehabilitated Pavement Structures. Part 2: Design Inputs. Chapter 4: Traffic. Illinois, 2004.

usualmente asumido como el 50% cuando el Tránsito Promedio Diario Anual y el Tránsito de Camiones Promedio Diario Anual son dados en dos direcciones.

- **Factor de distribución de carril (Truck Lane Distribution Factor [LDF]):** corresponde al porcentaje de camiones en el carril de diseño. Los valores por defecto (Nivel 3) utilizados para el LDF son:
 - Calzadas de un solo carril en una dirección → 1.0
 - Calzadas de dos carriles en una dirección → 0.90
 - Calzadas de tres carriles en una dirección → 0.60
 - Calzadas de cuatro carriles en una dirección → 0.45
- **Velocidad de operación:** es la velocidad media de desplazamiento que pueden lograr los usuarios en una carretera con una velocidad de diseño dada, bajo las condiciones prevalecientes del tránsito.

➤ Factores de Ajuste del Volumen de Tráfico

Para la caracterización del tráfico son requeridos los siguientes factores de ajuste:

- **Factor de ajuste mensual:** representa la proporción del tráfico de camiones anual para una clase de camión dada que ocurre en un mes específico. Si no hay información disponible se recomienda asumir una distribución mensual constante (1.0 para todos los meses y todas las clases de camiones).
- **Distribución por clase de vehículo:** representa el porcentaje de cada clase de camión dentro del TPDA (Tránsito Promedio Diario Anual) de camiones durante el año base. La suma de los porcentajes de todas las clases de camiones debe ser igual a 100.

- **Distribución Horaria de Camiones:** los factores de distribución horaria representan el porcentaje del TPD de camiones en cada hora del día.
- **Factores de Crecimiento:** los factores de crecimiento son utilizados para estimar los volúmenes de tráfico futuro de camiones a partir del porcentaje de crecimiento o decadencia de estos a través del tiempo. Para estimar estos factores se utilizan tres (3) funciones diferentes de crecimiento: crecimiento lineal, crecimiento exponencial y crecimiento nulo. Además, es posible utilizar diferentes funciones de crecimiento para cada clase de vehículo.

➤ **Factores de Distribución de Carga por Eje**

Los factores de distribución de carga por eje representan el porcentaje del total de aplicaciones del eje para cada intervalo de carga, para un tipo específico de eje (simple, tandem, tridem y cuádruple) y una clase de vehículo (C2-C3-C4, etc.) en cada mes.

➤ **Entradas Generales de Tráfico**

- **Media de ubicación de rueda:** esta medida se refiere a la distancia a la que se encuentra la huella de la rueda (desde el borde exterior del neumático) del borde del carril.
- **Desviación estándar de ubicación de rueda:** desviación estándar del movimiento lateral de los vehículos en la calzada (huella de la rueda). Esta medida es utilizada para predecir el comportamiento del pavimento frente a determinado número de aplicaciones de carga sobre un punto específico.

- **Ancho del carril de diseño:** este parámetro se refiere a la distancia entre marcas de líneas divisorias de carriles.
- **Número de ejes por clase de camión:** representa cuántos ejes tiene en promedio cada clase de vehículo para cada tipo de eje.
- **Configuración de eje:** se refiere a una serie de elementos que describen la configuración típica de las cargas en las llantas y en los ejes que serán aplicadas en el pavimento. Estos son:
 - *Ancho promedio del eje:* distancia entre extremos externos de las llantas del eje.
 - *Distancia entre llantas de ejes dobles:* distancia de centro a centro de llantas dobles.
 - *Espaciamiento entre ejes:* distancia entre dos ejes consecutivos tándem o un tridem.
 - *Presión de inflado*
- **Base de la rueda:** es el espaciamiento promedio entre ejes.

1.2.3.3 Caracterización del Clima⁶

Las condiciones ambientales a las que está expuesto el pavimento ya sea flexible o rígido tales como precipitación, temperatura y humedad, entre otros, tienen un efecto significativo en el comportamiento de la estructura, además, dichos factores externos pueden influir internamente en la estructura de pavimento, alterando las características iniciales de los materiales que componen sus capas y disminuyendo su capacidad de carga.

⁶NATIONAL COOPERATIVE HIGHWAY RESEARCH PROGRAM. Background, Scope and Overview. En: Guide for Mechanistic-Empirical Design of New and Rehabilitated Pavement Structures. Part 2: Design Inputs. Chapter 3: Environmental Effects. Illinois, 2004.

Los cambios en los perfiles de variables climáticas como la temperatura y la humedad en la estructura del pavimento a través de la vida de diseño son consideradas en la guía de diseño AASHTO 2002 a través de modelo climático llamado EICM, *Enhanced Integrated Climatic Model* por sus siglas en inglés, el cual simula los cambios en el comportamiento y las características del pavimento y los materiales de la subrasante causados por las condiciones climáticas a través de sus años de operación.

Los datos referentes a las condiciones climáticas requeridos para el análisis del comportamiento del pavimento son los siguientes, deben proporcionarse mínimo para 24 meses y en intervalos horarios:

- **Temperatura del aire horaria:** se refiere a la cantidad de energía calorífica acumulada en el aire.
- **Precipitación horaria:** este parámetro se mide en mm y equivale al espesor de la lámina de agua que se formaría a causa de la precipitación sobre una superficie de 1 m².
- **Velocidad del viento horaria:** es una medida del movimiento del aire producido por las diferencias de temperatura existentes en las diversas zonas de la Tierra y de la atmósfera.
- **Porcentaje de brillo solar horario:** el brillo solar es el número de horas de incidencia de los rayos solares sobre la superficie terrestre medida en horas.
- **Humedad relativa horaria:** se refiere a la relación porcentual entre la cantidad de vapor de agua real que contiene el aire y la que necesitaría contener para saturarse a idéntica temperatura.

- ***Profundidad del nivel freático:*** este parámetro influye en el contenido de humedad del suelo de fundación o subrasante del pavimento y por lo tanto en los módulos respectivos.

Las primeras 5 variables pueden ser obtenidas de estaciones meteorológicas cercanas al sitio del proyecto si es posible.

2. RECOPIACIÓN, PROCESAMIENTO Y ADAPTACIÓN DE LOS REQUERIMIENTOS DE LA GUIA AASHTO 2002 PARA EL TRÁFICO

2.1 FACTOR DE DISTRIBUCIÓN DIRECCIONAL

Si no existen mediciones directas en cada carril se asume que en cada dirección va la mitad (50%) del tránsito. A continuación se relaciona el factor direccional con la clase de vía a diseñar; las carreteras anchas tienen carriles de más de 3.5 m de ancho, las de ancho medio poseen un ancho de calzada entre 5 a 6 m y las carreteras estrechas una calzada de menos de 5 m de ancho.

Tabla 1. Factor de distribución direccional

Clasificación de la vía	% por dirección
Autopistas y vías multicarril con tres carriles por dirección	50
Autopistas y vías multicarril con dos carriles por dirección	50
Carreteras anchas y de dos direcciones	50
Carreteras de dos direcciones y con ancho medio	100
Carreteras estrechas y de dos direcciones	100

Fuente: Manual de diseño de pavimentos de concreto para vías con bajos, medios y altos volúmenes de tránsito. Instituto Nacional de Vías

2.2 FACTOR DE DISTRIBUCIÓN DE CARRIL

La asignación del tránsito para el carril de diseño en las carreteras de dos direcciones dependerá del ancho de la vía. Los factores de distribución vehicular para el carril de mayor sollicitación se muestran en la siguiente tabla:

Tabla 2. Porcentaje de vehículos para el carril de diseño

Clasificación de la vía	% para el carril de diseño	% del TPDA en el carril de diseño
Autopistas y vías multicarril con tres carriles por dirección	75	37.5
Autopistas y vías multicarril con dos carriles por dirección	90	45
Carreteras anchas y de dos direcciones	100	50
Carreteras de dos direcciones y con ancho medio	75	75
Carreteras estrechas y de dos direcciones	100	100

Fuente: Manual de diseño de pavimentos de concreto para vías con bajos, medios y altos volúmenes de tránsito. Instituto Nacional de Vías

2.3 VELOCIDAD DE OPERACIÓN

La velocidad de operación es la velocidad media de desplazamiento que pueden lograr los vehículos en una carretera con una velocidad de diseño dada, bajo las condiciones del tránsito prevalecientes, y se asume como el percentil 85 de la distribución de velocidades observadas en un lugar determinado. Si el tránsito y la interferencia son bajos la velocidad de operación puede llegar a ser muy similar a la velocidad de diseño.

Es necesario realizar estudios de tránsito para determinar la velocidad de operación de los vehículos que transitan por una vía determinada, sin embargo, a continuación se presenta una relación entre las velocidad de operación probables para una velocidad de diseño dada y un determinado volumen de tránsito⁷.

La siguiente tabla presenta la velocidad de diseño que se le podría asignar a un tramo en función de la categoría de la carretera y del tipo de terreno:

⁷ BRAVO, Pablo Emilio. Diseño de carreteras. Técnica y análisis del proyecto. 6 ed. Bogotá D.C, 1998. p. 139.

Tabla 3. Velocidad de diseño de los tramos homogéneos en función de la categoría de la carretera y el tipo de terreno

CATEGORÍA DE LA CARRETERA	TIPO DE TERRENO	PENDIENTE LONGITUDINAL (%)	VELOCIDAD DE DISEÑO DEL TRAMO HOMOGÉNEO (VTR) (km/h)																	
			20	30	40	50	60	70	80	90	100	110								
Primaria de dos calzadas	Plano	<3																		
	Ondulado	3-6																		
	Montañoso	6-8																		
	Escarpado	8																		
Primaria de una calzada	Plano	<3																		
	Ondulado	3-6																		
	Montañoso	6-8																		
	Escarpado	8																		
Secundaria	Plano	<3																		
	Ondulado	3-6																		
	Montañoso	6-8																		
	Escarpado	8																		
Terciaria	Plano	<3																		
	Ondulado	3-6																		
	Montañoso	6-8																		
	Escarpado	8																		

Fuente: Manual de diseño geométrico de carreteras. Capítulo 2. Instituto Nacional de Vías

A partir de la velocidad de diseño especificada dada según la categoría de la vía y el tipo de terreno se puede estimar la velocidad de operación una condición de tráfico determinada:

Tabla 4. Velocidades de operación promedio para diferentes condiciones de tráfico

VELOCIDAD DE DISEÑO (km/h)	CONDICIONES DE TRÁFICO		
	VOLUMENES BAJOS	VOLUMENES MEDIOS	VOLUMENES ALTOS
40	38	35	33
50	47	42	40

60	56	52	45
70	63	60	55
80	72	65	60
100	88	75	-
120	105	85	-

Fuente: Diseño de Carreteras. Paulo Emilio Bravo

2.4 TIPOLOGÍA DE VEHÍCULOS DE CARGA

La resolución 4100 de 2004 del Ministerio de Transporte de Colombia, en el artículo 4, reglamenta la tipología para vehículos automotores de carga para transporte terrestre, así como otros requisitos relaciones con dimensiones, máximos pesos brutos vehiculares y máximos pesos por eje, para su operación normal en la red vial en todo el territorio nacional. Los vehículos de carga se designan a partir de la configuración de sus ejes como se describe a continuación:


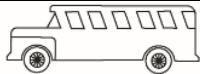

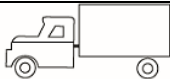
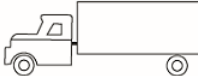
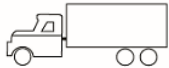
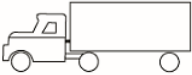
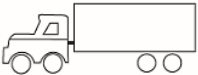
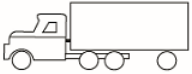
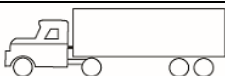
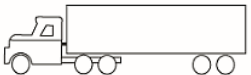
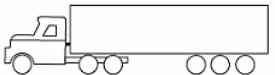
El primer dígito indica el número de ejes del camión o del tracto camión (cabazote).

La letra **S** hace referencia a semirremolque y el dígito inmediato el número de sus ejes.⁸

La tipología descrita en la siguiente tabla corresponde a lo estipulado en la resolución 4100 de 2004, de acuerdo con las definiciones, designación y clasificación establecidas en la Norma Técnica Colombiana NTC 4788 “Tipología para vehículos de transporte de carga terrestre” y la clasificación utilizada por el Instituto Nacional de Vías, INVIAS para la realización de aforos vehiculares.

⁸COLOMBIA. MINISTERIO DE TRANSPORTE. Resolución 4100 (28, diciembre, 2004). Por la cual se adoptan los límites de pesos y dimensiones en los vehículos de transporte terrestre automotor de carga por carretera, para su operación normal en la red vial a nivel nacional. Bogotá D.C: El Ministerio, 2004. p. 2.

Tabla 5. Tipología para vehículos de transporte de carga

TIPO		CONFIGURACIÓN
Buses	Buseta	
	Bus	
	Bus Metropolitano	
C₂P	Camión 2 ejes pequeño	
C₂G	Camión 2 ejes grande	
C₃ Y C₄	Camión C ₃	
	Tracto-Camión C ₂ -S ₁	
	Camión C ₄	
	Tracto-Camión C ₃ -S ₁	
	Tracto-Camión C ₂ -S ₂	
C₅	Tracto-Camión C ₃ -S ₂	
>C₅	Tracto-Camión C ₃ -S ₃	

Fuente: Volúmenes de Tránsito 2008. Instituto Nacional de Vías.

2.5 DIMENSIONES DE LOS VEHÍCULOS DE CARGA

Los vehículos de transporte de carga que circulen por el territorio nacional deben cumplir con las dimensiones establecidas en la resolución 4100 de 2004, art.7:

Tabla 6. Dimensiones máximas de los vehículos de carga

Designación	Dimensiones		
	Ancho máximo [m]	Altura máxima [m]	Longitud máxima [m]
C ₂	2.60	4.40	10.80
C ₃ y C ₄	2.60	4.40	12.20
2S ₁ , 2S ₂ , 3S ₁ , 3S ₂ , 3S ₃	2.60	4.40	18.50
Remolque (R) y remolque balanceado (B)	2.60	4.40	10.00
Semirremolque (S)	2.60	4.40	13.00







Fuente: Resolución 4100 de 2004, art 7. Ministerio de Transporte.

2.6 PESO BRUTO VEHICULAR DE VEHÍCULOS DE CARGA Y PESO MÁXIMO POR EJE

El peso bruto vehicular (PBV) se refiere al peso de un vehículo en condiciones de marcha más el máximo de carga que puede transportar. Para vehículos de transporte de carga a nivel nacional el PBV debe ser el establecido en la siguiente tabla, según lo dispuesto en la resolución 001782 de 2009 por la cual se modifica el artículo 8 de la resolución 4100 de 2004.




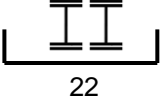



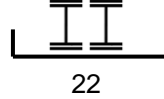

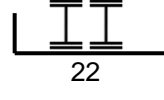

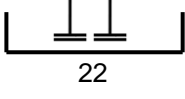

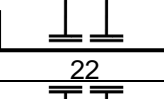
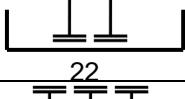

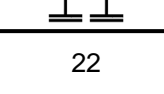
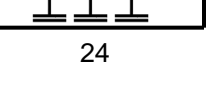
El máximo peso por eje para los vehículos de transporte de carga a nivel nacional se encuentra contemplado en el artículo 9 de la resolución 4100 de 2004.

Tabla 7. Peso máximo por eje

Tipo de Eje	Convención	Peso Máximo por Eje [Toneladas]
Eje sencillo, llanta sencilla. Dos (2) llantas		6
Eje sencillo, llanta doble. Cuatro (4) llantas		11
Eje tandem, llanta sencilla. Cuatro (4) llantas		11
Eje tandem, llanta mixta. Seis (6) llantas		17
Eje tandem, llanta doble. Ocho (8) llantas		22
Eje tridem, llanta doble, Doce (12) llantas		24

Fuente: Resolución 4100 de 2004, art 9. Ministerio de Transporte.

Tabla 8. Peso máximo bruto vehicular

Tipo	Esquema del Vehículo	Peso Máximo Bruto Vehicular [Toneladas]	Eje Direccional	2do. Eje del Tracto Camión	Eje del Semiremolque	Peso Máximo Bruto Vehicular con Tolerancia [Toneladas]
C ₂ -Pequeño		8.5	I 2.5	I 6		8.5
C ₂ -Grande		16	I 6	I 11		16.40
C ₃		28	I 6			28.70
C ₂ -S ₁		27	I 6	I 11	I 11	27.68
C ₄		36				36.9
C ₃ -S ₁		29	I 6		I 11	29.73
C ₂ -S ₂		32	I 6	I 11		32.8
C ₃ -S ₂		48	I 6			49.2
C ₃ -S ₃		52	I 6			53.3

Fuente: Resolución 001782 de 2009. Dirección de Tránsito de Bucaramanga

2.7 REGLAMENTACIÓN DEL TRÁNSITO DE TRANSPORTE DE CARGA EN LA CIUDAD DE BUCARAMANGA⁹

La Dirección de Tránsito de Bucaramanga en el ejercicio de sus funciones expide la reglamentación correspondiente al tránsito de transporte de carga en las vías del Municipio de Bucaramanga, contemplada en la resolución 735 de 2010 en la cual dispone algunas restricciones al tránsito de vehículos de carga.

En primera instancia, el plan de Ordenamiento Territorial, reglamentado por el Decreto No.078 de Junio de 2008, en el artículo 563 establece que “se prohíbe en la meseta de Bucaramanga, la circulación de vehículos de carga de más de 15 toneladas o más de tres ejes”.

La resolución en mención prohíbe el tránsito de vehículos cuya capacidad de carga supere las cuatro punto cinco (4.5) toneladas (excepto vehículos de emergencia, de transporte de valores, alimentos perecederos y oxígeno hospitalarios, y aquellos que estén dotados tecnológicamente para el mantenimiento de redes de servicios públicos esenciales), de lunes a sábados, hábiles, en los horarios expuestos a continuación y por los siguientes sectores y corredores viales del municipio de Bucaramanga:

➤ Horarios:

De 06:00 A.M a 09.00 A.M

De 12:00 P.M a 02:00 P.M

De 05:30 P.M a 08:00 P.M

⁹BUCARAMANGA. DIRECCIÓN DE TRÁNSITO DE BUCARAMANGA. Resolución 735 (6, diciembre, 2010). Por la cual se reglamenta el tránsito, cargue y descargue de los vehículos de carga, en las vías del Municipio de Bucaramanga. Bucaramanga: La Dirección, 2010.

➤ **Sectores y corredores viales:**

1. Primer Sector: Carrera 9ª a Carrera 27 y Av. Quebrada Seca a Av. Rosita Calle 48.
2. Segundo Sector: Carrera 27 a Carrera 36 y Calle 41 a Calle 56.
3. Diagonal 15 entre la Calle 45 y Calle 61 Puerta del Sol
4. Carrera 21 entre Diagonal 15 y Av. Quebrada Seca
5. Carrera 22 entre Av. Quebrada Seca y Calle 58
6. Carrera 27 entre Calle 9ª UIS y Calle 67 La Salle
7. Carrera 33 entre Av. Quebrada Seca y Calle 67 Carrera 30 Puerta del Sol.
8. Carrera 35 entre Calles 34 y 53
9. Carrera 36 entre Calles 56 y 34
10. Calle 56 entre Carrera 36 y Carrera 5ª Ciudadela Real de Minas.
11. Av. González Valencia entre Calle 45 y Calle 63 Urbanización CONUCOS.
12. Av. Rosita entre Diagonal 15 y Carrera 32 Parque Mejoras Públicas Glorieta Chicamocha.
13. Vía Bucaramanga-Girón entre Calle 61 Carrera 27 Puerta del Sol Hasta el puente el Bueno.
14. Calle 67 entre Carrera 27 La Salle y Carrera 33 Urbanización CONUCOS.
15. Calle 61 entre Diagonal 15 Puerta del Sol y Av. Los Búcaros Ciudadela Real de Minas.
16. Calle 64 entre Av. Los Búcaros Ciudadela Real de Minas y Puente El Bueno.
17. Calle 105 entre el Puente de Provenza Colegio INEM y Carrera 16 Barrio El Rocío.
18. Transversal Central Metropolitana entre la Calle 105 Carrera 16 Barrio El Rocío Puente El Bueno y Calle 64 Ciudadela Real de Minas.
19. Carrera 24 entre Calle 80 Urbanización NEPTUNO Barrio Diamante 2 y Avenida 89 Calle 70 Barrio San Luis Vía Bucaramanga-Girón.

20. Vía Antigua Bucaramanga-Floridablanca entre Carrera 33 Urbanización CONUCOS y Plaza Satélite Calle 107.

21. Perimetral Oriental entre Carrera 33 Urbanización CONUCOS Viaducto La Flora e Intersección SemafORIZADA Urbanizaciones Santa Bárbara y Tajamar.

Además se prohíbe el tránsito de vehículos cuya capacidad de carga sea mayor de 2.5 toneladas por el corredor vial de la Autopista Bucaramanga-Floridablanca, entre la Puerta del Sol, carrera 27, Diagonal 15, calle 61 y calle 105 del municipio de Bucaramanga en los horarios mencionados anteriormente.

De manera permanente (a ninguna hora del día) se prohíbe el tránsito de vehículos de carga en el corredor de la Carrera 15 entre la Avenida Quebrada Seca y la calle 45, y de aquellos cuya capacidad de carga supere las cuatro punto cinco (4.5) toneladas, por los siguientes corredores viales:

1. Calles 34 y 35 con Carreras 9ª y 5ª del Barrio Alfonso López.
2. Calles 56 y 61 con Carrera 6ª y Diagonal 15.
3. Calle 45 entre Carrera 9ª y Anillo Vial (Chimitá).
4. Calles 20 y 21 con Carreras 7ª y 4ª del Barrio Nariño.
5. Calle 21 entre carreras 32 y 28.
6. Calle 70 entre carreras 28 y 33 vía Antigua a Floridablanca.

Las vías del sector de Malpaso al barrio el Porvenir:

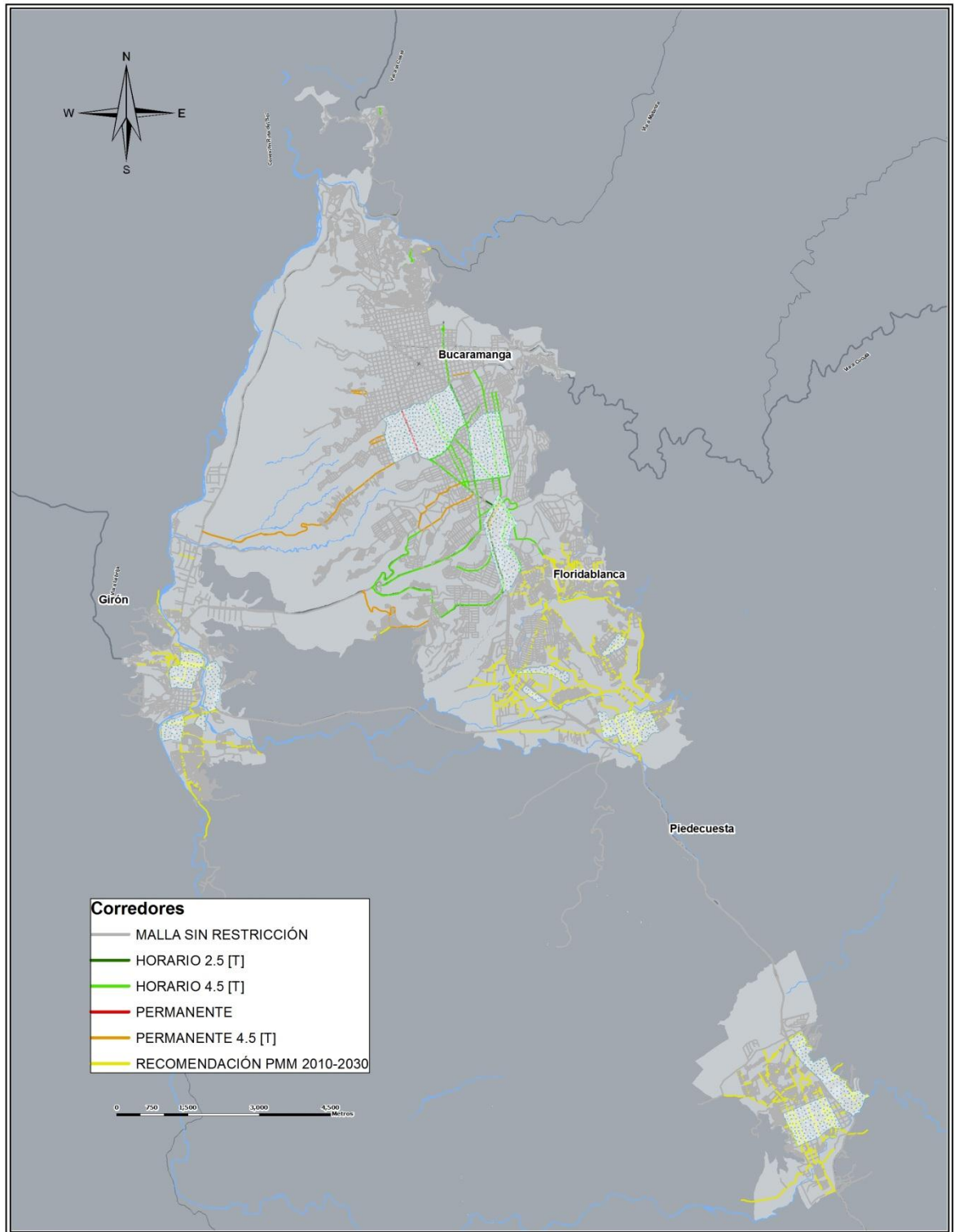
7. Calle 104E entre Carrera 15 y Carrera 10
8. Calle 105 entre Carrera 10 y Carrera 8
9. Carrera 8 entre Calle 105 y Calle 103 F (Iglesia)
10. Calle 103F entre Carrera 8 y Carrera 8A
11. Calle 103D entre Carrera 8A y Carrera 9

12. Carrera 9 entre Calle 103A y Calle 103D
13. Calle 103A entre Carrera 9 y Carrera 8Bis
14. Transversal El Porvenir entre Calle 103A Carrera 8 Bis y la Autopista Girón Bucaramanga.

Para resumir la información recopilada correspondiente a la normativa del transporte de carga en el Área Metropolitana de Bucaramanga se elaboró el mapa mostrado en la figura 4, que muestra las regiones o sectores en los cuales se prohíbe intermitentemente el paso de vehículos de tráfico pesado, los corredores con algún tipo de restricción de carga según las resoluciones vigentes, y se incluyeron además los corredores con restricciones sugeridas según el Plan Maestro de Movilidad de Bucaramanga 2010 – 2030¹⁰.

¹⁰ Grupo de investigación Geomática, gestión y optimización de sistemas, Universidad Industrial de Santander. Plan Maestro de Movilidad de Bucaramanga 2010 – 2030. Bucaramanga, 2011.

Figura 4. Restricciones de carga en el Área Metropolitana de Bucaramanga

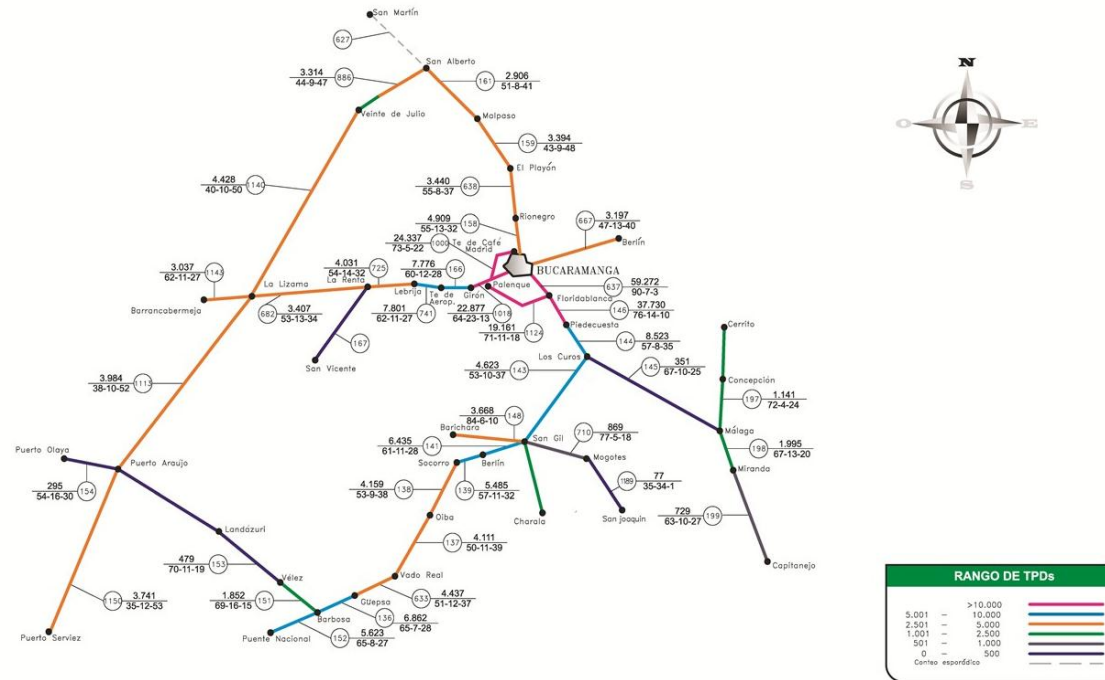


Fuente: Elaboración propia

2.8 VOLÚMENES DE TRÁNSITO Y COMPOSICIÓN VEHICULAR

2.8.1 Estaciones de Conteo-Vías de Santander

Figura 5. Estaciones de conteo INVIAS-Santander



	INSTITUTO NACIONAL DE VÍAS	SUBDIRECCIÓN DE APOYO TÉCNICO	PROYECTO Y PROCESO: EDGAR ALBERTO FONSECA HERRERA CONTRATO 3169 - 2009		TERRITORIAL No. 22 SANTANDER	VOLUMEN DE TRÁNSITO PROMEDIO DIARIO SEMANAL	FECHA DE PROCESO 16 Dic 2008
--	----------------------------	-------------------------------	--	--	------------------------------	---	---------------------------------

Fuente: Volúmenes de Tránsito 2008. Instituto Nacional de Vías

Los volúmenes de tránsito en el año 2008 proporcionados por el INVIAS se presentan en el Anexo A. Estos corresponden a resultados que se realizan de forma manual cada año durante siete días consecutivos las veinticuatro horas del día. Estas estaciones se clasificaron según el Tránsito Promedio Diario Semanal (TPDs) calculado a partir de la información obtenida en los conteos manuales de tránsito, según los siguientes parámetros:

➤ **Rango de TPDs:**

T1→ 0-500

T2→ 501-1000

T3→ 1001-2500

T4→ 2501-5000

T5→ 5001-10000

T6→10000

➤ **TPDs de 0-500**

Tabla 9. Estaciones de conteo con TPDs de 0-500

ESTACIÓN	SECTOR	TPDS	%A	%B	%C	C2P	C2G	C3-C4	C5	> C5	σ
145	LOS CUROS-MALAGA	351	67	10	23	39.61	60.04	0.36	0	0	29
153	VELEZ-LANDAZURI	479	70	11	19	44.31	52	3.23	0.31	0.15	57
154	PUERTO OLAYA-PUERTO ARAUJO	295	54	16	30	40.67	40.99	12.92	3.19	2.23	28
1189	MOGOTES-SAN JOAQUIN	77	65	34	1	100	0	0	0	0	19

Fuente: Volúmenes de Tránsito 2008. Instituto Nacional de Vías

➤ **TPDs de 501-1000**

Tabla 10. Estaciones de conteo con TPDs de 501-1000

ESTACIÓN	SECTOR	TPDS	%A	%B	%C	C2P	C2G	C3-C4	C5	> C5	σ
199	CAPIT NEJ O-MIRANDA	729	63	10	27	29.64	54.32	7.54	2.29	6.21	134
710	N GIL- MOGOTES	869	77	5	18	34.12	64.52	1.18	0.09	0.09	71

Fuente: Volúmenes de Tránsito 2008. Instituto Nacional de Vías

➤ **TPDs de 1001-2500**

Tabla 11. Estaciones de conteo con TPDs de 1001-2500

ESTACIÓN	SECTOR	TPDS	%A	%B	%C	C2P	C2G	C3-C4	C5	> C5	σ
151	VELEZ- B RBOSA	1852	69	16	15	47.58	45.93	4.85	0.21	1.44	141
197	CONCEPCI ON- MALAGA	1141	72	4	24	36.74	51.87	6.34	0.56	4.5	238

Fuente: Volúmenes de Tránsito 2008. Instituto Nacional de Vías

➤ **TPDs de 2501-5000**

Tabla 12. Estaciones de conteo con TPDs de 2501-5000

ESTACIÓN	SECTOR	TPDS	%A	%B	%C	C2P	C2G	C3-C4	C5	> C5	σ
137	VADO REAL-OIBA	4111	50	11	39	25.94	30.45	7.89	6.93	28.78	625
138	OIBA- SOCORRO	4159	53	9	38	28.84	31.33	8.72	9.1	22.02	410
143	SAN GIL- LOS CUROS	4623	53	10	37	28.56	32.63	9.14	6.58	23.1	416
148	BARICHAR A-SAN GIL	3668	84	6	10	23.03	72.45	2.5	0.52	.51	410

158	BUCARAMA NGA-RIO NEGRO	4909	55	13	32	27.82	35.68	6.76	3.48	26.26	199
159	EL PLAYON- MAL PASO	3394	43	9	48	15.99	23.38	5.66	4.12	50.85	1091
161	MAL PASO- SAN ALBERTO	2906	51	8	41	21.19	31.77	7.89	5.49	33.66	81
633	GUEPSA- VADO REAL	4437	51	12	37	29.86	31.89	8.04	8.14	22.07	487
638	RIONEGRO -EL PLAYON	3440	55	8	37	22.7	32.6	7.79	5.1	31.81	202
667	BUCARAMA NGA- BERLIN	3197	47	13	40	26.64	26.23	9.53	10.5 9	27.01	430
682	LA RENTA- LA LIZAMA	3407	53	13	34	21.07	43.81	11.1 7	5.97	17.98	298
725	LA RENTA- LEBRIJA	4031	54	14	32	22.33	46.91	8.31	4.31	18.14	334
886	20 DE JULIO-SAN ALBERTO	3314	44	9	47	14.36	21.44	12.6 6	11.3 6	40.18	546
1113	PUERTO ARAUJO-LA LIZAMA	3984	38	10	52	13.43	23.37	14.6 6	11.5 7	36.97	940
1140	LA LIZAMA- 20 DE JJULIO	4428	40	10	50	8.33	30.53	20.2 7	9.73	31.15	467
1143	LA LIZAMA- BARRANCA BERMEJA	3037	62	11	27	12.44	42.52	18.3 7	8.93	17.74	301
1150	PUERTO ARAUJO- PUERTO SERVIEZ	3741	35	12	53	15.39	19.79	11.3 3	14.6 4	38.85	455

Fuente: Volúmenes de Tránsito 2008. Instituto Nacional de Vías

➤ **TPDs de 5001-10000**

Tabla 13. Estaciones de conteo con TPDs de 5001-10000

ESTACIÓN	SECTOR	TPDS	%A	%B	%C	C2P	C2G	C3- C4	C5	> C5	σ
136	BARBOSA- GUEPSA	6862	65	7	28	22.8 4	40.97	8.87	8.39	18. 2	759

139	SOC RRO-BERLIN	5485	57	11	32	34.2 4	28.32	8.45	7.69	21.29	504
14	BERLIN-SAN GIL	6435	61	11	28	21.5 4	43.58	7.81	6.47	20.6	565
144	LOS CUROS-PIEDECUESTA	85 3	57	8	35	7.66	61.73	13.52	4.97	12.11	959
152	PUENTE NACIONAL-BARBOSA	5623	65	8	27	22.6 8	45.88	7.86	7.95	15.64	258
166	GIRON-TE DE AEROPUER TO	7776	60	12	28	35.5 4	34.53	14.32	4.62	11	497
198	MIRANDA-MALAGA	6862	67	13	20	32.8 7	51.54	9.58	1.98	4.03	759
741	LEBRIJA-TE DE AEROPUER TO	7801	62	11	27	31.2 1	37.92	10.41	5.34	15.12	626

Fuente: Volúmenes de Tránsito 2008. Instituto Nacional de Vías

➤ **TPDS >10000**

Tabla 14. Estaciones de conteo con TPDs>10000

ESTACIÓN	SECTOR	TPDS	% A	% B	% C	C2P	C2G	C3-C4	C5	> C5	σ
146	PIEDECUESTA-FLORIDABLANCA	37730	76	14	10	19.41	45.68	18.61	6.07	10.23	3250
637	FLORIDABLANCA BUCARAMANGA	9272	90	7	3	41.64	40.67	11.15	2.9	3.63	7754
1 00	PALENQUE-TE DE CAFÉ MADRID	24337	73	5	22	28.87	43.76	9.33	4.96	13.08	2925
1018	GIRON-BUCARAMANGA	22877	64	23	13	52.21	27.73	8.03	3.41	8.62	2958
1124	FLORIDABLANCA-PALENQUE	19161	71	11	18	21.64	45.8	15.01	5.6	11.95	1158

Fuente: Volúmenes de Tránsito 2008. Instituto Nacional de Vías

2.8.2 Tránsito Promedio Diario Anual (TPDA)

Debido a los costos que implica, la información sobre el número total de vehículos que pasan durante el año por un punto de referencia es difícil de obtener, por lo cual, el análisis de volúmenes de tránsito promedio diario anual TPDA se puede estimar a partir del tránsito promedio diario semanal por medio de la siguiente ecuación:

$$TPDA = TPDS \pm K$$

$$K = \left[\frac{B\sigma}{\sqrt{n}} \right] \left[\sqrt{\frac{N-n}{N-1}} \right]; B = \sqrt{\frac{1}{1-Cc}}$$

Donde:

N: Días del año (365 o 366)

n: Días del conteo (7)

B: Valoración adoptado para el coeficiente de confianza

σ : Desviación estándar

Cc: Porcentaje de Confiabilidad requerido

Para el cálculo del TPDA con un intervalo de confianza del 90% para la Estación No. 145, Sector Los Curos-Málaga de la Territorial Santander se tiene:

TPDS: 351

σ : 29

Cc: 90%

$$K = \left[\frac{\sqrt{\frac{1}{1-0.9}} \cdot 29}{\sqrt{7}} \right] \cdot \left[\sqrt{\frac{365-7}{7-1}} \right] = 34$$

$$TPDA = 351 + 34 = 385 \quad \text{Límite Superior}$$

$$TPDA = 351 - 34 = 317 \quad \text{Límite Inferior}$$

Lo cual significa que en un punto de este rango se puede obtener el TPDA de cualquier día del año con un coeficiente de confiabilidad del 90%. Para efectos de este análisis se trabajará con el mayor TPDA esperado.

2.8.3 Tránsito Promedio Diario Anual de Tráfico Pesado (TPDAT)

Se refiere al tráfico pesado total compuesto por camiones y buses correspondientes a un día del año, el cual se calcula a partir del TDA de la siguiente forma:

$$\text{Tráfico Pesado Total (TPDAT)} = \%Buses \cdot TPDA + \%Camiones \cdot TPDA$$

2.8.4 Distribución Vehicular

Con la información obtenida en los conteos manuales de tránsito, es posible establecer el tránsito de interés para el análisis de la metodología AASHTO 2002, lo cual se refiere a la distribución porcentual de transporte de carga y buses correspondiente al TPDA, discretizando por tipo de camión.

A partir del tránsito promedio diario anual de tráfico pesado (TPDAT), se calcula la composición vehicular de este por tipo de camión:

$$\%Buses = \frac{\%Buses \cdot TPDA}{TPDAT}$$

$$\%C2P = \frac{\%C2P \cdot (\%Camiones \cdot TPDA)}{TPDAT}$$

Para el cálculo de la distribución vehicular para la Estación No.145, Sector Los Curos-Málaga de la Territorial Santander se tiene:

$$TPDAT = 10\% \cdot 385 + 23\% \cdot 385 = 128$$

$$\%C2P = \frac{\%39.6 \cdot (23\% \cdot 385)}{128} = 27.5\%$$

Los resultados obtenidos para las estaciones de conteo analizadas se muestran a continuación:

➤ **TPDs de 0-500**

Tabla 15. TPDA, TPDAT y distribución vehicular para estaciones con TPDS de 0-500

ESTACIÓN	SECTOR	TPDA MÁX	TOTAL B	TOTAL C	TPDAT	%BUS	%C2P	%C2G	%C3 - C4	%C5	% > C5
145	LOS CUROS-MALAGA	385	39	89	128	30.47%	27.54%	41.75%	0.25%	0.00%	0.00%
153	VELEZ-LANDAZURI	547	60	104	164	36.59%	28.10%	32.98%	2.05%	0.20%	0.10%
154	PUERTO OLAYA-PUERTO ARAUJO	328	53	98	151	35.10%	26.40%	26.60%	8.39%	2.07%	1.45%
1189	MOGOTES-SAN JOAQUIN	100	34	1	35	97.14%	2.86%	0.00%	0.00%	0.00%	0.00%

Fuente: Elaboración Propia

➤ **TPDs de 501-1000**

Tabla 16. TPDA, TPDAT y distribución vehicular para estaciones con TPDS de 501-1000

ESTACIÓN	SECTOR	TPDA MÁX	TOTAL B	TOTAL C	TPDAT	%BUS	%C2P	%C2G	%C3 - C4	%C5	% > C5
199	CAPITANEJO-MIRANDA	888	89	240	329	27.05%	21.62%	39.63%	5.50%	1.67%	4.53%
710	SAN GIL-MOGOTES	953	48	172	220	21.82%	26.68%	50.44%	0.92%	0.07%	0.07%

Fuente: Elaboración Propia

➤ **TPDs de 1001-2500**

Tabla 17. TPDA, TPDAT y distribución vehicular para estaciones con TPDS de 1001-2500

ESTACIÓN	SECTOR	TPDA MÁX	TOTAL B	TOTAL C	TPDAT	%BUS	%C2P	%C2G	%C3 - C4	%C5	% > C5
151	VELEZ-BARBOSA	2019	323	303	626	51.60%	23.03%	22.23%	2.35%	0.10%	0.70%
197	CONCEPCION-MALAGA	1423	57	342	399	14.29%	31.49%	44.46%	5.43%	0.48%	3.86%

Fuente: Elaboración Propia

➤ **TPDs de 2501-5000**

Tabla 18. TPDA, TPDAT y distribución vehicular para estaciones con TPDS de 2501-5000

ESTACIÓN	SECTOR	TPDA MÁX	TOTAL B	TOTAL C	TPDAT	%BUS	%C2P	%C2G	%C3 - C4	%C5	% > C5
137	VADO REAL-OIBA	4852	534	1892	2426	22.01%	20.23%	23.75%	6.15%	5.40%	22.45%
138	OIBA-SOCORRO	4645	418	1765	2183	19.15%	23.32%	25.33%	7.05%	7.36%	17.80%
143	SAN GIL-LOS CUROS	5116	512	1893	2405	21.29%	22.48%	25.68%	7.19%	5.18%	18.18%
148	BARICHARA-SAN GIL	4154	249	415	664	37.50%	14.39%	45.28%	1.56%	0.33%	0.94%
158	BUCARAMANGA-RIO NEGRO	5145	669	1646	2315	28.90%	19.78%	25.37%	4.81%	2.47%	18.67%
159	EL PLAYON-MAL PASO	4687	422	2250	2672	15.79%	13.46%	19.69%	4.77%	3.47%	42.82%

161	MAL PASO-SAN ALBERTO	3002	240	1231	1471	16.32%	17.73%	26.59%	6.60%	4.59%	28.17%
633	GUEPSA-VADO REAL	5014	602	1855	2457	24.50%	22.54%	24.08%	6.07%	6.15%	16.66%
638	RIONEGRO-EL PLAYON	3679	294	1361	1655	17.76%	18.67%	26.81%	6.41%	4.19%	26.16%
667	BUCARAMANGA-BERLIN	3707	482	1483	1965	24.53%	20.11%	19.80%	7.19%	7.99%	20.38%
682	LA RENTA-LA LIZAMA	3760	489	1278	1767	27.67%	15.24%	31.69%	8.08%	4.32%	13.00%
725	LA RENTA-LEBRIJA	4427	620	1417	2037	30.44%	15.53%	32.63%	5.78%	3.00%	12.62%
886	20 DE JULIO-SAN ALBERTO	3961	357	1862	2219	16.09%	12.05%	17.99%	10.62%	9.53%	33.72%
1113	PUERTO ARAUJO-LA LIZAMA	5098	510	2651	3161	16.13%	11.26%	19.60%	12.29%	9.70%	31.01%
1140	LA LIZAMA-20 DE JULIO	4982	498	2491	2989	16.66%	6.94%	25.44%	16.89%	8.11%	25.96%
1143	LA LIZAMA-BARRANCABERMEJA	3394	373	916	1289	28.94%	8.84%	30.22%	13.05%	6.35%	12.61%
1150	PUERTO ARAUJO-PUERTO SERVIEZ	4280	514	2269	2783	18.47%	12.55%	16.13%	9.24%	11.94%	31.67%

Fuente: Elaboración Propia

➤ **TPDs de 5001-10000**

Tabla 19. TPDA, TPDAT y distribución vehicular para estaciones con TPDS de 5001-10000

ESTACIÓN	SECTOR	TPDA MÁX	TOTAL B	TOTAL C	TPDAT	%BUS	%C2P	%C2G	%C3 - C4	%C5	% > C5
136	BARBOSA-GUEPSA	7762	543	2173	2716	19.99%	18.27%	32.78%	7.10%	6.71%	15.14%
139	SOCORRO-BERLIN	6082	669	1946	2615	25.58%	25.48%	21.07%	6.29%	5.72%	15.84%
141	BERLIN-SAN GIL	7105	782	1989	2771	28.22%	15.46%	31.28%	5.61%	4.64%	14.79%
144	LOS CUROS-PIEDECUESTA	9660	773	3381	4154	18.61%	6.23%	50.24%	11.00%	4.05%	9.86%
152	PUENTE NACIONAL-BARBOSA	5929	474	1601	2075	22.84%	17.50%	35.40%	6.06%	6.13%	12.07%
166	GIRON-TE DE AEROPUERTO	8365	1004	2342	3346	30.01%	24.88%	24.17%	10.02%	3.23%	7.70%
198	MIRANDA-MALAGA	7762	1009	1552	2561	39.40%	19.92%	31.23%	5.81%	1.20%	2.44%
741	LEBRIJA-TE DE AEROPUERTO	8543	940	2307	3247	28.95%	22.17%	26.94%	7.40%	3.79%	10.74%

Fuente: Elaboración Propia

➤ **TPDs > 10000**

Tabla 20. TPDA, TPDAT y distribución vehicular para estaciones con TPDS mayor a 10000

ESTACIÓN	SECTOR	TPDA MAX	TOTAL L B	TOTAL C	TPDAT	%BUS	%C2P	%C2G	%C3 - C4	%C5	% > C5
146	PIEDRECUESTA- FLORIDABLANCA	41582	5822	4158	9980	58.34%	8.09%	19.03%	7.75%	2.53%	4.26%
637	FLORIDABLANCA- BUCARAMANGA	68463	4792	2054	6846	70.00%	12.49%	12.20%	3.35%	0.87%	1.09%
1000	PALENQUE-TE DE CAFÉ MADRID	27804	1390	6117	7507	18.52%	23.52%	35.66%	7.60%	4.04%	10.66%
1018	GIRON- BUCARAMANGA	26383	6068	3430	9498	63.89%	18.85%	10.01%	2.90%	1.23%	3.11%
1124	FLORIDABLANCA- PALENQUE	20534	2259	3696	5955	37.93%	13.43%	28.43%	9.32%	3.48%	7.42%

Fuente: Elaboración Propia

2.8.5 Variación horaria

La variación horaria del tránsito depende del tipo de vía y de la actividad que prevalezca en ella. Esta variación refleja el porcentaje de camiones que pasan por un punto en cada hora de un día. A partir de la información disponible en la subdirección de apoyo técnico del Instituto Nacional de Vías se obtuvo la variación horaria del tránsito de camiones para cada estación estudiada. A continuación se muestra los resultados en función del tránsito promedio diario semanal (TPDs) de la estación:

➤ **TPDs de 0-500**

Tabla 21. Variación horaria para estaciones con TPDS de 0-500

ESTACIÓN	SECTOR\HORA	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
145	LOS CUROS-MALAGA	1.3 %	0.5 %	1.0 %	2.2 %	3.5 %	5.0 %	6.3 %	5.2 %	3.5 %	6.0 %	5.7 %	6.1 %	3.6 %	6.4 %	5.6 %	5.1 %	4.6 %	3.5 %	3.3 %	4.7 %	4.7 %	6.3 %	2.7 %	3.0 %
153	VELEZ-LANDAZURI	0.6 %	0.5 %	1.2 %	0.8 %	1.8 %	5.5 %	5.7 %	8.4 %	8.5 %	8.9 %	6.9 %	5.3 %	4.9 %	6.3 %	5.0 %	3.5 %	7.6 %	4.4 %	4.6 %	4.5 %	2.8 %	1.6 %	0.7 %	0.2 %
154	PUERTO OLAYA-PUERTO ARAUJO	1.2 %	0.8 %	0.9 %	0.9 %	3.0 %	4.6 %	5.9 %	4.6 %	6.0 %	5.1 %	6.6 %	5.7 %	7.7 %	5.3 %	5.9 %	4.5 %	6.7 %	5.0 %	5.1 %	4.9 %	3.0 %	3.2 %	1.6 %	1.6 %
1189	MOGOTES-SAN JOAQUIN	5.9 %	3.2 %	3.7 %	3.2 %	8.6 %	3.2 %	3.2 %	1.6 %	1.1 %	1.6 %	3.2 %	4.3 %	8.0 %	5.9 %	5.9 %	2.7 %	8.0 %	4.8 %	4.8 %	3.2 %	3.7 %	4.8 %	2.1 %	3.2 %

Fuente: Volúmenes de tránsito 2008. INVIAS

➤ **TPDs de 501-1000**

Tabla 22. Variación horaria para estaciones con TPDS de 501-1000

ESTACIÓN	SECTOR\HORA	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
199	CAPITANEJO-MIRANDA	1.8 %	1.7 %	2.3 %	3.8 %	4.7 %	6.2 %	6.7 %	3.9 %	5.7 %	4.2 %	5.8 %	6.0 %	5.3 %	3.7 %	6.0 %	5.8 %	3.5 %	3.8 %	5.1 %	3.0 %	3.8 %	3.2 %	2.4 %	1.8 %
710	SAN GIL-MOGOTES	1.1 %	1.0 %	0.5 %	2.3 %	2.9 %	4.2 %	6.2 %	4.5 %	8.3 %	5.9 %	7.5 %	7.9 %	5.4 %	6.1 %	6.7 %	6.4 %	5.3 %	4.7 %	4.5 %	4.0 %	2.1 %	1.4 %	0.5 %	0.8 %

Fuente: Volúmenes de tránsito 2008. INVIAS

➤ TPDs de 1001-2500

Tabla 23. Variación horaria para estaciones con TPDS de 1001-2500

ESTACIÓN	SECTOR\HORA	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
151	VELEZ-BARBOSA	0.7 %	0.3 %	0.7 %	1.0 %	2.4 %	3.0 %	4.5 %	5.4 %	7.6 %	7.4 %	6.3 %	6.6 %	6.1 %	7.2 %	5.9 %	6.5 %	7.1 %	6.4 %	5.4 %	3.6 %	2.3 %	2.0 %	0.7 %	0.7 %
197	CONCEPCIÓN-MÁLAGA	2.4 %	1.9 %	2.0 %	2.6 %	3.9 %	3.5 %	4.1 %	5.4 %	5.9 %	6.2 %	5.7 %	5.3 %	5.2 %	5.4 %	5.6 %	4.7 %	5.8 %	6.2 %	4.2 %	3.4 %	2.1 %	3.1 %	3.1 %	2.2 %

Fuente: Volúmenes de tránsito 2008. INVIAS

➤ TPDs de 2501-5000

Tabla 24. Variación horaria para estaciones con TPDS de 2501-5000

ESTACIÓN	SECTOR\HORA	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
137	VADO REAL-OIBA	4.1 %	4.2 %	4.1 %	3.5 %	4.3 %	4.5 %	4.8 %	3.8 %	3.4 %	3.7 %	3.8 %	3.6 %	3.7 %	3.2 %	3.9 %	3.8 %	5.0 %	4.5 %	5.5 %	4.6 %	5.5 %	4.9 %	3.9 %	3.6 %
138	OIBA-SOCORRO	3.7 %	4.2 %	4.3 %	3.9 %	3.4 %	5.0 %	5.4 %	4.8 %	3.6 %	3.4 %	4.0 %	3.9 %	3.4 %	3.8 %	3.8 %	4.0 %	4.7 %	4.8 %	4.7 %	4.5 %	4.8 %	4.6 %	3.5 %	3.8 %
143	SAN GIL-LOS CUROS	2.9 %	2.5 %	2.4 %	2.8 %	4.0 %	4.3 %	5.3 %	5.4 %	5.2 %	4.4 %	4.3 %	4.4 %	4.6 %	4.0 %	4.6 %	4.6 %	4.5 %	5.1 %	5.3 %	4.5 %	4.2 %	3.2 %	4.1 %	3.4 %
148	BARICHARA-SAN GIL	0.5 %	0.3 %	0.4 %	0.4 %	1.5 %	2.6 %	4.6 %	7.8 %	6.5 %	7.2 %	8.0 %	8.1 %	5.8 %	5.8 %	7.0 %	6.6 %	7.0 %	6.8 %	5.4 %	3.4 %	2.0 %	1.1 %	0.7 %	0.3 %
158	BUCARAMANGA-RIO NEGRO	1.7 %	2.3 %	1.7 %	2.1 %	3.7 %	4.6 %	5.9 %	5.1 %	5.3 %	5.1 %	5.5 %	4.9 %	4.8 %	5.2 %	4.7 %	5.1 %	4.7 %	4.5 %	5.6 %	4.9 %	4.0 %	3.2 %	3.0 %	2.3 %
159	EL PLAYÓN-MAL PASO	1.9 %	1.5 %	1.7 %	2.0 %	1.9 %	3.4 %	4.6 %	4.0 %	3.9 %	4.1 %	3.7 %	4.5 %	26.0 %	4.0 %	4.1 %	3.4 %	3.5 %	4.0 %	3.8 %	3.0 %	3.3 %	3.1 %	2.3 %	2.0 %
161	MAL PASO-SAN ALBERTO	2.7 %	2.2 %	2.1 %	2.2 %	3.0 %	4.2 %	5.6 %	5.6 %	4.8 %	4.8 %	5.5 %	4.8 %	5.2 %	4.9 %	5.2 %	4.5 %	4.8 %	5.0 %	4.5 %	4.5 %	3.7 %	3.8 %	3.3 %	3.0 %
633	GUEPSA-VADO	4.0 %	4.1 %	4.0 %	3.3 %	3.9 %	4.5 %	5.1 %	4.2 %	3.9 %	4.0 %	4.0 %	3.8 %	4.1 %	3.9 %	3.9 %	4.2 %	4.5 %	5.3 %	4.6 %	4.9 %	4.2 %	4.3 %	3.6 %	3.6 %

	REAL	%	%	%	%	%	%	%	%	%	%	%	%	%	%	%	%	%	%	%	%	%	%	%	%
638	RIONEGRO-EL PLAYON	2.3%	2.0%	2.0%	2.7%	3.4%	4.5%	5.6%	5.2%	5.5%	4.6%	5.4%	5.2%	4.7%	5.1%	5.0%	4.6%	4.1%	4.6%	5.0%	4.4%	4.0%	3.6%	3.5%	3.1%
667	BUCARAMANGA-BERLIN	3.5%	4.1%	3.1%	4.0%	3.8%	5.0%	6.0%	4.3%	3.4%	2.9%	3.6%	3.9%	4.3%	3.3%	4.8%	4.9%	5.5%	4.5%	5.6%	5.3%	3.8%	3.0%	3.4%	4.1%
682	LA RENTA-LA LIZAMA	1.6%	1.1%	1.0%	1.3%	2.6%	5.5%	7.8%	5.8%	4.1%	4.1%	4.7%	5.0%	5.1%	5.4%	5.3%	5.3%	5.7%	5.6%	5.4%	5.4%	4.5%	3.2%	2.7%	1.7%
725	LA RENTA-LEBRIJA	1.3%	1.3%	1.0%	1.6%	3.4%	5.3%	6.6%	4.9%	4.9%	4.6%	5.0%	5.3%	5.3%	5.4%	5.4%	4.9%	5.9%	6.0%	5.4%	4.8%	4.0%	3.5%	2.6%	1.9%
886	20 DE JULIO-SAN ALBERTO	2.0%	2.4%	2.5%	3.0%	3.7%	5.1%	5.6%	5.3%	4.8%	4.7%	4.6%	5.1%	5.3%	5.6%	4.9%	4.6%	5.3%	5.2%	4.6%	4.1%	3.8%	2.8%	3.1%	2.0%
1113	PUERTO ARAUJO-LA LIZAMA	3.0%	2.8%	2.8%	2.7%	3.5%	5.0%	5.5%	5.2%	5.5%	4.8%	4.3%	4.7%	5.2%	5.0%	4.8%	5.2%	4.7%	5.2%	4.4%	4.3%	3.2%	3.0%	2.8%	2.6%
1140	LA LIZAMA-20 DE JJULIO	1.7%	2.1%	1.9%	2.0%	2.4%	3.8%	6.7%	6.2%	5.4%	5.8%	5.5%	5.5%	5.3%	6.2%	5.2%	5.3%	4.6%	5.2%	4.9%	4.1%	3.0%	2.7%	2.3%	2.1%
1143	LA LIZAMA-BARRANCABERM EJA	0.4%	0.4%	0.3%	0.7%	1.8%	4.0%	6.5%	7.4%	6.2%	5.3%	5.7%	5.7%	6.1%	6.0%	6.7%	6.2%	6.2%	6.6%	5.3%	4.3%	3.2%	2.4%	1.6%	0.8%
1150	PUERTO ARAUJO-PUERTO SERVEZ	3.4%	3.3%	2.9%	3.1%	3.1%	3.2%	4.5%	4.7%	4.6%	6.4%	6.1%	5.9%	4.3%	4.4%	5.1%	4.7%	5.6%	4.8%	3.7%	3.5%	2.9%	3.3%	3.2%	3.2%

Fuente: Volúmenes de tránsito 2008. INVIAS

➤ **TPDs de 5001-10000**

Tabla 25. Variación horaria para estaciones con TPDS de 5001-10000

ESTACIÓN	SECTOR/HORA	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
136	BARBOSA-GUEPSA	3.7%	3.0%	3.9%	3.4%	3.7%	3.9%	4.7%	4.0%	4.6%	3.8%	4.7%	4.2%	4.2%	3.6%	4.6%	4.7%	4.9%	4.9%	5.6%	4.7%	4.3%	3.8%	3.9%	3.3%
139	SOCORRO-BERLIN	3.5%	2.8%	3.1%	2.8%	4.0%	3.9%	5.4%	5.2%	4.5%	4.0%	4.0%	4.3%	4.2%	3.9%	4.1%	4.7%	4.7%	4.8%	6.5%	4.8%	4.5%	4.0%	3.4%	3.0%
141	BERLIN-SAN GIL	3.1%	2.7%	2.5%	2.6%	3.6%	3.8%	5.8%	5.2%	4.6%	4.5%	4.5%	4.4%	4.6%	4.2%	4.5%	4.8%	5.1%	4.7%	5.5%	4.9%	4.0%	3.6%	3.3%	3.5%

144	LOS CUROS-PIEDRECUESTA	6.1 %	5.4 %	6.3 %	5.7 %	5.7 %	5.8 %	5.5 %	5.0 %	5.4 %	5.6 %	5.7 %	4.5 %	4.4 %	4.1 %	3.2 %	3.5 %	2.9 %	2.5 %	1.5 %	1.4 %	1.6 %	1.9 %	3.2 %	3.5 %
152	PUENTE NACIONAL-BARBOSA	3.3 %	3.7 %	3.8 %	3.6 %	3.7 %	4.0 %	4.7 %	4.3 %	4.6 %	4.6 %	4.1 %	4.3 %	3.9 %	3.6 %	3.7 %	4.3 %	4.2 %	4.7 %	5.1 %	5.3 %	4.9 %	4.6 %	4.1 %	2.8 %
166	GIRON-TE DE AEROPUERTO	1.3 %	1.2 %	1.2 %	1.8 %	3.5 %	4.0 %	5.3 %	5.3 %	6.3 %	5.4 %	5.9 %	5.6 %	5.6 %	5.8 %	5.8 %	5.6 %	5.9 %	5.2 %	4.7 %	4.3 %	3.7 %	2.8 %	2.0 %	1.7 %
198	MIRANDA-MALAGA	1.3 %	1.1 %	1.3 %	2.0 %	4.3 %	4.3 %	4.7 %	6.8 %	6.8 %	6.0 %	6.5 %	6.4 %	5.8 %	5.4 %	5.9 %	4.4 %	5.4 %	5.8 %	3.2 %	3.4 %	3.2 %	2.5 %	2.1 %	1.4 %
741	LEBRIJA-TE DE AEROPUERTO	1.5 %	1.6 %	1.5 %	2.2 %	3.7 %	4.8 %	5.0 %	4.7 %	5.2 %	5.2 %	5.2 %	5.2 %	5.4 %	4.9 %	5.0 %	5.4 %	5.2 %	5.4 %	5.7 %	4.7 %	4.6 %	3.3 %	2.4 %	2.2 %

Fuente: Volúmenes de tránsito 2008. INVIAS

➤ TPDs > 10000

Tabla 26. Variación horaria para estaciones con TPDS de 5001-10000

ESTACIÓN	SECTOR\HORA	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	
146	PIEDRECUESTA-FLORIDABLANCA	0.9 %	1.1 %	1.0 %	1.1 %	1.9 %	5.3 %	4.5 %	5.1 %	5.3 %	5.3 %	4.7 %	5.2 %	4.7 %	4.8 %	4.7 %	3.1 %	4.9 %	4.9 %	6.0 %	8.3 %	6.9 %	5.9 %	2.8 %	1.8 %	377
637	FLORIDABLANCA-BUCARAMANGA	0.8 %	1.0 %	1.0 %	1.2 %	1.4 %	2.3 %	4.4 %	5.8 %	6.2 %	5.9 %	5.9 %	5.6 %	5.8 %	5.3 %	5.7 %	5.9 %	5.7 %	6.2 %	5.4 %	5.3 %	4.7 %	4.4 %	2.9 %	1.3 %	592
1000	PALENQUE-TE DE CAFÉ MADRID	1.1 %	1.1 %	1.8 %	2.1 %	3.0 %	7.8 %	5.1 %	5.5 %	6.1 %	6.5 %	6.4 %	5.7 %	5.9 %	5.3 %	5.3 %	2.2 %	5.3 %	5.2 %	5.1 %	4.3 %	3.0 %	2.5 %	1.9 %	1.6 %	243
1018	GIRON-BUCARAMANGA	0.9 %	0.8 %	0.8 %	0.9 %	1.4 %	4.1 %	3.9 %	6.3 %	6.9 %	6.6 %	6.8 %	5.6 %	4.5 %	5.2 %	6.4 %	5.2 %	6.6 %	7.0 %	6.2 %	4.1 %	3.9 %	2.6 %	2.0 %	1.6 %	228
1124	FLORIDABLANCA-PALENQUE	1.0 %	1.0 %	1.0 %	1.3 %	2.0 %	6.5 %	5.6 %	6.9 %	6.7 %	6.4 %	6.3 %	6.2 %	5.5 %	5.6 %	5.6 %	3.3 %	5.5 %	4.9 %	4.8 %	3.9 %	3.5 %	2.9 %	2.3 %	1.5 %	191
																										61

Fuente: Volúmenes de tránsito 2008. INVIAS

2.9 ESPECTRO DE CARGA

En Colombia existe un estudio acerca de la distribución de cargas por eje para los diferentes tipos de eje por cada mil camiones en función del tránsito promedio diario semanal (TPDs), realizado a partir del análisis de series históricas de estaciones de pesaje a cargo del INVIAS de los años 1987 a 2000.¹¹

El espectro completo establecido para la red vial nacional está consignado en el ANEXO B para ejes sencillos, tándem y tridem.

Tabla 27. Número de ejes sencillos, tandem y tridem por cada mil camiones para vías con diferentes TPDs

EJE	Ejes por 100 camiones-TPDS					
	0 - 500	500 - 1.000	1.000 - 2.500	2.500 - 5.000	5.000 - 10.000	Mayor a 10.000
Sencillo	1879	1876	1796	1760	1782	1839
Tándem	114	120	228	282	253	177
Tridem	23	30	67	76	64	38

Fuente: Elaboración propia

Con base en los espectros para cada tipo de eje es posible obtener el espectro porcentual para la red vial nacional que se muestra en las siguientes tablas para los ejes simples, tándem y tridem:

¹¹ GARCIA ALADIN, Maria Fernanda. Catálogo de diseño de pavimentos rígidos de la PCA adaptado a las condiciones de tránsito colombianas. Popayán: UNICAUCA, 2002. p. 191.

Tabla 28. Distribución porcentual para el eje sencillo

CARGA EJE SIMPLE	TPDS						
	KN	0 - 500	500 - 1.000	1.000 - 2.500	2.500 - 5.000	5.000 - 10.000	Mayor a 10.000
19.6	2.29%	2.24%					
21.56			2.17%	2.10%			
23.52					2.13%	2.23%	
29.4	19.64%	19.56%					
32.34			18.60%	18.18%			
35.28					18.52%	19.14%	
39.2	21.50%	21.48%					
43.12			21.10%	20.91%			
47.04					21.10%	21.37%	
49	12.03%	12.21%					
53.9			14.31%	15.23%			
58.8					14.59%	13.10%	
58.8	6.33%	6.34%					
64.68			7.35%	7.84%			
70.56					7.46%	6.74%	
68.6	3.89%	3.84%					
75.46			3.84%	3.81%			
82.32					3.82%	3.86%	
78.4	3.67%	3.68%					
86.24			3.51%	3.47%			
94.08					3.48%	3.59%	
88.2	4.47%	4.48%					
97.02			4.23%	4.15%			
105.84					4.21%	4.35%	
98	4.42%	4.42%					
107.8			4.23%	4.09%			
117.6					4.15%	4.35%	
107.8	6.17%	6.18%					
118.58			5.85%	5.74%			
129.36					5.84%	6.04%	
117.6	6.44%	6.45%					
129.36			6.12%	5.97%			
141.12					6.06%	6.31%	
127.4	5.48%	5.44%					

140.14			5.18%	5.06%		
152.88					5.16%	5.33%
137.2	2.66%	2.67%				
150.92			2.56%	2.50%		
164.64					2.53%	2.61%
147	1.01%	1.01%				
161.7			0.95%	0.97%		
176.4					0.95%	0.98%

Fuente: Elaboración propia

Tabla 29. Distribución porcentual para el eje tandem

CARGA EJE TANDEM	TPDS						
	KN	0 - 500	500 - 1.000	1.000 - 2.500	2.500 - 5.000	5.000 - 10.000	Mayor a 10.000
58.8	1.75%	1.67%					
64.68			1.75%	1.77%			
70.56					1.98%	1.69%	
78.4	7.02%	6.67%					
86.24			7.46%	7.45%			
94.08					7.51%	7.34%	
98	5.26%	5.00%					
107.8			4.82%	4.96%			
117.6					5.14%	5.08%	
117.6	5.26%	5.00%					
129.36			5.26%	5.67%			
141.12					5.53%	5.65%	
137.2	6.14%	5.83%					
150.92			5.70%	5.67%			
164.64					5.93%	6.21%	
156.8	6.14%	5.83%					
172.48			5.70%	6.03%			
188.16					5.93%	6.21%	
176.4	7.89%	7.50%					
194.04			7.89%	8.16%			
211.68					8.30%	8.47%	
196	12.28%	12.50%					

215.6			13.16%	13.48%		
235.2					13.44%	13.56%
215.6	23.68%	24.17%				
237.16			24.12%	23.76%		
258.72					23.72%	23.73%
235.2	17.54%	17.50%				
258.72			16.23%	15.60%		
282.24					15.42%	15.82%
254.8	5.26%	5.83%				
280.28			5.26%	4.96%		
305.76					4.74%	4.52%
274.4	0.88%	1.67%				
301.84			1.75%	1.77%		
329.28					1.58%	1.13%
294	0.88%	0.83%				
323.4			0.88%	0.71%		
352.8					0.79%	0.56%

Fuente: Elaboración propia

Tabla 30. Distribución porcentual par el eje tridem

CARGA EJE TRIDEM	TPDS						
	KN	0 - 500	500 - 1.000	1.000 - 2.500	2.500 - 5.000	5.000 - 10.000	Mayor a 10.000
58.8		0.00%	3.33%				
64.68				1.49%	2.63%		
70.56						1.56%	2.63%
78.4	4.35%	3.33%					
86.24			2.99%	3.95%			
94.08					3.13%	2.63%	
98	0.00%	3.33%					
107.8			1.49%	1.32%			
117.6					1.56%	2.63%	
117.6	0.00%	0.00%					
129.36			1.49%	1.32%			
141.12					1.56%	2.63%	
137.2	0.00%	0.00%					

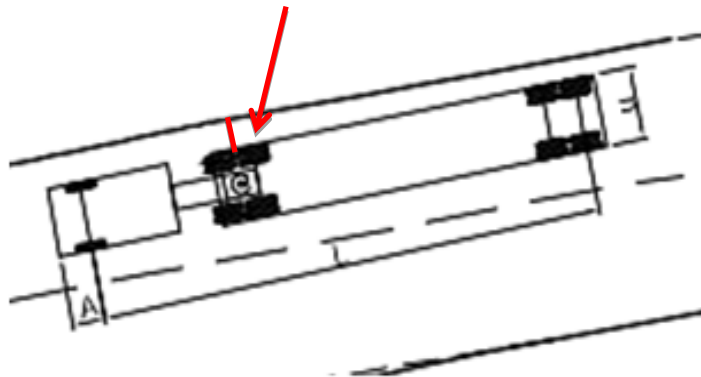
150.92			1.49%	1.32%		
164.64					1.56%	2.63%
156.8	4.35%	3.33%				
172.48			2.99%	2.63%		
188.16					3.13%	2.63%
176.4	4.35%	3.33%				
194.04			2.99%	2.63%		
211.68					3.13%	2.63%
196	4.35%	3.33%				
215.6			4.48%	3.95%		
235.2					4.69%	5.26%
215.6	4.35%	6.67%				
237.16			7.46%	6.58%		
258.72					6.25%	7.89%
235.2	17.39%	16.67%				
258.72			16.42%	17.11%		
282.24					17.19%	15.79%
254.8	21.74%	23.33%				
280.28			22.39%	22.37%		
305.76					21.88%	21.05%
274.4	13.04%	13.33%				
301.84			13.43%	13.16%		
329.28					12.50%	13.16%
294	8.70%	6.67%				
323.4			7.46%	7.89%		
352.8					7.81%	7.89%
313.6	4.35%	3.33%				
344.96			4.48%	3.95%		
376.32					4.69%	2.63%
333.2	4.35%	3.33%				
366.52			2.99%	3.95%		
399.84					3.13%	2.63%
352.8	4.35%	3.33%				
388.08			2.99%	2.63%		
423.36					3.13%	2.63%
372.4	4.35%	3.33%				
409.64			2.99%	2.63%		
446.88					3.13%	2.63%

Fuente: Elaboración propia

2.10 ENTRADAS GENERALES DE TRÁFICO

2.10.1 Ubicación de la rueda

Figura 6. Ubicación de la rueda



Fuente: Manual de diseño geométrico de carreteras. Capítulo 5. Instituto Nacional de Vías

Cada vehículo requiere un espacio lateral de seguridad, medido a partir de la cara interna que demarca el carril. A continuación se presenta este valor en función del ancho de calzada, para una calzada de ancho diferente se puede encontrar el valor por interpolación.

Tabla 31. Ubicación de la rueda en función del ancho de la calzada

	ANCHO DE CALZADA [m]		
	6.00	6.60	7.20
Ubicación de la rueda [m]	0.3	0.375	0.45

Fuente: Manual de diseño geométrico de carreteras. Capítulo 5. Instituto Nacional de Vías

2.10.2 Desviación de ubicación de rueda

Este valor representa la variabilidad en la ubicación de la rueda del vehículo, el cual se tomará por defecto igual a 0.2 m.

2.10.3 Ancho de calzada

El ancho de calzada es función de la categoría de la carretera, del tipo de terreno y de la velocidad de diseño del tramo homogéneo. El ancho mínimo para carreteras de una sola calzada es de seis metros (6 m) con el fin de permitir el cruce de dos vehículos de diseño que viajen en sentido contrario.

Tabla 32. Ancho de calzada (m)

CATEGORÍA DE LA CARRETERA	TIPO DE TERRENO	VELOCIDAD DE DISEÑO DEL TRAMO HOMOGÉNEO (VTR) (km/h)									
		20	30	40	50	60	70	80	90	100	110
Primaria de dos calzadas	Plano	-	-	-	-	-	-	7.3	7.3	7.3	7.3
	Ondulado	-	-	-	-	-	-	7.3	7.3	7.3	7.3
	Montañoso	-	-	-	-	-	7.3	7.3	7.3	7.3	-
	Escarpado	-	-	-	-	-	7.3	7.3	7.3	-	-
Primaria de una calzada	Plano	-	-	-	-	-	-	7.3	7.3	7.3	-
	Ondulado	-	-	-	-	-	7.3	7.3	7.3	7.3	-
	Montañoso	-	-	-	-	7.3	7.3	7.3	7.3	-	-
	Escarpado	-	-	-	-	7	7	7	-	-	-
Secundaria	Plano	-	-	-	-	7.3	7.3	7.3	-	-	-
	Ondulado	-	-	-	7	7.3	7.3	7.3	-	-	-
	Montañoso	-	-	6.6	7	7	7	-	-	-	-
	Escarpado	-	-	6	6.6	7	-	-	-	-	-
Terciaria	Plano	-	-	6	-	-	-	-	-	-	-
	Ondulado	-	6	6	-	-	-	-	-	-	-
	Montañoso	6	6	6	-	-	-	-	-	-	-
	Escarpado	6	6	-	-	-	-	-	-	-	-

Fuente: Manual de diseño geométrico de carreteras. Capítulo 5. Instituto Nacional de Vías

2.10.4 Número de ejes por clase de camión

Tabla 33. Número de ejes por clase de camión

CLASE	NÚMERO DE EJES		
	SIMPLE	TÁNDEM	TRIDEM
Bus	2	0	0
C2P	2	0	0
C2G	2	0	0
C3	1	1	0
C4	2	1	0
C5	1	2	0
>C5	1	1	1

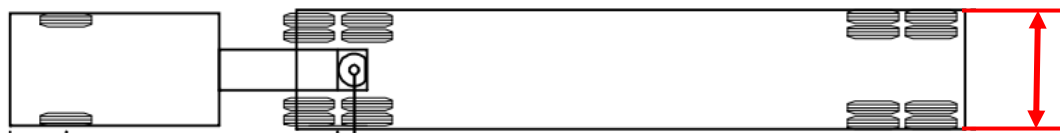
Fuente: Elaboración propia

2.10.5 Configuración del eje

2.10.5.1 Ancho del eje

Corresponde a la distancia entre extremos externos de las llantas del eje como se muestra en la siguiente figura:

Figura 7. Ancho del eje



Fuente: Manual de diseño geométrico de carreteras. Capítulo 5. Instituto Nacional de Vías

Los valores correspondientes al ancho del eje para cada clase de camión son:

Tabla 34. Ancho promedio del eje (m)

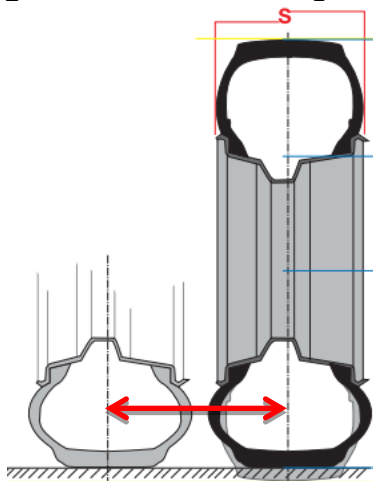
	CATEGORÍA	ANCHO PROMEDIO DEL EJE [m]
B	Bus mediado	2.44
B	Bus grande	2.6
C2	Camión de dos ejes	2.5
C3	Camión de tres ejes o dobletroque	2.5
C3S 2	Tractocamión de tres ejes con semirremolque de dos ejes	2.59

Fuente: Manual de diseño geométrico de carreteras. Capítulo 5. Instituto Nacional de Vías

2.10.5.2 Distancia mínima de gemelado

Según especificaciones técnicas de marcas de llantas para buses y camiones como Brigdestone, Pirelli y Michelin, la distancia centro a centro de llantas gemelas debe cumplir un valor mínimo para evitar el contacto en la zona de flexión.

Figura 8. Distancia de gemelado



Fuente: Technical Data-Pirelli Truck Tyres

Los valores promedio para diferentes marcas de llantas a partir de la distancia de gemelado promedio de acuerdo a su dimensión se muestran a continuación, el valor a utilizar en la metodología será el promedio total de las tres marcas analizadas, el cual es 30.3 cm.

Tabla 35. Distancia mínima de gemelado (mm)

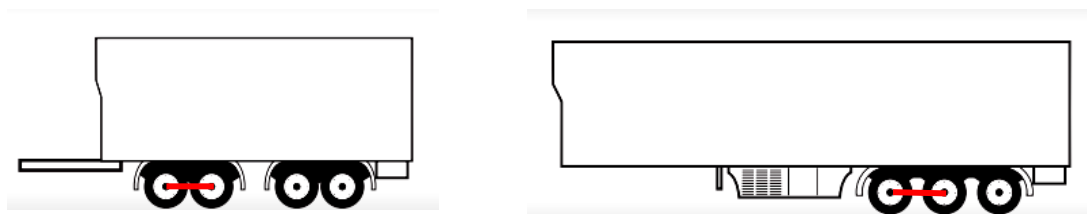
Dimensión	Bridgestone	Pirelli	Michelin	
17.5 "	260.8	248.9	256.2	
19.5"	286.6	306.4	298.09	
20"	-	323.3	-	
22.5"	302.8	329.2	328.2	
24"	-	370.3	353.5	
24.5"	330.2	-	-	
Promedio	283.4	315.6	309.0	<u>302.7</u>

Fuente: Elaboración propia

2.10.5.3 Espaciamiento entre ejes

Se refiere a la separación entre las líneas de rotación de un eje tándem o tridem. Este valor varía de acuerdo a las especificaciones del fabricante y de la clase de camión, sin embargo, se encuentra entre 1 y 1,6 m.

Figura 9. Espaciamiento entre ejes



Fuente:Tabla Cargas/Presiones- Pirelli

2.10.5.4 Presión de inflado

La presión adecuada de aire de los neumáticos se establece conjuntamente por el fabricante de estos y el constructor del vehículo, esta medida es importante para garantizar un rendimiento kilométrico óptimo, confort y una adecuada transmisión de esfuerzos de tracción y de frenada. A partir de especificaciones dadas por fabricantes de neumáticos de diferentes dimensiones para camiones se establece un valor promedio para la presión de inflado:

Figura 10. Presión de inflado [kPa]

Dimensión	Presión de inflado [kPa]
17.5 "	752.5
19.5"	811.5
22.5"	869.3
24"	850.0
Promedio	820.8

Fuente: Elaboración Propia

2.10.6 Batalla

El espaciamiento entre el eje direccional y el eje motor es usado en el análisis de pavimentos de concreto con juntas para determinar las cargas en las losas de concreto. Esta distancia está dada en las especificaciones de cada tipo de camión, en los siguientes rangos: corto, medio y largo.

Figura 11. Batalla

	Espaciamiento [m]
Corto	3.6-3.9
Medio	4.5-4.8
Largo	5.5-6.0

Fuente: Elaboración Propia

3. RECOPIACIÓN, PROCESAMIENTO Y ADAPTACIÓN DE LOS REQUERIMIENTOS DE LA GUIA AASHTO 2002 PARA LOS MATERIALES

3.1 CAPA ASFÁLTICA

3.1.1 MEZCLA ASFÁLTICA EN CALIENTE

Se define como la combinación de un ligante hidrocarbonado, agregados y eventualmente aditivos, de manera que todas las partículas del agregado queden recubiertas de manera homogénea por el ligante. El proceso de fabricación implica calentar el ligante y los agregados, y su utilización en obra debe realizarse a temperaturas superiores a la del ambiente.

Las especificaciones de construcción que deben tener sus componentes de acuerdo al tipo de mezcla son los siguientes:

3.1.1.1 Agregados pétreos y llenante mineral

La granulometría del agregado utilizado para mezclas asfálticas en caliente, derivado de la combinación de diferentes fracciones de agregado grueso (porción retenida en el tamiz de 4.75 mm (No.4)), agregado fino (porción comprendida entre los tamices de 4.75 mm y 75 μ m (No.4 y No.200)) y llenante mineral (pasa el tamiz No.200), debe estar comprendida dentro de las siguientes franjas. El análisis granulométrico se deberá efectuar de acuerdo con la norma INV E-213.

Tabla 36. Franjas granulométricas para mezclas asfálticas en caliente

TAMIZ		MEZCLA DENSA EN CALIENTE % RETENIDO								
		MDC-1			MDC-2			MDC-3		
U.S Standard	mm	Min	Max	Media	Min	Max	Media	Min	Max	Media
3/4"	19	5	20	12.5						
1/2"	12.5	10	13	11.5	5	20	12.5			

3/8"	9.5	7	8	7.5	7	10	8.5			
No.4	4.75	17	18	17.5	21	23	22	13	35	24
No.10	2	14	14	14	20	20	20	22	26	24
No.40	0.425	15	20	17.5	15	20	17.5	27	32	29.5
No.80	0.18	6	8	7	6	8	7	7	10	8.5
No.200	0.075	4	9	6.5	4	9	6.5	4	9	6.5
Fondo		4	8	6	4	8	6	5	10	7.5

Fuente: Especificaciones Generales de Construcción de Carreteras. . Mezclas asfálticas en caliente. Art 450-07. Instituto Nacional de Vías

3.1.1.2 Material Bituminoso

El material bituminoso para elaborar la mezcla en caliente se selecciona en función de las características climáticas de la zona y las condiciones de operación de la vía. Existen dos tipos de material bituminoso: cemento asfáltico convencional, y cemento asfáltico modificado con polímeros.

➤ Cemento asfáltico convencional

Las especificaciones que debe cumplir el cemento asfáltico convencional son las siguientes:

Tabla 37. Especificaciones del cemento asfáltico

CARACTERÍSTICA	UNIDADES	NORMA DE ENSAYO O INV	GRADO DE PENETRACIÓN			
			60-70		80-100	
			Min	Max	Min	Max
Penetración (25°C, 100 g, 5s)	0.1 mm	E-706	60	70	80	100
Índice de penetración	-	E-724	-1	+1	-1	+1
Viscosidad absoluta (60°C)	P	E-716 o E-717	1500	-	1000	-
Ductilidad (25°C, 5 cm/min)	cm	E-702	100	-	100	-
Solubilidad en tricloroetileno	%	E-713	99	-	99	-
Contenido de agua	%	E-704	-	0.2	-	0.2
Punto de ignición mediante copa abierta de Cleveland	°C	E-709	230	-	230	-

Pérdida de masa por calentamiento en película delgada en movimiento (163°C, 75 mins)	%	E-720	-	1	-	1
Penetración del residuo luego de la pérdida por calentamiento (E-720), en % de la penetración original	%	E-706	52	-	48	-
Incremento en el punto de ablandamiento luego de la pérdida por calentamiento en película delgada en movimiento (E-720)	°C	E-712	-	9	-	9

Fuente: Especificaciones Generales de Construcción de Carreteras. Art 400-07.
Instituto Nacional de Vías

➤ **Cemento asfáltico modificado con polímeros**

El cemento asfáltico modificado con polímeros se define como el ligante hidrocarbonado resultado de la interacción física y/o química de polímeros con un cemento asfáltico convencional. Las especificaciones requeridas son las siguientes:

Tabla 38. Especificaciones de cementos asfálticos modificados con polímeros

CARACTERÍSTICA	UNIDAD	NORMA DE ENSAYO INV	TIPO I		TIPO II		TIPO III		TIPO IV		TIPO V	
			Mi n	Ma x	Mi n	Ma x	Mi n	Ma x	Mi n	Ma x	Mi n	Ma x
Asfalto original												
Penetración (25°C, 100 g, 5s)	0.1 mm	E-706	55	70	55	70	55	70	80	130	15	40
Punto de ablandamiento con aparato de anillo y bola	°C	E-712	58	-	58	-	65	-	60	-	65	-
Ductilidad (5°C, 5 cm/min)	cm	E-702	-	-	15	-	15	-	30	-	-	-
Recuperación elástica por torsión a 25°C	%	E-727	15	-	40	-	70	-	70	-	15	-
Estabilidad al almacenamiento Diferencia en el punto de ablandamiento	°C	E-726 y E-712	-	5	-	5	-	5	-	5	-	5
Contenido de agua	%	E-704	-	0.2	-	0.2	-	0.2	0.2	-	-	0.2

Punto de ignición mediante la copa abierta Cleveland	°C	E-709	23 0	-	23 0	-	23 0	-	23 0	-	23 0	-
Residuo del ensayo de pérdida por calentamiento en película delgada en movimiento (INV E-720)												
Pérdida de masa	%	E-720		1	-	1	-	1	-	1	-	1
Penetración del residuo luego de la pérdida por calentamiento en película delgada en movimiento, % de la penetración original	%	E-706	65	-	65	-	65	-	60	-	70	-
Ductilidad (5°C, 5 cm/min)	cm	E-702	-	-	8	-	8	-	15	-	-	-

Fuente: Especificaciones Generales de Construcción de Carreteras. Art 400-07.
Instituto Nacional de Vías

3.1.1.3 Diseño de la mezcla

El diseño de la mezcla se realiza en base al método Marshall de acuerdo a los siguientes criterios:

Tabla 39. Criterios de diseño de la mezcla asfáltica en caliente por el método Marshall

CARACTERÍSTICA	NORMA DE ENSAYO INV	MEZCLAS DENSAS, SEMIDENSAS Y GRUESAS			
		Min	Max	Media	
Compactación (golpes/cara)	E-748		75	75	
Estabilidad mínima (kg)	E-748		750	750	
Flujo (mm)	E-748	2	4	3	
Vacíos con aire (Va)*, %	E-736 o E-799	3	5	4	
Vacíos en los agregados minerales (VAM) , %	MDC-1	E-799	14	-	14
	MDC-2		15	-	15
	MDC-3		16	-	16
% de vacíos llenos de asfalto (VFA) (Volumen de asfalto efectivo/ Vacíos en los agregados minerales) x100	E-799	65	75	70	

Relación llenante/Asfalto efectivo, en peso	E-799	0.8	1.2	1
Concentración de llenante, valor máximo	E-745	Valor crítico		

Fuente: Especificaciones Generales de Construcción de Carreteras. Mezclas asfálticas en caliente. Art 450-07. Instituto Nacional de Vías

3.2 PAVIMENTO DE CONCRETO HIDRÁULICO

3.2.1 Cemento

El cemento a utilizar será Portland, de tipo I a menos que el proyecto a realizar demande lo contrario. Sin embargo, el cemento Portland se clasifica en ocho (8) tipos de acuerdo a sus propiedades:

- **Tipo 1**→ Utilizado en obras de concreto hidráulico en general, al que no se le exigen propiedades especiales.
- **Tipo 1A**→ Cemento con agente para incorporar aire, destinado para los mismo usos del cemento Tipo 1, donde se requiere aire incluido.
- **Tipo 2**→ Utilizado en obras de concreto hidráulico expuestas a la acción moderada de sulfatos y a obras en las que se requiera calor de hidratación moderado.
- **Tipo 2A**→ Cemento con agente para incorporar aire, destinado para los mismo usos del cemento Tipo 2, donde se requiere aire incluido.
- **Tipo 3**→ Desarrolla altas resistencias iniciales.
- **Tipo 3A**→ Cemento con agente para incorporar aire, destinado para los mismo usos del cemento Tipo 3, donde se requiere aire incluido.
- **Tipo 4**→ Desarrolla bajo calor de hidratación.
- **Tipo 5**→ Cemento que ofrece alta resistencia a la acción de los sulfatos.¹²

¹² COLOMBIA. MINISTERIO DE TRANSPORTE. Especificaciones generales de Construcción. Artículo 501-07. Suministro de cemento portland normal. Bogotá: INVIAS, 2007.

El cemento Portland debe cumplir con los requisitos físicos mostrados en la siguiente tabla:

Tabla 40. Requisitos físicos del cemento Portland

ENSAYO	TIPO							
	1	1A	2	2A	3	3A	4	5
Contenido de aire del mortero, % en volumen Norma INV E-328								
Máximo	12	22	12	22	12	22	12	22
Mínimo	-	-	-	-	-	-	-	-
Finura Método del turbidímetro (m²/Kg). Norma INV E-303								
Valor medio, mínimo	160	160	160	160	-	-	160	160
Valor mínimo, un solo resultado	150	150	150	150	-	-	150	150
Valor medio, máximo	220	220	220	220	-	-	220	220
Valor máximo, un solo resultado	230	230	230	230	-	-	230	230
Ensayo de permeabilidad al aire Norma INV E-302								
Valor medio, mínimo	280	280	280	280	-	-	280	280
Valor mínimo, un solo resultado	260	260	260	260	-	-	260	260
Valor medio, máximo	400	400	400	400	-	-	400	400
Valor máximo, un solo resultado	420	420	420	420	-	-	420	420
Expansión en autoclave, % máximo Norma INV E-304	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8
Resistencia a la compresión Mpa/(psi) Norma INV E-323								
1 día	-	-	-	-	12 (1740)	10 (1450)	-	-
3 días	12 (1740)	10 (1450)	10 (1450)	8 (1160)	24 (3480)	19 (2760)	-	8.0 (1160)
7 días	19 (2760)	16 (2320)	17 (2470)	14 (2030)	-	-	7.0 (1020)	15 (2180)

28 días	-	-	-	-	-	-	17 (2470)	21 (3050)
Tiempo de fraguado Gillmore Norma INV E-306								
Tiempo inicial, en minutos valor mínimo	60	60	60	60	60	60	60	60
Tiempo final, en minutos valor máximo	600	600	600	600	600	600	600	600
Tiempo de fraguado Vicat Norma INV E-305								
Tiempo inicial, en minutos valor mínimo	45	45	45	45	45	45	45	45
Tiempo final, en minutos valor máximo	375	375	375	375	375	375	375	375
Falso fraguado, penetración final, % mínimo Norma INV 3-308	50	50	50	50	50	50	50	50
Calor de hidratación Norme INV E-309								
7 días, máximo Kj/Kg (cal/g)	-	-	290 (70)	290 (70)	-	-	250 (60)	-
28 días, máximo Kj/Kg (cal/g)	-	-	-	-	-	-	290 (70)	-
Resistencia a la compresión 28 días Mpa(psi) Norma INV E-323	28 (4060)	22 (3190)	28 (4060)	22 (3190)	-	-	-	-
Resistencia a los sulfatos, 14 días, % máximo de expansión Norma INV E-329	-	-	-	-	-	-	-	0.04

Fuente: Especificaciones Generales de Construcción de Carreteras. Suministro de cemento portland normal. Art 501-07. Instituto Nacional de Vías

3.2.2 Agregado fino

Se considera agregado fino a la fracción que pasa el tamiz de 4.75 mm (No. 4), y debe cumplir la siguiente granulometría:

Tabla 41. Granulometría para el agregado fino para pavimentos de concreto hidráulico

TAMIZ		PORCENTAJE QUE PASA
Normal	Alterno	
9.5 mm	3/8"	100
4.75 mm	No.4	95-100
2.36 mm	No.8	80-100
1.18 mm	No.16	50-85
600 µm	No.30	25-60
300 µm	No.50	10-30
150 µm	No.100	2-10
75 µm	No.200	3

Fuente: Especificaciones Generales de Construcción de Carreteras. Pavimento de concreto hidráulico. Art 500-07. Instituto Nacional de Vías

3.2.3 Agregado grueso

Se define como la porción de agregado retenida en el tamiz de 4.75 mm (No.4) y se debe ajustar a algunas de las siguientes granulometrías:

Tabla 42. Granulometrías para el agregado grueso para pavimentos de concreto hidráulico

TAMIZ		PORCENTAJE QUE PASA				
		AG1		AG2		AG3
Normal (mm)	Alterno	2"-1"	1"-No.4	1 1/2"-3/4"	3/4"-No.4	1"-No.4
63	2 1/2"	100	-	-	-	-
50	2"	90-100	-	100	-	-
37.5	1 1/2"	35-70	100	90-100	-	100
25	1"	0-15	95-100	20-55	100	95-100
19	3/4"	-	-	0-15	90-100	-
12.5	1/2"	0-5	25-60	-	-	25-60
9.5	3/8"	-	-	0-5	20-55	-
4.75	No.4	-	0-10	-	0-10	0-10
2.36	No.8	-	0-5	-	0-5	0-5

Fuente: Especificaciones Generales de Construcción de Carreteras. Pavimento de concreto hidráulico. Art 500-07. Instituto Nacional de Vías

3.2.4 Métodos de Curado¹³

La selección del tipo de curado y el momento de su aplicación depende de las condiciones ambientales y el tipo de mezcla. El curado del concreto se podrá llevar a cabo por:

- Curado por humedad
- Curado con productos químicos que forman película impermeable
- Curado mediante membranas de polietileno o de papel

3.2.5 Módulo de rotura

Cuando se cuente con curvas que relacionen la resistencia a la compresión con la resistencia a la flexión se podrán usar los resultados de las pruebas de compresión. A continuación se muestran cuatro niveles de resistencia a la flexión de acuerdo con el nivel de tránsito:

Tabla 43. Resistencia a flexión del concreto

Calidad del concreto	Número de camiones por día			
	>300	150-300	25-150	<25
A flexión (Mpa)	4.5	4.2	4	3.8

Fuente: Manual de diseño de pavimentos de concreto para vías con bajos, medios y altos volúmenes de tránsito. Instituto Nacional de Vías

¹³ COLOMBIA. MINISTERIO DE TRANSPORTE. Especificaciones generales de Construcción. Artículo 500-07. Pavimento de concreto hidráulico . Bogotá: INVÍAS, 2007.

3.2.6 Módulo de Elasticidad

La determinación del módulo de elasticidad se hace a partir del método de ensayo descrito en la norma INV E-416-07, sin embargo, si no se cuenta con ensayos es posible utilizar las correlaciones mostradas a continuación:

Tabla 44. Correlación entre la resistencia a la compresión y el módulo de elasticidad

Tipo de agregado y origen	Módulo de elasticidad E_c (Mpa·kg/cm ²)
Grueso-Ígneo	$E_c=5500 \cdot \sqrt{f'_c}-17500 \cdot \sqrt{f'_c}$
Grueso-Metamórfico	$E_c=4700 \cdot \sqrt{f'_c}-15000 \cdot \sqrt{f'_c}$
Grueso-Sedimentario	$E_c=3600 \cdot \sqrt{f'_c}-11500 \cdot \sqrt{f'_c}$
Sin información	$E_c=3900 \cdot \sqrt{f'_c}-12500 \cdot \sqrt{f'_c}$

Fuente: Manual de diseño de pavimentos de concreto para vías con bajos, medios y altos volúmenes de tránsito. Instituto Nacional de Vías

3.3 CLASIFICACIÓN DE SUELOS

Las dos clasificaciones más usuales para la clasificación de suelos y agregados para la construcción de carreteras en función del tamaño de las partículas son la del American Association of State Highway and Transportation Officials (AASHTO) y el sistema de clasificación de suelos unificado, propuesto por la ASTM. Las denominaciones utilizadas son las siguientes:

Tabla 45. Límites de tamaño de partículas

Sistema de Clasificación	Tamaño de grano
Unificado	Gravas: 75 mm a 4.75 mm
	Arenas: 4.75 mm a 0.075 mm
	Finos < 0.075 mm
AASHTO	Gravas: 75mm a 2mm
	Arenas: 2 mm a 0.05 mm

	Limos: 0.05 mm a 0.002 mm
	Arcillas <0.002 mm

Fuente: Manual de diseño de pavimentos de concreto para vías con bajos, medios y altos volúmenes de tránsito. Instituto Nacional de Vías

3.3.1 Sistema de clasificación de suelos según la AASHTO

En este sistema se clasifica el suelo en ocho grupos de A-1 al A-8 basados en la distribución del tamaño de grano, límite líquido e índice plástico. Los grupos A-1, A-2 y A-3 se refieren a los suelos gruesos granulares, mientras que los grupos A-4, A-5, A-6 y A-7 denotan los suelos finos granulares, las turbas y suelos orgánicos se clasifican bajo el grupo A-8.

Tabla 46. Sistema de clasificación de suelos según la AASHTO

Clasificación general	Materiales granulares (35% o menos del total de la muestra para el tamiz No.200)							Materiales limosos y arcillosos (Más del 35% del total de la muestra pasa el tamiz No.200)				
	A-1		A-3	A-2				A-4	A-5	A-6	A-7	
Grupo de clasificación	A-1-a	A-1-b		A-2-4	A-2-5	A-2-6	A-2-7				A-7-5a	A-7-5b
Análisis de tamizado % que pasa												Si $IP \leq LL-30$, entonces es A-7-5-a Si $IP > LL-30$, entonces es A-7-5-b
Tamiz No.10	50 max											
Tamiz No.40	30 max	50 max	51 max									
Tamiz No.200	15 max	25 max	10 max	35 max	35 max	35 max	35 max	36 min	36 min	36 min	36 min	36 min
Para fracciones que pasan el tamiz No. 40												
Límite líquido (LL)				40 max	41 max	40 max	41 min	40 max	41 min	40 max	41 min	41 min
Límite plástico (LP)	6 max		No plástico	10 max	10 max	11 min	11 min	10 max	10 max	11 min	11 min	11 min
Tipo de material	Fragmentos de roca, gravas y arenas		Arena fina	Grava limosa o arcillosa y arenas				Suelos limosos			Suelos arcillosos	
Calificación	De excelente a buena							De buena a pobre				

Fuente: Mecánica de suelos: fundamentos de la mecánica de suelos¹⁴

¹⁴ RICO RODRIGUEZ, Alfonso. JUAREZ BADILLO, Eulalio. Mecánica de suelos: fundamentos de la mecánica de suelos. México: Limusa, 1980.

3.3.2 Sistema unificado de clasificación de suelos

El procedimiento para la clasificación de suelos por el sistema Unificado utiliza una tabla de criterios de escogencia junto con la carta de plasticidad de Casagrande, en la cual se definen cinco sectores a partir de tres rectas de ecuaciones:

$IP=0.73(LL-20)$ -> Línea A

$LL=20$

$IP=50$

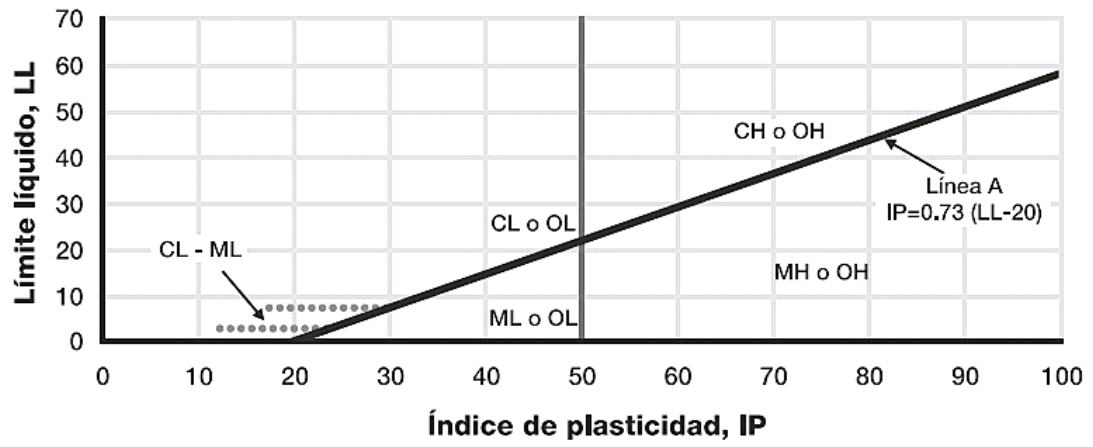
La carta de plasticidad se muestra en la siguiente figura. Para la identificación de los suelos se usan los símbolos mostrados en la siguiente tabla:

Tabla 47. Símbolos usados en el sistema Unificado de clasificación de suelos

Descripción	Símbolo
Gravas	G
Arenas	S
Limos	M
Arcillas	C
Limos y arcillas orgánicas	O
Turba y suelos orgánicos	Pt
Alta plasticidad o alta compresibilidad	H
Baja plasticidad o baja compresibilidad	L
Bien gradado	W
Pobrementemente gradado	Pt

Fuente: Manual de diseño de pavimentos de concreto para vías con bajos, medios y altos volúmenes de tránsito. Instituto Nacional de Vías

Figura 12. Carta de plasticidad de Casagrande



Fuente: Mecánica de suelos: fundamentos de la mecánica de suelos

Tabla 48. Clasificación unificada de suelos

Divisiones	Criterio	Símbolo
Suelos Gruesos Granulares	$F_{200} < 5, C_u \geq 4, 1 \leq C_C \leq 3$	GW
$R_{200} > 50$	$F_{200} < 5, C_u < 4$, y/o C_C no se encuentra entre 1-3	GP
Gravas $R_4 > 0.5R_{200}$	$F_{200} > 12, IP < 4$, o se localiza debajo de la línea A en la carta de plasticidad	GM
	$F_{200} > 12, IP > 7$, o se localiza en o sobre la línea A en la carta de plasticidad	GC
	$F_{200} > 12, LL < 50, 4 \leq IP \leq 7$, o se localiza en o sobre la línea A en la carta de plasticidad	GC-GM*
	$5 \leq F_{200} \leq 12$; reúne el criterio de gradación de GW y el criterio de plasticidad de GM	GW-GM*
	$5 \leq F_{200} \leq 12$; reúne el criterio de gradación de GW y el criterio de plasticidad de GC	GW-GC*
	$5 \leq F_{200} \leq 12$; reúne el criterio de gradación de GP y el criterio de plasticidad de GM	GP-GM*
	$5 \leq F_{200} \leq 12$; reúne el criterio de gradación de GP y el criterio de plasticidad de GC	GP-GC*
Arenas $R_4 \leq 0.5R_{200}$	$F_{200} < 5, C_u \geq 6, 1 \leq C_C \leq 3$	SW
	$F_{200} < 5, C_u < 6$, y/o C_C no se encuentra entre 1-3	SP
	$F_{200} > 12, IP < 4$, o se localiza debajo de la línea A en la carta de plasticidad	SM
	$F_{200} > 12, IP > 7$, o se localiza en o sobre la línea A en la carta de plasticidad	SC
	$F_{200} > 12, LL > 50, 4 \leq IP \leq 7$, o se localiza en o sobre la línea A en la carta de plasticidad	SC-SM*
	$5 \leq F_{200} \leq 12$; reúne el criterio de gradación de SW y el criterio de plasticidad de SM	SW-SM*
	$5 \leq F_{200} \leq 12$; reúne el criterio de gradación de SW y el criterio de plasticidad de SC	SW-SC*
	$5 \leq F_{200} \leq 12$; reúne el criterio de gradación de SP y el criterio de plasticidad de SM	SP-SM*

	$5 \leq F_{200} \leq 12$; reúne el criterio de gradación de SP y el criterio de plasticidad de SC	SP-SC*
Suelo fino granular inorgánico	IP < 4, o se localiza debajo de la línea A en la carta de plasticidad	ML
$R_{200} \leq 50$	IP > 7, o se localiza en o sobre la línea A en la carta de plasticidad	CL
Limos y arcillas LL < 50	$4 \leq IP \leq 7$ o se localiza sobre la línea A en la carta de plasticidad	CL-ML*
	Se localiza bajo la línea A en la carta de plasticidad	MH
Limos y arcillas LL \geq 50	Se localiza en o sobre la línea A en la carta de plasticidad	CH
Suelo fino granular orgánico		
Limo y arcilla orgánica LL < 50	$LL_{\text{Sin secar en horno}} / LL_{\text{Seco en horno}} < 0.75$	OL
Limo y arcilla orgánica LL > 50	$LL_{\text{Sin secar en horno}} / LL_{\text{Seco en horno}} < 0.75$	OH
<p>Nota: F_{200} = porcentaje finos retenidos tamiz No. 200; R_{200} = porcentaje finos retenidos sobre tamiz No.200; R_4 = porcentaje finos retenidos sobre tamiz No.4; Cu = Coeficiente de uniformidad; Cc = Coeficiente de curvatura; LL= límite líquido; IP = índice de plasticidad; Límites de Atterberg basados en la fracción que pasa el tamiz No. 40.</p> <p>*Condiciones de frontera, clasificación dual.</p>		

Fuente: Manual de diseño de pavimentos de concreto para vías con bajos, medios y altos volúmenes de tránsito.
Instituto Nacional de Vías

3.4 BASE GRANULAR

Se denomina base granular a la capa granular localizada entre la subbase granular y la capa asfáltica en los pavimentos asfálticos. Los agregados para la construcción de la base granular deben satisfacer los siguientes requisitos:

Tabla 49. Requisitos de los agregados para bases granulares

ENSAYO	NORMA DE ENSAYO INV	TRÁNSITO		
		BAJO	MEDIO	ALTO
Dureza				
Desgaste en la máquina de los ángeles (Gradación A)	E-218			
- En seco, 500 revoluciones (%)		≤ 40	≤ 40	≤ 35
- En seco, 100 revoluciones (%)		≤ 8	≤ 8	≤ 7
- Después de 48 horas de inmersión, 500 revoluciones (%)		≤ 55	≤ 55	≤ 50
- Relación húmedo/seco, 500 revoluciones		≤ 2	≤ 2	≤ 2
Desgaste en el equipo Micro-Deval (%)	E-238	-	≤ 30	≤ 25
Evaluación de la resistencia mecánica por el método del 10% de finos	E-224			
- Valor en seco (KN)		-	≥ 70	≥ 90
- Relación húmedo/seco (%)		-	≥ 75	≥ 75
Contenido de terrones de arcilla y partículas deleznales (%)	E-211	≤ 2	≤ 2	≤ 2
Durabilidad				
Pérdidas en el ensayo de solidez en sulfatos	E-220			
- Sulfato de sodio (%)		≤ 12	≤ 12	≤ 12
- Sulfato de magnesio (%)		≤ 18	≤ 18	≤ 18
Limpieza				
Límite líquido (%)	E-125	≤40	-	-
Índice de plasticidad (%)	E-126	≤3	0	0
Equivalente de arena (%)	E-133	≥30	≥30	≥30
Geometría de las partículas				
Índices de alargamiento y aplanamiento (%)	E-230	≤ 35	≤ 35	≤ 35
Porcentaje de caras fracturadas (una cara)	E-227	≥ 50	≥ 50	≥ 60
Angularidad de la fracción fina (%)	E-239	-	≥ 35	≥ 35
Resistencia del material				

CBR (%) Nota: Porcentaje asociado al valor mínimo especificado de la densidad seca, medido en una muestra sometida a cuatro días de inmersión. Método D.	E-148	≥ 80	≥ 80	≥ 100

Fuente: Especificaciones Generales de Construcción de Carreteras. Disposiciones generales para la ejecución de afirmados, subbases granulares y bases granulares y estabilizadas. Art 300-07. Instituto Nacional de Vías

Los agregados se deben ajustar a alguna de las franjas granulométricas que se indican a continuación, la relación entre el porcentaje que pasa el tamiz de 75 µm (No.200) y el porcentaje que pasa el tamiz de 425 µm (No.40), no deberá exceder de 2/3 y el tamaño máximo nominal no deberá exceder de 1/3 del espesor de la capa compactada.

Tabla 50. Granulometría para base granular

TAMIZ		PORCENTAJE QUE PASA			
		BG-1		BG-2	
Normal	Alternativo	Rango	Media	Rango	Media
37.5 mm	1 1/2"	100	100	-	-
25 mm	1"	70-100	85	100	100
19 mm	3/4"	60-90	75	70-100	85
9.5 mm	3/8"	45-75	60	50-80	65
4.75 mm	No.4	30-60	45	35-65	50
2 mm	No.10	20-45	32.5	20-45	32.5
425 µm	No.40	10-30	20	10-30	20
75 µm	No.200	5-15	10	5-15	10

Fuente: Especificaciones Generales de Construcción de Carreteras. Base Granular. Art 330-07. Instituto Nacional de Vías

3.5 SUBBASE GRANULAR

Se denomina subbase granular a la capa granular localizada entre la subrasante y la base granular en los pavimentos asfálticos o a la que sirve de soporte a los pavimentos de concreto hidráulico. Los agregados para la construcción de la subbase granular deben satisfacer los siguientes requisitos:

Tabla 51. Requisitos de los agregados para subbases granulares

ENSAYO	NORMA DE ENSAYO INV	TRÁNSITO		
		BAJO	MEDIO	ALTO
Dureza				
Desgaste en la máquina de los ángeles (Gradación A)	E-218			
- En seco, 500 revoluciones (%)		≤ 50	≤ 50	≤ 50
- En seco, 100 revoluciones (%)		-	-	-
- Después de 48 horas de inmersión, 500 revoluciones (%)		-	-	-
- Relación húmedo/seco, 500 revoluciones		-	-	-
Desgaste en el equipo Micro-Deval (%)	E-238	-	≤ 35	≤ 30
Evaluación de la resistencia mecánica por el método del 10% de finos	E-224			
- Valor en seco (KN)		-	-	-
- Relación húmedo/seco (%)		-	-	-
Contenido de terrones de arcilla y partículas deleznablez (%)	E-211	≤ 2	≤ 2	≤ 2
Durabilidad				
Pérdidas en el ensayo de solidez en sulfatos	E-220			
- Sulfato de sodio (%)		≤ 12	≤ 12	≤ 12
- Sulfato de magnesio (%)		≤ 18	≤ 18	≤ 18
Limpieza				
Límite líquido (%)	E-125	≤40	≤40	≤40
Índice de plasticidad (%)	E-126	≤6	≤6	≤6
Equivalente de arena (%)	E-133	≥25	≥25	≥25
Geometría de las partículas				
Índices de alargamiento y aplanamiento (%)	E-230	-	-	-
Porcentaje de caras fracturadas (una cara)	E-227	-	-	-
Angularidad de la fracción fina (%)	E-239	-	-	-

Resistencia del material				
CBR (%) Nota: Porcentaje asociado al valor mínimo especificado de la densidad seca, medido en una muestra sometida a cuatro días de inmersión. Método D.	E-148	≥ 30	≥ 30	≥ 30

Fuente: Especificaciones Generales de Construcción de Carreteras. Disposiciones generales para la ejecución de afirmados, subbases granulares y bases granulares y estabilizadas. Art 300-07. Instituto Nacional de Vías

Los agregados se deben ajustar a alguna de las franjas granulométricas que se indican a continuación, la relación entre el porcentaje que pasa el tamiz de 75 µm (No.200) y el porcentaje que pasa el tamiz de 425 µm (No.40), no deberá exceder de 2/3 y el tamaño máximo nominal no deberá exceder de 1/3 del espesor de la capa compactada.

Tabla 52. Granulometría para subbase granular

TAMIZ		PORCENTAJE QUE PASA			
		SBG-1		SBG-2	
Normal	Alternativo	Rango	Media	Rango	Media
50 mm	2"	100	100	-	-
37.5 mm	1 1/2"	70-95	82.5	100	100
25 mm	1"	60-90	75	75-95	85
12.5 mm	1/2"	45-75	60	55-85	70
9.5 mm	3/8"	40-70	55	45-75	60
4.75 mm	No.4	25-55	40	30-60	45
2 mm	No.10	15-40	27.5	20-45	32.5
425 µm	No.40	6-25	15.5	8-30	19
75 µm	No.200	2-15	8.5	2-15	8.5

Fuente: Especificaciones Generales de Construcción de Carreteras. Subbase Granular. Art 320-07. Instituto Nacional de Vías

3.6 SUBRASANTE

El suelo es un material en el que se encuentran mezclas que pueden ser homogéneas, con unos pocos minerales, hasta heterogéneas; con granos de diferentes tamaños como fragmentos de roca, gravas, arenas, arcillas y limos.

Se debe establecer la resistencia del suelo como material de fundación, para determinar la capacidad de soporte de este. Existen correlaciones que permiten determinar los valores de la resistencia de la subrasante en función del CBR, el cual a su vez puede ser determinado mediante otros parámetros. A continuación se presentan algunas correlaciones¹⁵:

- **Correlación para suelos granulares limpios, típicamente no plásticos, en donde el índice de plasticidad ponderado ($w_xIP = P_{200} \cdot IP$) es igual a cero.**

$$CBR = 28.09x(D_{60})^{0.358} \quad \text{si } 0.01 \text{ mm} < D_{60} \leq 30 \text{ mm}$$

$$CBR=5\% \text{ si } D_{60} < 0.01 \text{ mm}$$

$$CBR=95\% \text{ si } D_{60} > 30 \text{ mm}$$

CBR: en porcentaje

D_{60} : Abertura del tamiz que deja pasar el sesenta por ciento (60%) del suelo, en peso. [mm]

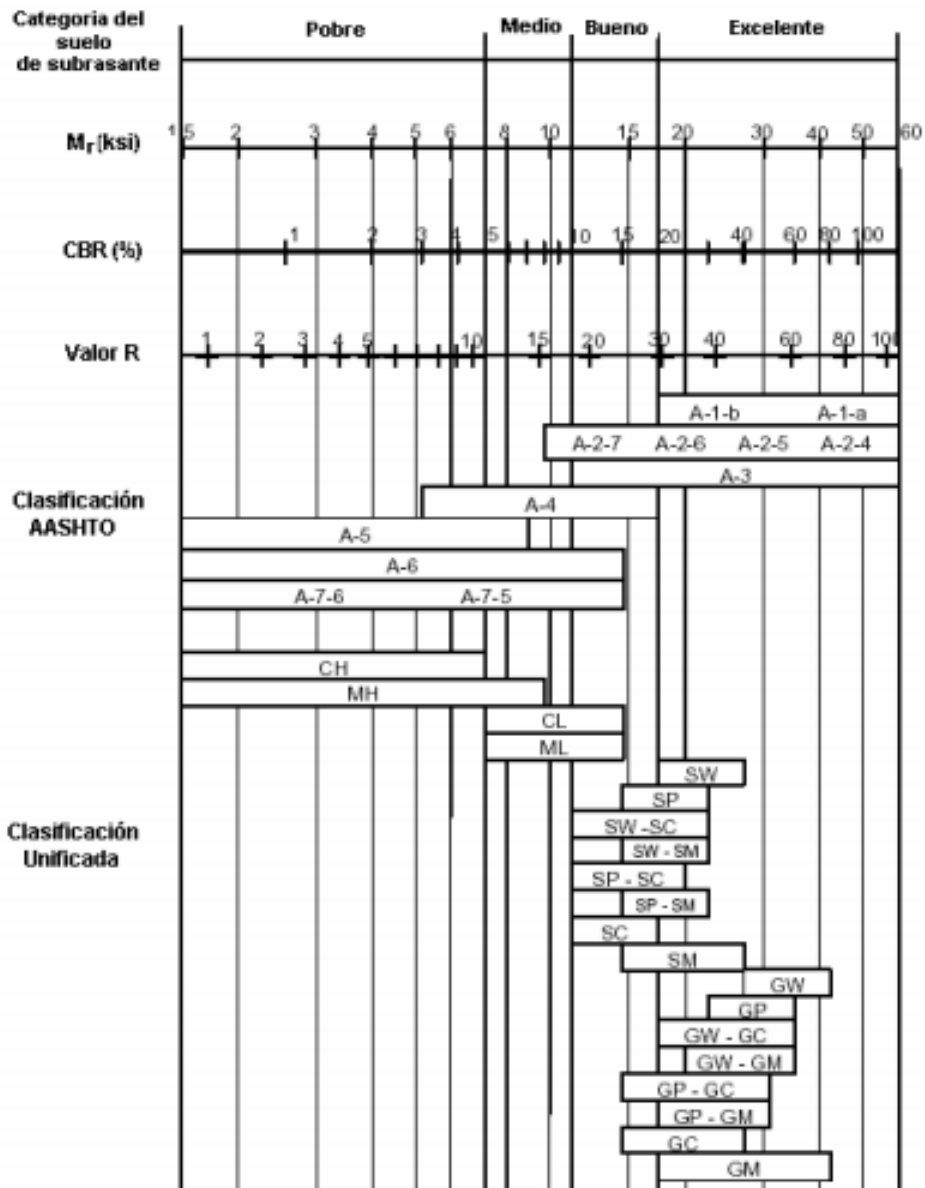
- **Correlación para suelos que presentan plasticidad ($w_xIP > 0$ y $P_{200} > 11\%$)**

$$CBR = \frac{75}{1 + 0.728(w_xIP)}$$

¹⁵ COLOMBIA. MINISTERIO DE TRANSPORTE. Manual de diseño de pavimentos asfálticos en vías con bajos volúmenes de tránsito. Bogotá: INVIAS, 2007. p. 67.

- Correlación general de la AASHTO entre la clasificación general y su probable valor de CBR:

Figura 13. Correlación AASHTO. Clasificación del suelo vs CBR y MR



Fuente: Manual de diseño de pavimentos asfálticos para vías con bajos volúmenes de tránsito. Instituto Nacional de Vías

➤ **Cálculo del módulo Resiliente:**

$$Mr = 2555x(CBR)^{0.64}$$

Mr: Valor del módulo resiliente [psi]

CBR: %

Además, en función del CBR se han definido las siguientes categorías que indican el comportamiento de la subrasante:

Tabla 53. Categorías de subrasante

CATEGORÍA	CBR (%)	COMPORTAMIENTO COM SUBRASANTE
S1	$CBR \leq 3$	Malo
S2	$3 < CBR \leq 5$	Regular
S3	$5 < CBR \leq 10$	Bueno
S4	$CBR > 10$	Muy Bueno

Fuente: Manual de diseño de pavimentos asfálticos para vías con bajos volúmenes de tránsito. Instituto Nacional de Vías

4. MANUAL DEL SOFTWARE SOPORTE PARA EL DESARROLLO DE LA METODOLOGÍA AASHTO 2002

4.1 CÓMO USAR ESTE MANUAL

Este manual tiene como propósito servir de ayuda para la implementación de la metodología AASHTO 2002, presentando una interfaz gráfica que compila los parámetros de entrada de esta metodología relacionados con el tráfico y la estructura, además pretende servir de base a futuras investigaciones que tengan como fin la aplicación y calibración del método.

4.2 INSTALACIÓN DEL SOFTWARE

Para utilizar este software es necesario en primera instancia instalar el ejecutable dando doble clic al instalador, esta acción puede trasladarlo a la siguiente dirección de internet:

* <http://www.microsoft.com/download/en/details.aspx?displayland=en&id=17113>

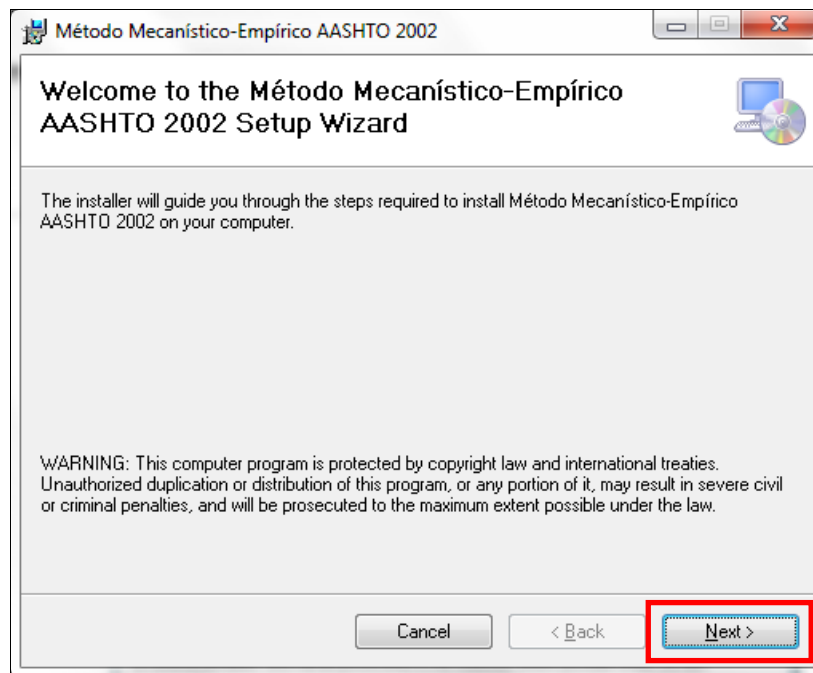
Figura 14. Microsoft.NET Framework 4 ClientProfile



Fuente: Microsoft Download Center

Si fue enviado a esta página o no puede instalar el programa, debe descargar primero el “Microsoft.NET Framework 4 ClientProfile (Web Installer)” e instalarlo para posteriormente poder instalar el software. Una vez el asistente para la instalación haya comenzado podrá ver la siguiente ventana en la cual debe seleccionar el botón Next.

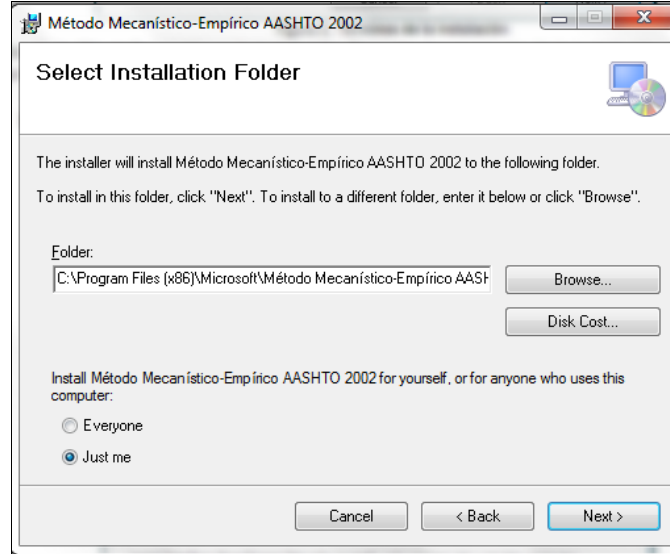
Figura 15. Asistente de instalación-Paso1



Fuente: Instalador software

Al continuar con la instalación, se pedirá la dirección en la que se ubicará la carpeta raíz de la interfaz, por defecto saldrá una dirección, sin embargo, es posible cambiarla manualmente mediante el botón “Browser”.

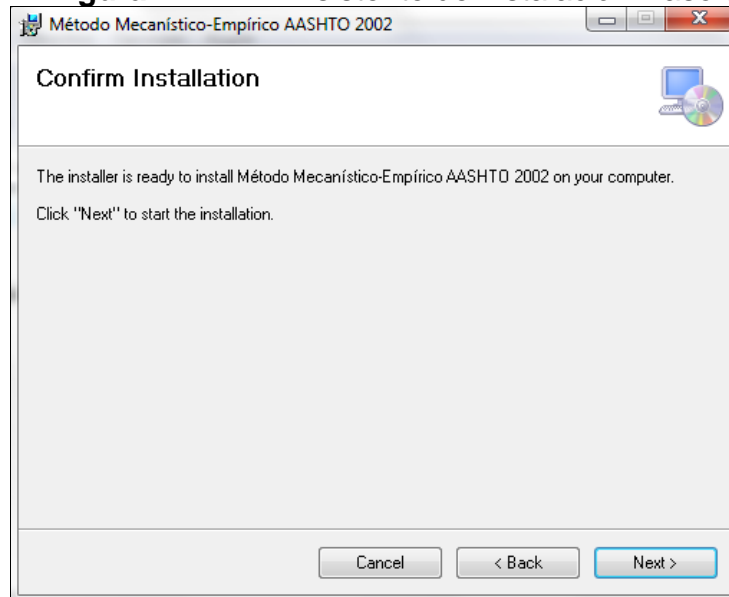
Figura 16. Asistente de instalación-Paso 2



Fuente: Instalador software

El siguiente paso del proceso de instalación consiste en confirmar las opciones anteriores, Si se requiere regresar a cambiar algún parámetro debe dar clic en el botón "Back", para continuar en el botón "Next".

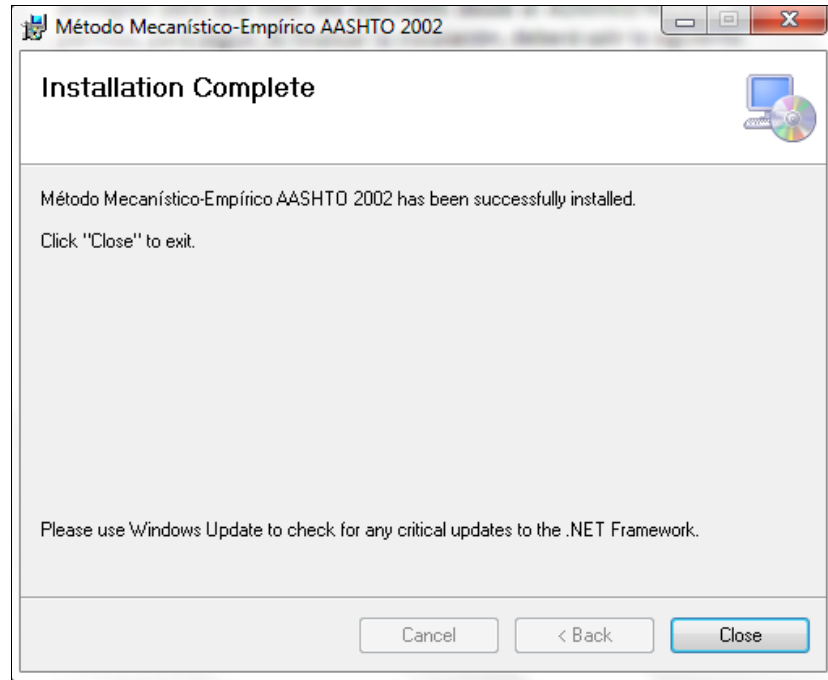
Figura 17. Asistente de instalación-Paso 3



Fuente: Instalador software

El programa comenzará a instalarse, este proceso puede tardar algunos minutos. Al finalizar la instalación deberá salir la siguiente ventana, en la cual debe seleccionar la opción "Close":

Figura 18. Asistente de instalación-Paso 4

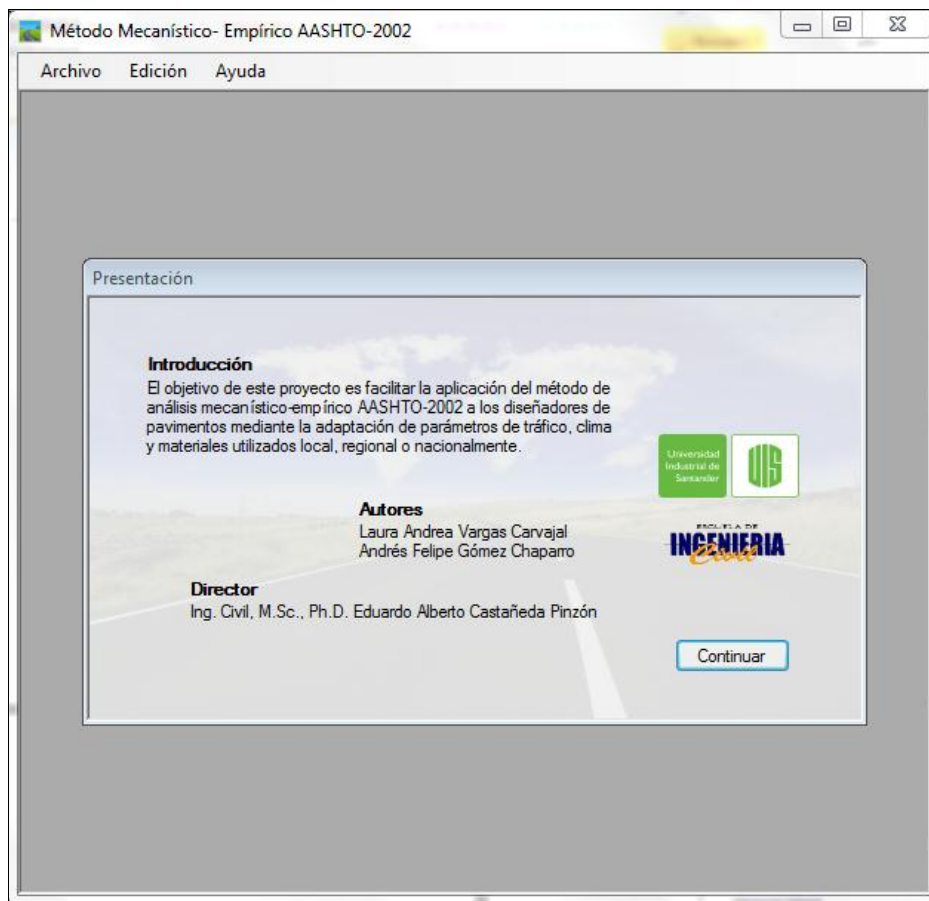


Fuente: Instalador software

4.3 INTRODUCCIÓN

Al ingresar a la aplicación se mostrará la ventana mostrada a continuación, la cual presenta una breve descripción del objetivo de este programa, sus autores y director. En la parte superior de la ventana se sitúa la barra de menús, en la cual se encuentra el Menú Archivo, Edición y Ayuda.

Figura 19. Introducción



Fuente: Elaboración propia

4.4 INFORMACIÓN GENERAL

En la siguiente ventana se muestra la información general del proyecto en evaluación, la vida de diseño de la vía, fechas de construcción y apertura al tráfico, el tipo de análisis a realizar y el tipo de pavimento, la ubicación del proyecto y sus estaciones y los límites admisibles de servicio, además, en la parte inferior izquierda se observan dos botones, guardar y cargar respectivamente, los cuales posibilitan guardar la información general de un proyecto específico o cargar esta información de un proyecto existente.

Para continuar con el siguiente paso debe seleccionar qué parámetros desea ingresar, referentes al tráfico o a la configuración estructural; la opción clima se encuentra deshabilitada ya que este campo no fue objeto de la investigación

Figura 20. Información general del proyecto

The screenshot shows a software window titled 'Método Mecanístico-Empírico AASHTO-2002 - [General]'. The interface is divided into several sections:

- Información General:** Includes 'Vida de diseño [años]' (text input), 'Calendario' with three date pickers for 'Construcción de base/subbase', 'Construcción del pavimento', and 'Apertura de tráfico', and an 'Análisis' section with 'Tipo' (radio buttons for 'Nuevo pavimento' and 'Sobrecarpeta') and 'Material' (radio buttons for 'Flexible' and 'Rígido').
- Ubicación:** Includes 'Lugar' with 'Departamento' and 'Municipio' text inputs, and 'Estaciones' with 'Estación inicial' and 'Estación final' (both 'K +' text inputs).
- Variables de Entrada:** Includes 'Límites admisibles de servicio' with an 'Editar' button, and a section for 'Ingrese los parámetros de tráfico, configuración estructural y clima de la zona del proyecto.' with three icons: 'Tráfico', 'Estructura', and 'Clima'. A red callout box points to the 'Clima' icon with the text 'Ingresar parámetros'.
- Buttons:** At the bottom left, a red box highlights two buttons: 'Cargar/Guardar' and 'Información general y Ubicación'. At the bottom right, there is a 'Continuar' button.

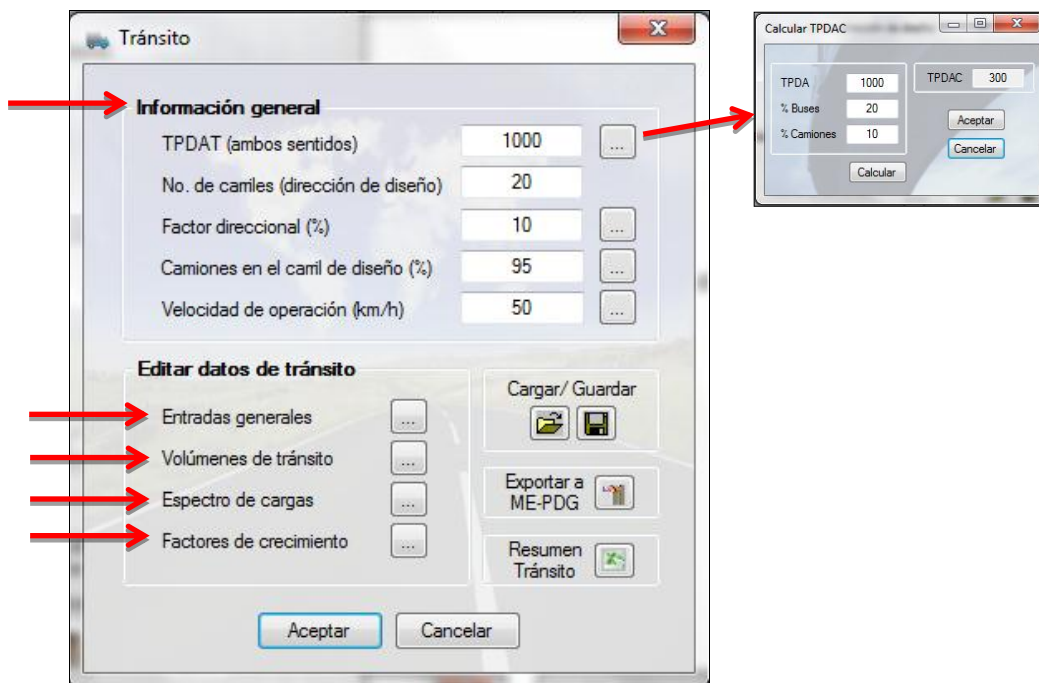
Fuente: Elaboración Propia

4.5 TRÁNSITO

La información relacionada con el tránsito se encuentra dividida en cinco categorías:

- Información general
- Entradas generales
- Volúmenes de tránsito
- Espectro de cargas
- Factores de crecimiento

Figura 21. Tránsito



Fuente: Elaboración Propia

Así mismo, es posible almacenar la información introducida guardando un nuevo proyecto y también cargar los datos relacionados con el tránsito de un proyecto previamente creado. Por otro lado, es posible exportar archivos que puedan ser utilizados en el software de aplicación de la metodología AASHTO 2002, ME-PDG.

4.5.1 Información general

Esta categoría recoge parámetros como: el tránsito promedio diario anual de camiones (TPDAC), el cual puede ser calculado a partir del tránsito promedio diario anual (TPDA) y el porcentaje correspondiente a buses y camiones; el número de carriles en la dirección de diseño, el factor direccional, el factor de carril y la velocidad de operación.

Para cada parámetro se muestra una ayuda de los valores que podría tomar tal parámetro:

Figura 22. Información general

The main window 'Información general' contains the following input fields:

- TPDAT (ambos sentidos): 1000
- No. de camiles (dirección de diseño): 20
- Factor direccional (%): 10
- Camiones en el camil de diseño (%): 95
- Velocidad de operación (km/h): 50

The 'Factor direccional' help window shows the following table:

Clasificación de la vía	Factor Direccional
Autopistas y vías multicarril con tres carriles por dirección	50
Autopistas y vías multicarril con dos carriles por dirección	50
Carreteras anchas y de dos direcciones	50
Carreteras de dos direcciones y con ancho medio	100
Carreteras estrechas y de dos direcciones	100

The 'Velocidad de operación' help window shows a table with categories and terrain types, and a traffic volume table:

CATEGORIA DE LA CARRETERA	TIPO DE TERRENO	PENDIENTE LONGITUDINAL (%)	VELOCIDAD DE DISEÑO DEL TRAMO HOMOGÉNEO (VTR) (km/h)												
			20	30	40	50	60	70	80	90	100	110			
Primaria de dos calzadas	Plano	<3													
	Ondulado	3-6													
	Montañoso	6-8													
Primaria de una calzada	Escarpado	8													
	Plano	<3													
	Ondulado	3-6													
Secundaria	Montañoso	6-8													
	Escarpado	8													
	Plano	<3													
Terciaria	Ondulado	3-6													
	Montañoso	6-8													
	Escarpado	8													

VELOCIDAD DE DISEÑO (km/h)	CONDICIONES DE TRÁFICO		
	VOLUMENES BAJOS	VOLUMENES MEDIOS	VOLUMENES ALTOS
40	38	35	33
50	47	42	40
60	56	52	45
70	63	60	55
80	72	65	60
100	88	75	-
120	105	85	-

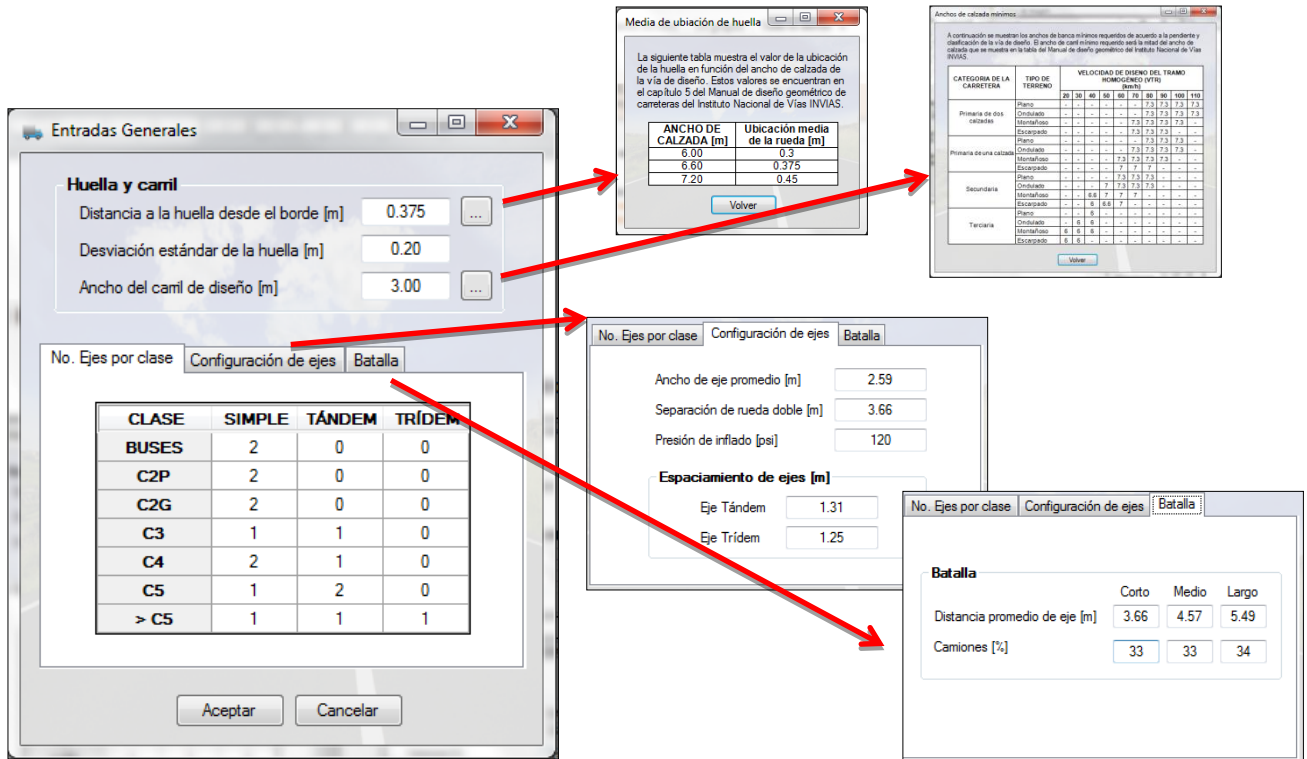
The 'Factor carril' help window shows the following table:

Clasificación de la vía	Factor Carril
Autopistas y vías multicarril con tres carriles por dirección	75
Autopistas y vías multicarril con dos carriles por dirección	90
Carreteras anchas y de dos direcciones	100
Carreteras de dos direcciones y con ancho medio	75
Carreteras estrechas y de dos direcciones	100

Fuente: Elaboración Propia

4.5.2 Entradas generales

Figura 23. Entradas generales de tránsito



Fuente: Elaboración Propia

La figura anterior muestra la ventana de entradas generales, en ella se muestra los parámetros de la ubicación de la huella y el ancho del carril y una serie de ayudas acerca de estos, además del número de ejes de cada tipo por clase de vehículo, la configuración del eje, y la distancia de batalla para camiones cortos, medios y largos

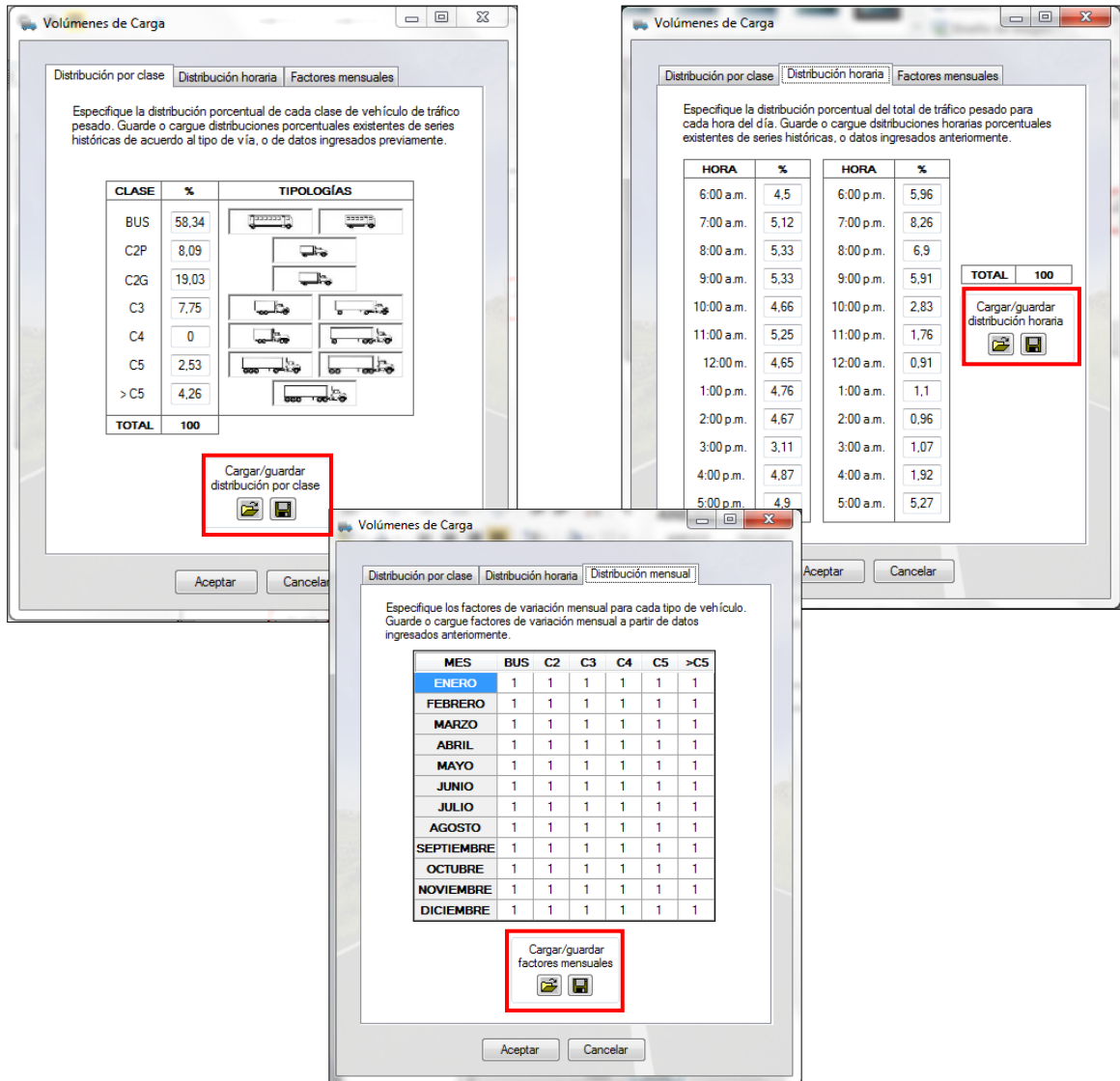
4.5.3 Volúmenes de tránsito

Esta categoría a su vez se divide en tres subcategorías que son:

- Distribución por clase
- Distribución horaria

- Distribución mensual

Figura 24. Volúmenes de tránsito

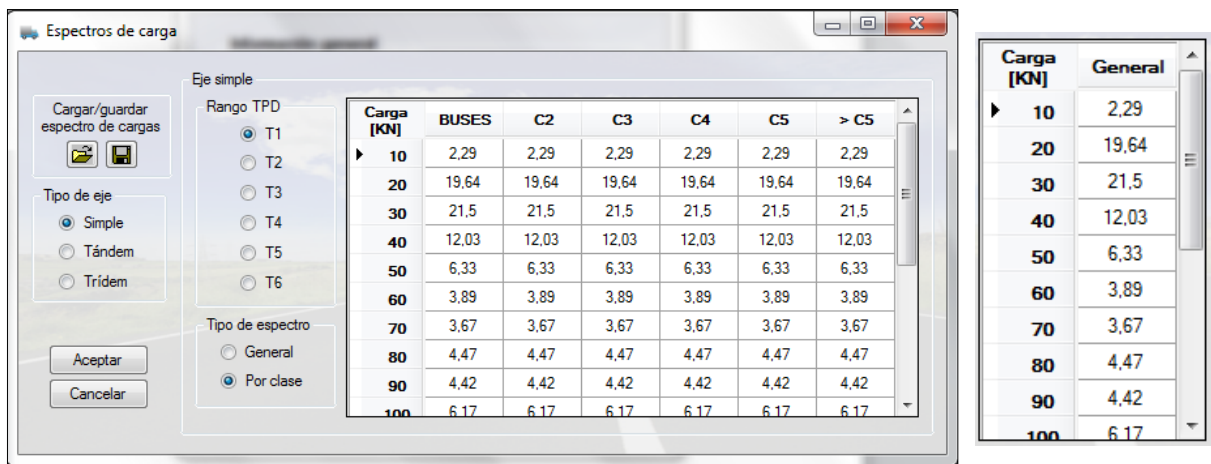


Fuente: Elaboración propia

Cada una de ellas con posibilidad de guardarse independientemente de las otras y cargarse de la misma forma. La distribución por clase especifica la distribución porcentual de cada clase de vehículo de tráfico pesado , en ella se muestran las tipologías de vehículos de conformidad con las normas colombianas y la suma de los porcentajes debe ser igual a 100%; la distribución horaria describe la distribución porcentual del total de tráfico pesado para cada hora del día y al igual que la distribución anterior, la suma de los porcentajes debe ser igual a 100; por último, la distribución mensual establece los factores de variación mensual para cada tipo de vehículo.

4.5.4 Espectro de carga

Figura 25. Espectro de carga



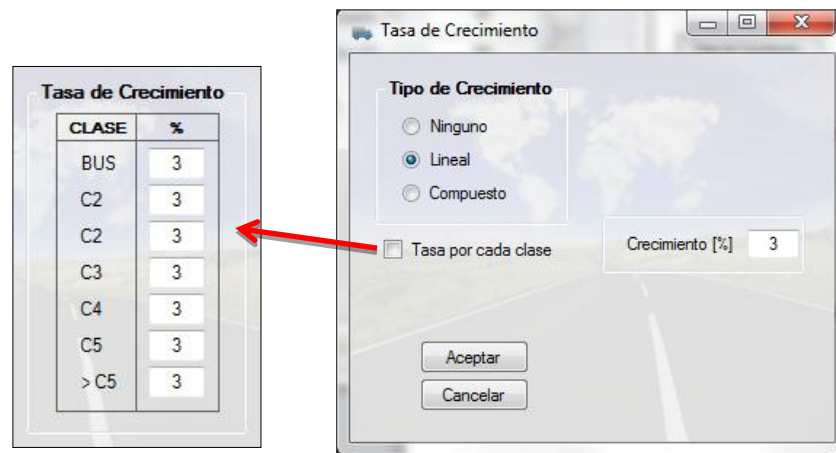
Fuente: Elaboración propia

Esta categoría describe las cargas correspondientes a cada tipo de eje (simple, tándem, tridem) por cada tipo de vehículo si su tipo es por clase y de manera general, por rangos de TPD que varían de 0 a más de 10000. Por defecto el software muestra los datos del espectro nacional, pero es posible cargar diferentes espectros así como guardarlos.

4.5.5 Factores de Crecimiento

El tipo de crecimiento del tránsito se puede dar de tres maneras: que haya un crecimiento nulo en cuyo caso el tránsito permanece constante a través del tiempo, que exista un crecimiento lineal o un crecimiento compuesto, es necesario introducir la tasa de crecimiento de tránsito la cual puede ser única o diferente para cada clase de vehículo:

Figura 26. Factores de crecimiento

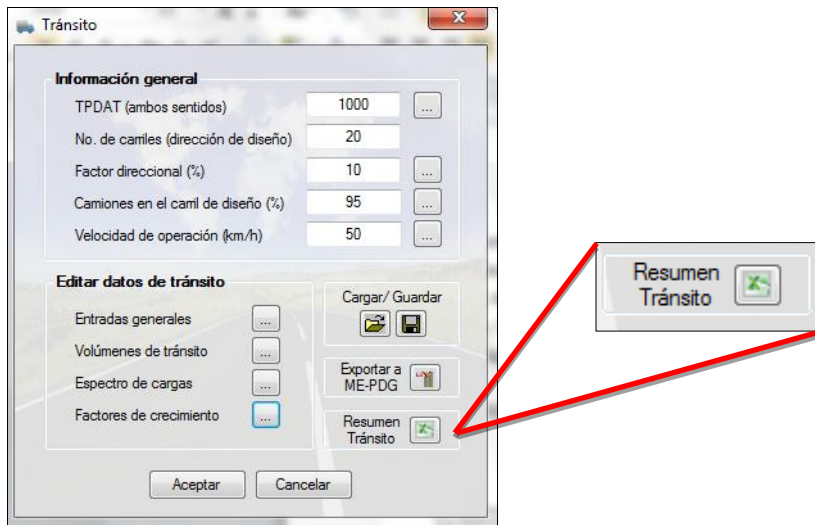


Fuente: Elaboración propia

4.5.6 Resumen de Tránsito

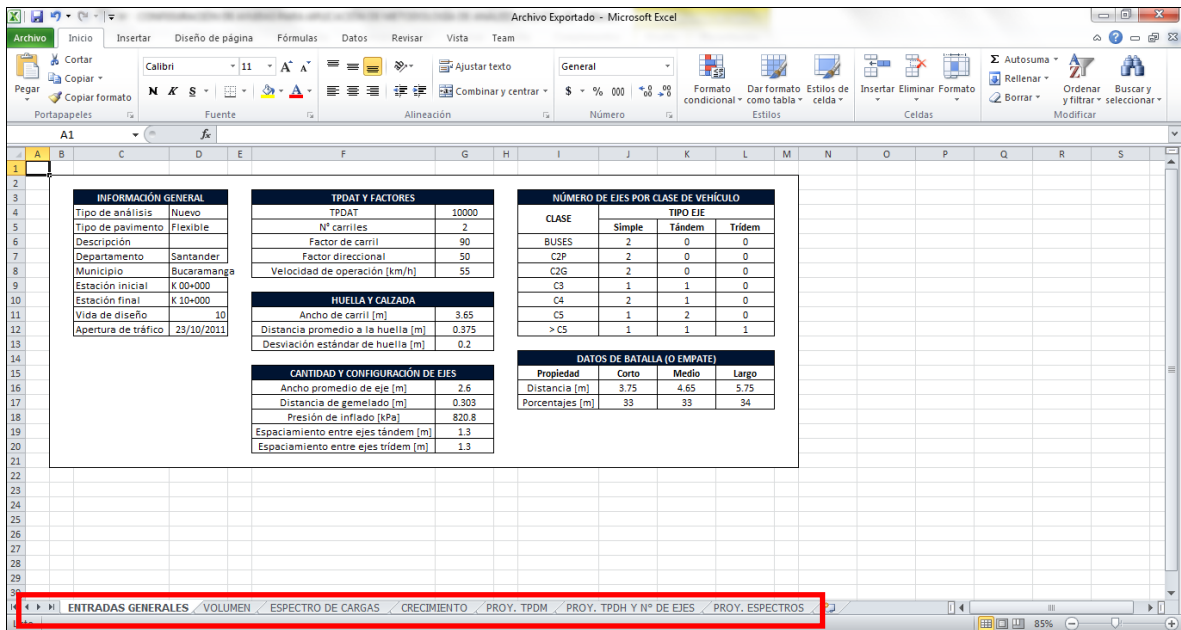
La opción "Resumen de Tránsito", a la cual se accede mediante el botón mostrado en la siguiente figura ubicado en la ventana principal de tránsito, genera un archivo de Excel que contiene un resumen detallado de toda la información referente al tráfico, además de la predicción de este de acuerdo a la vida diseño introducida y factores de crecimiento correspondientes:

Figura 27. Resumen de tránsito



: Elaboración propia

Figura 28. Resumen de tránsito – Excel



Fuente: Elaboración propia

La información de cada categoría se encuentra organizada en diferentes hojas del libro de Excel, las últimas tres hojas corresponden a la proyección del TPD

mensual, la proyección del TPD horario y número de ejes, y la proyección de espectros, respectivamente:

Figura 29. Resumen de tránsito-Proyección

AÑO	MES	TPD MENSUAL PROYECTADO						
		CLASE						
		BUS	C2P	C2G	C3	C4	C5	> C5
2011	enero	2625.3	364.05	856.35	348.75	0	113.85	191.7
2011	febrero	2625.3	364.05	856.35	348.75	0	113.85	191.7
2011	marzo	2625.3	364.05	856.35	348.75	0	113.85	191.7
2011	abril	2625.3	364.05	856.35	348.75	0	113.85	191.7
2011	mayo	2625.3	364.05	856.35	348.75	0	113.85	191.7
2011	junio	2625.3	364.05	856.35	348.75	0	113.85	191.7
2011	julio	2625.3	364.05	856.35	348.75	0	113.85	191.7
2011	agosto	2625.3	364.05	856.35	348.75	0	113.85	191.7
2011	septiembre	2625.3	364.05	856.35	348.75	0	113.85	191.7
2011	octubre	2625.3	364.05	856.35	348.75	0	113.85	191.7
2011	noviembre	2625.3	364.05	856.35	348.75	0	113.85	191.7
2011	diciembre	2625.3	364.05	856.35	348.75	0	113.85	191.7
2012	enero	2704.059	374.9715	882.0405	359.2125	0	117.2655	197.451
2012	febrero	2704.059	374.9715	882.0405	359.2125	0	117.2655	197.451
2012	marzo	2704.059	374.9715	882.0405	359.2125	0	117.2655	197.451
2012	abril	2704.059	374.9715	882.0405	359.2125	0	117.2655	197.451
2012	mayo	2704.059	374.9715	882.0405	359.2125	0	117.2655	197.451
2012	junio	2704.059	374.9715	882.0405	359.2125	0	117.2655	197.451
2012	julio	2704.059	374.9715	882.0405	359.2125	0	117.2655	197.451
2012	agosto	2704.059	374.9715	882.0405	359.2125	0	117.2655	197.451
2012	septiembre	2704.059	374.9715	882.0405	359.2125	0	117.2655	197.451

AÑO	MES	HORA	TPD HORARIO PROYECTADO							EJES SIMPLES							
			CLASE							EJES SIMPLES							
			BUS	C2P	C2G	C3	C4	C5	> C5	BUS	C2P	C2G	C3	C4	C5	> C5	BUS
2011	enero	00:00	23.89023	3.312855	7.792785	3.173625	0	1.036035	1.74447	23.89023	3.312855	7.792785	3.173625	0	1.036035	1.74447	23.89023
2011	enero	01:00	28.8783	4.00455	9.41985	3.83625	0	1.25235	2.1087	28.8783	4.00455	9.41985	3.83625	0	1.25235	2.1087	28.8783
2011	enero	02:00	25.20288	3.49488	8.22096	3.348	0	1.09296	1.84032	25.20288	3.49488	8.22096	3.348	0	1.09296	1.84032	25.20288
2011	enero	03:00	28.09071	3.895335	9.162945	3.731625	0	1.218195	2.05119	28.09071	3.895335	9.162945	3.731625	0	1.218195	2.05119	28.09071
2011	enero	04:00	50.40576	6.98976	16.44192	6.696	0	2.18592	3.68064	50.40576	6.98976	16.44192	6.696	0	2.18592	3.68064	50.40576
2011	enero	05:00	138.3533	19.18544	45.12965	18.37913	0	5.999895	10.10259	138.3533	19.18544	45.12965	18.37913	0	5.999895	10.10259	138.3533
2011	enero	06:00	118.1385	16.38225	38.53575	15.69375	0	5.12325	8.6265	118.1385	16.38225	38.53575	15.69375	0	5.12325	8.6265	118.1385
2011	enero	07:00	134.4154	18.63936	43.84512	17.856	0	5.82312	9.81504	134.4154	18.63936	43.84512	17.856	0	5.82312	9.81504	134.4154
2011	enero	08:00	139.9285	19.40387	45.64346	18.58838	0	6.068205	10.21761	139.9285	19.40387	45.64346	18.58838	0	6.068205	10.21761	139.9285
2011	enero	09:00	139.9285	19.40387	45.64346	18.58838	0	6.068205	10.21761	139.9285	19.40387	45.64346	18.58838	0	6.068205	10.21761	139.9285
2011	enero	10:00	122.339	16.96473	39.90591	16.25175	0	5.30541	8.93322	122.339	16.96473	39.90591	16.25175	0	5.30541	8.93322	122.339
2011	enero	11:00	137.8283	19.11263	44.95838	18.30938	0	5.977125	10.06425	137.8283	19.11263	44.95838	18.30938	0	5.977125	10.06425	137.8283
2011	enero	12:00	122.0765	16.92833	39.82028	16.21688	0	5.294025	8.91405	122.0765	16.92833	39.82028	16.21688	0	5.294025	8.91405	122.0765
2011	enero	13:00	124.9643	17.32878	40.76226	16.6005	0	5.41926	9.12492	124.9643	17.32878	40.76226	16.6005	0	5.41926	9.12492	124.9643
2011	enero	14:00	122.6015	17.00114	39.99155	16.28663	0	5.316795	8.95239	122.6015	17.00114	39.99155	16.28663	0	5.316795	8.95239	122.6015
2011	enero	15:00	81.64683	11.32196	26.63249	10.84613	0	3.540735	5.96187	81.64683	11.32196	26.63249	10.84613	0	3.540735	5.96187	81.64683

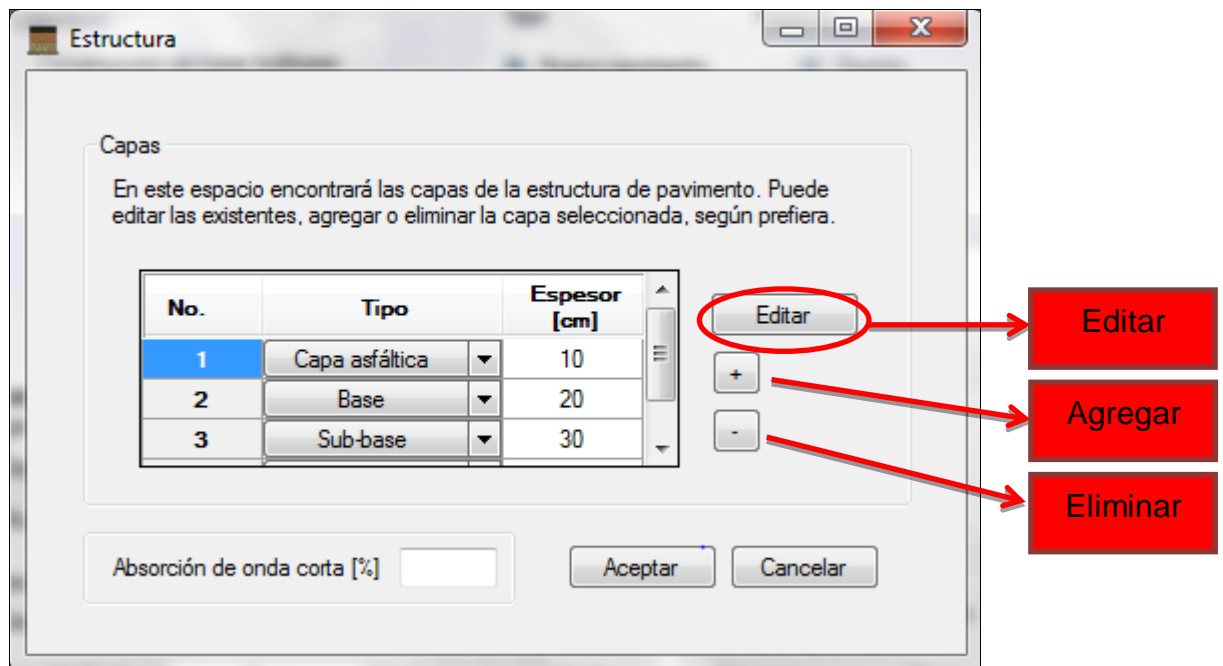
AÑO	MES	HORA	EJES SIMPLES																	
			20-oct	20-30	30-40	40-50	50-60	60-70	70-80	80-90	90-100	100-110	110-120	120-130	130-140	140-150	150-160	160-170	170-180	180-190
2011	enero	00:00	173.916	1431.58	1632.84	913.629	480.137	295.429	278.721	339.478	335.681	468.586	489.091	416.183	202.016	76.7053	0	0	0	0
2011	enero	01:00	210.228	1803.01	1973.76	1044.39	581.111	357.113	336.916	410.358	405.768	566.423	581.209	503.079	244.195	92.7207	0	0	0	0
2011	enero	02:00	183.412	1573.53	1722.55	963.828	507.151	311.662	294.036	358.131	354.125	494.333	515.965	439.051	213.116	80.9199	0	0	0	0
2011	enero	03:00	204.495	1753.83	1919.93	1074.27	565.263	341.373	327.727	399.166	394.702	550.375	575.085	489.358	237.535	90.192	0	0	0	0
2011	enero	04:00	366.344	3147.06	3445.1	1821.66	1014.3	623.324	589.071	716.251	768.249	989.665	1091.93	878.801	426.232	161.84	0	0	0	0
2011	enero	05:00	1007.18	8638.03	9456.1	5291.02	2784.05	1710.89	1614.13	1985.99	1944	2713.68	2832.43	2410.2	1168.92	444.217	0	0	0	0
2011	enero	06:00	860.024	7375.33	8074.46	4517.34	2317.27	1460.31	1378.29	1678.74	1659.36	2317.18	2418.58	2058.05	988.38	379.312	0	0	0	0
2011	enero	07:00	378.517	6392.17	3186.95	5140.42	2704.81	1662.2	1568.19	1910.03	1869.67	2636.44	2751.81	2341.6	1136.62	443.573	0	0	0	0
2011	enero	08:00	1018.65	8796.38	3563.75	5351.25	2815.75	1730.37	1632.51	1988.37	1966.13	2744.58	2864.68	2437.65	1183.24	443.274	0	0	0	0
2011	enero	09:00	1018.65	8796.38	3563.75	5351.25	2815.75	1730.37	1632.51	1988.37	1966.13	2744.58	2864.68	2437.65	1183.24	443.274	0	0	0	0
2011	enero	10:00	830.603	7639.18	8261.56	4619.58	2461.8	1512.86	1427.3	1738.43	1719.98	2399.57	2504.58	2151.22	1064.5	392.799	0	0	0	0
2011	enero	11:00	1093.36	8605.25	3420.21	5270.94	2713.48	1704.4	1608.01	1959.53	1936.62	2703.98	2823.28	2401.06	1165.48	442.531	0	0	0	0
2011	enero	12:00	888.692	7621.79	8343.61	4668.54	2456.52	1509.61	1424.24	1734.7	1715.29	2394.42	2499.2	2126.65	1032.28	391.566	0	0	0	0
2011	enero	13:00	909.715	7802.09	8540.99	4778.98	2514.63	1545.32	1457.93	1775.73	1715.87	2451.07	2558.32	2176.96	1056.7	401.228	0	0	0	0
2011	enero	14:00	892.514	7654.58	8379.5	4688.62	2467.08	1516.1	1430.36	1742.16	1722.67	2404.72	2509.35	2135.8	1036.72	393.642	0	0	0	0
2011	enero	15:00	534.372	5097.53	5580.35	3122.4	1642.96	1009.65	952.553	1160.19	1147.82	1601.43	1671.51	1422.34	690.406	262.147	0	0	0	0
2011	enero	16:00	330.738	7982.39	8738.37	4883.42	2572.74	1581.03	1491.62	1816.77	1796.45	2507.71	2617.45	2227.27	1081.12	410.5	0	0	0	0
2011	enero	17:00	336.471	8031.51	8732.19	4919.54	2586.59	1590.77	1500.61	1821.96	1801.51	2523.16	2633.57	2240.99	1091.78	415.028	0	0	0	0
2011	enero	18:00	1139.05	8783.01	10634.2	5983.77	3148.57	1834.3	1825.47	2223.39	2198.52	3068.98	3203.28	2725.77	1323.09	502.378	0	0	0	0
2011	enero	19:00	1578.62	13538.9	14821.1	8232.34	4363.62	2681.59	2529.33	3061.42	3046.35	4253.32	4439.45	3777.66	1833.68	696.248	0	0	0	0
2011	enero	20:00	1318.7	11309.8	12380.8	6397.52	3645.15	2240.07	2113.38	2574.06	2545.27	3553.01	3708.5	3195.68	1531.77	581.612	0	0	0	0

Fuente: Elaboración propia

4.6 ESTRUCTURA

En primera instancia se encuentra un espacio para definir las capas de la estructura del pavimento, su tipo, el cual puede ser: capa asfáltica, losa de concreto, base, subbase y subrasante, y el espesor de cada capa; en la parte derecha se encuentran ubicadas tres opciones: agregar una capa, eliminar una capa y editar una capa, esta última opción debe escogerse para poder acceder a las propiedades de cada tipo de capa.

Figura 30. Capas de la estructura de pavimento



Fuente: Elaboración propia

4.6.1 Capa asfáltica

Para este tipo de capa se especifican parámetros generales como la temperatura de referencia y el coeficiente de Poisson, propiedades volumétricas como el contenido de vacíos y el porcentaje efectivo de ligante y propiedades térmicas

como conductividad térmica y la capacidad calórica. Además, describe propiedades para la mezcla y para el ligante.

4.6.1.1 Mezcla asfáltica

Las propiedades de la mezcla asfáltica dependen de si esta información se especificará por medio de ensayos (Nivel 1) o de manera estándar que hace referencia a las especificaciones de construcción y normas (Nivel 3). Si la opción escogida es “ensayo” es necesario introducir ensayos de módulo dinámico a diferentes temperaturas y frecuencias, además es posible cargar o guardar la información; si la opción escogida es “estándar” se especificará la granulometría que debe cumplir la mezcla.

Figura 31. Propiedades de la mezcla asfáltica

The screenshot shows the 'Capa Asfáltica' software window. The 'Mezcla' tab is active. On the left, there is a 'Gradación del agregado' section with four input fields: '% Retenido 3/4"', '% Retenido 3/8"', '% Retenido No. 4', and '% Pasa No. 200'. A red arrow points from this section to the 'Mezcla' section. The 'Mezcla' section has two radio buttons: 'Ensayo' (selected) and 'Estándar'. Below them are 'Guardar' and 'Cargar' buttons. To the right is a table titled 'Ensayo de módulo dinámico' with columns for 'Temp. [°C]' and 'Frecuencia [Hz]'. The table has rows for temperatures 20, 30, 40, and 50, and columns for frequencies 0.1, 1, and 10. Below the table are two sets of 'Agregar/Quitar' buttons for 'Temperaturas' and 'Frecuencias'. At the bottom, there are three sections: 'Información general' with 'Temp. referencia [°C]' and 'Coeficiente de Poisson' fields; 'Propiedades volumétricas' with '% Efectivo de ligante', '% Vacíos con aire', and 'Peso unitario' fields; and 'Propiedades térmicas' with 'Conductividad térmica' and 'Capacidad calórica' fields. 'Aceptar' and 'Cancelar' buttons are at the bottom right.

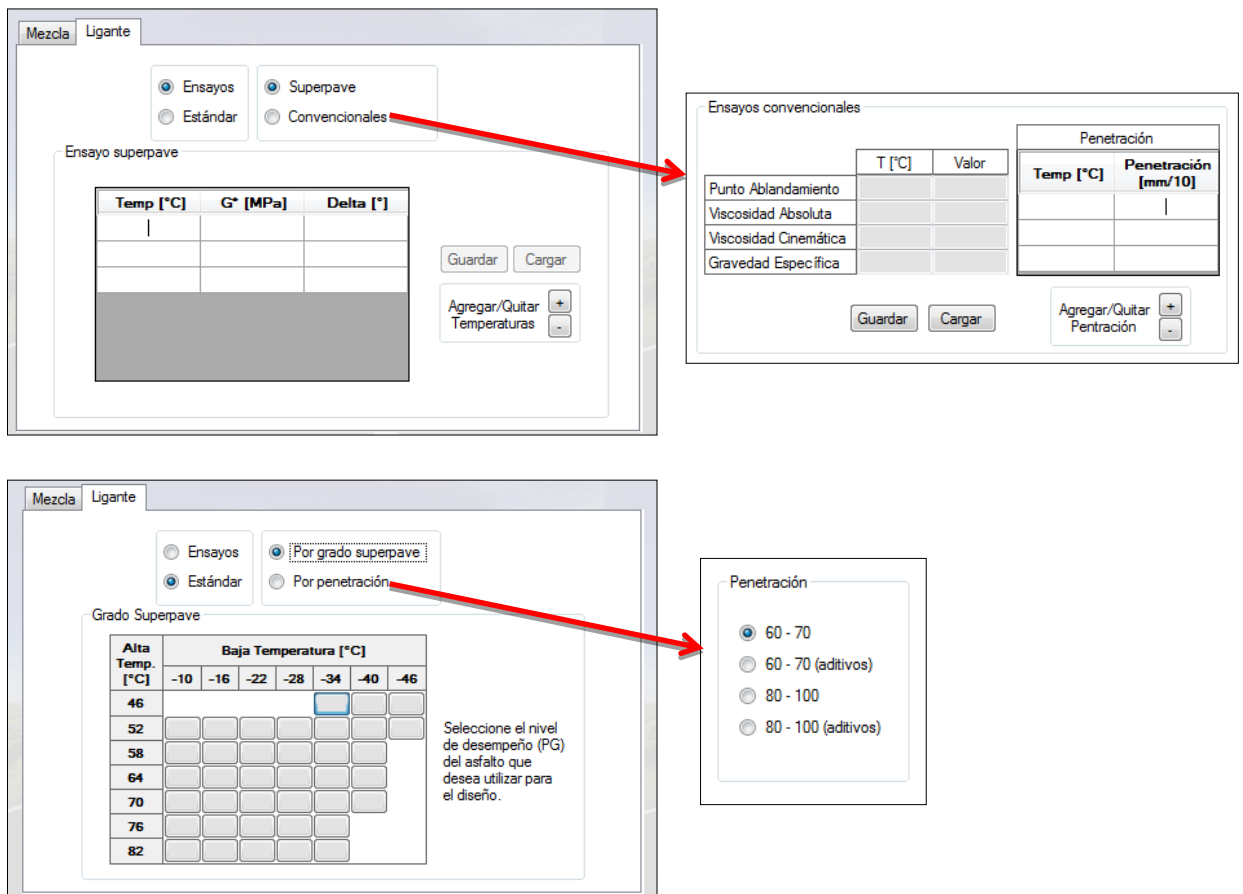
Temp. [°C]	Frecuencia [Hz]		
	0.1	1	10
20			
30			
40			
50			

Fuente: Elaboración propia

4.6.1.2 Ligante

De la misma forma, las propiedades de la mezcla asfáltica dependen también de los niveles jerárquicos de información, el Nivel 1 a partir de ensayos y el Nivel 3 a partir de valores estándar. Si la opción escogida es “ensayos” se pueden introducir directamente, cargar y guardar ensayos de superpave o ensayos convencionales, por otro lado, si la opción escogida es “Estándar” se podrá escoger el tipo de ligante a partir de dos criterios, por grado superpave o por penetración.

Figura 32. Propiedades del ligante

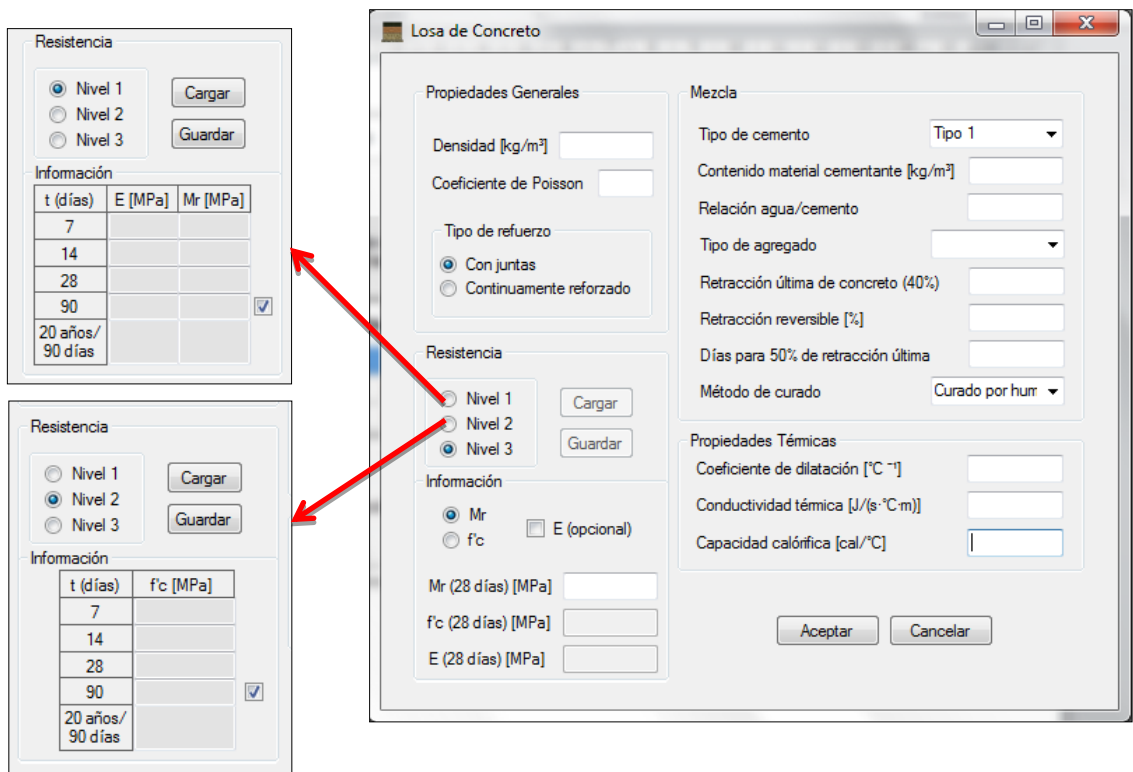


Fuente: Elaboración propia

4.6.2 Losa de concreto

La siguiente ventana muestra las propiedades de la losa de concreto por categorías: propiedades generales, propiedades de la mezcla, propiedades térmicas y resistencia del concreto en tres niveles jerárquicos: el nivel 1 a partir de ensayos de módulo de rotura, el nivel 2 a partir de ensayos de compresión y el tercer nivel calculando la resistencia mediante parámetros específicos.

Figura 33. Propiedades de la losa de concreto



Fuente: Elaboración propia

4.6.3 Base, Subbase y Subrasante

Las siguientes son las propiedades especificadas para cada una de las capas de base, subbase y subrasante. Los parámetros se clasifican en diferentes conjuntos: las propiedades generales ubicadas en la parte izquierda de la ventana contiene

entre otras cosas el tipo de suelo a partir de la clasificación AASHTO Y UCS; la segunda categoría se refiere al módulo resiliente, el cual puede ser obtenido de tres formas de acuerdo a un nivel de confianza, por medio de un ensayo triaxial (Nivel 1), a partir de correlaciones basadas en otros parámetros (Nivel 2) y valores por defecto (Nivel 3); y la última categoría especifica la granulometría de los agregados de la capa, límites de Atterberg, entre otros.

Figura 34. Propiedades de la base, subbase y subrasante

The image shows a software interface for defining material properties. It is divided into several sections:

- Propiedades Generales:** Includes input fields for K_0 (presión lateral) and Coeficiente de Poisson. It also has a 'Clasificación de Suelo' section with radio buttons for AASHTO (selected) and UCS, and a 'Material' dropdown menu set to 'A 1-a'. There are 'Aceptar' and 'Cancelar' buttons.
- Módulo Resiliente:** This section is further divided into:
 - Granulometría:** Contains three radio buttons for 'Nivel 1', 'Nivel 2', and 'Nivel 3'. A checkbox 'Calcular Mr a partir de granulometría' is present. Below this is a 'Coeficientes Mr' section with input fields for k_1 and k_2 , and another checkbox 'Calcular Mr a partir de ensayo triaxial' with an 'Ensayo' button. At the bottom, there is another set of radio buttons for 'Nivel 1', 'Nivel 2', and 'Nivel 3', a checkbox 'Calcular Mr a partir de granulometría', and an 'Mr [MPa]' input field.
 - Coeficientes Mr:** A separate section with input fields for k_1 and k_2 , and a checkbox 'Calcular Mr a partir de ensayo triaxial' with an 'Ensayo' button.
 - Mr [MPa]:** A section with radio buttons for 'Nivel 1', 'Nivel 2' (selected), and 'Nivel 3', a checkbox 'Calcular Mr a partir de granulometría', and an 'Mr [MPa]' input field.
 - Tipo propiedad:** Contains radio buttons for 'CBR' (selected), 'Valor R', 'Coeficiente ai', and 'Penetrómetro dinámico de cono'. Each has an associated input field. A 'Calcular' button is at the bottom.
- Granulometría Table:** A table with columns 'Tamiz', 'Límite Inferior', and 'Límite Superior'. The rows list sieve sizes from #200 down to #2".
- Soil Parameters:** A list of parameters with input fields: IP, LL, Grado Compactación (set to 'Si'), % Pasa 200, % Pasa 40, % Pasa 4, D10, D20, D30, D60, D90, Densidad Seca, Gravedad específica, Permeabilidad, Humedad óptima, Saturación óptima, af, bf, cf, and hr.

Fuente: Elaboración propia

Para este caso, la información de Nivel 1 corresponde a un ensayo triaxial, para el cual se introducen los resultados del ensayo correspondiente y se calculan los coeficientes K1 y K2:

Figura 35. Ensayo triaxial

Ensayo Triaxial

Longitud inicial [in] 4

PRESIÓN DE CONFINAMIENTO	ESFUERZO DESVIADOR	DEFORMACIÓN RECUPERABLE
20	1	0,264
20	2	0,496
20	5	1,184
20	10	2,284
20	15	3,428
20	20	4,42
15	1	0,26
15	2	0,512
15	5	1,3
15	10	2,5
15	15	3,636
15	20	4,572

Coeficientes

Calcular K1 0,3511445 Aceptar

K2 3685,3705 Cancelar

Fuente: Elaboració propia

5. CONCLUSIONES Y RECOMENDACIONES

- El factor de ajuste mensual es un parámetro utilizado para determinar la variación del TPD a lo largo de los meses del año, los conteos que se realizan en las diferentes estaciones de conteo a cargo del Instituto Nacional de Vías se llevan a cabo durante siete días, una semana del año, con lo cual es posible conocer la variación del tránsito anual, mas no mensualmente. En base a esta observación, se recomienda realizar conteos mensuales o implementar estaciones de conteo automatizadas para que sea posible determinar estos factores.
- A partir de la investigación se encontró que el estudio de transporte de carga más reciente realizado en el Área Metropolitana de Bucaramanga es del año 2006, por lo cual se plantea la posibilidad de realizar estos estudios anualmente.
- La distribución vehicular del transporte de carga utilizada en la metodología AASHTO 2002 consta de diez (10) clases, mientras que la utilizada para los conteos vehiculares en Colombia consta de seis (6) clases que corresponden a buses, C2P, C2G, C3-C4, C5 Y >C5.
- Dentro de los ensayos que se realizan en Colombia para caracterizar la mezcla asfáltica no se encuentran aquellos que miden el coeficiente de absorción de onda corta y propiedades térmicas como la conductividad térmica y la capacidad calórica, los cuales son parámetros importantes para estimar la temperatura a través de la sección transversal del pavimento.
- Se propone desarrollar un software de aplicación de la metodología AASHTO 2002, para lo cual se deja abierto el código fuente de la interfaz

gráfica desarrollada en el presente trabajo con el fin de que sirva de base a futuras investigaciones.

- Partiendo de que la metodología AASHTO 2002 combina métodos mecanísticos y empíricos, estos últimos basados en el experimento vial de la AASHO, se sugiere realizar ensayos similares para que sea posible calibrar los modelos mecanísticos y sus fórmulas.

BIBLIOGRAFÍA

BRAVO, Pablo Emilio. Diseño de carreteras. Técnica y análisis del proyecto. 6 ed. Bogotá D.C, 1998.

BUCARAMANGA. DIRECCIÓN DE TRÁNSITO DE BUCARAMANGA. Resolución 735 (6, diciembre, 2010). Por la cual se reglamenta el tránsito, cargue y descargue de los vehículos de carga, en las vías del Municipio de Bucaramanga. Bucaramanga: La Dirección, 2010.

COLOMBIA. MINISTERIO DE TRANSPORTE. Especificaciones generales de construcción de carreteras. Bogotá: INVIAS, 2007.

COLOMBIA. MINISTERIO DE TRANSPORTE. Manual de diseño de pavimentos asfálticos en vías con bajos volúmenes de tránsito. Bogotá: INVIAS, 2007.

COLOMBIA. MINISTERIO DE TRANSPORTE. Manual de diseño de pavimentos de concreto para vías con bajos, medios y altos volúmenes de tránsito. Bogotá: INVIAS, 2008.

COLOMBIA. MINISTERIO DE TRANSPORTE. Resolución 001782 (28, diciembre, 2004). Por la cual se adoptan los límites de pesos y dimensiones en los vehículos de transporte terrestre automotor de carga por carretera, para su operación normal en la red vial a nivel nacional. Bogotá D.C: El Ministerio, 2004.

COLOMBIA. MINISTERIO DE TRANSPORTE. Resolución 4100 (8, mayo, 2009). Por la cual se modifica el artículo 8° de la resolución 4100 del 28 de diciembre de 2004. Bogotá D.C: El Ministerio, 2004.

COLOMBIA. MINISTERIO DE TRANSPORTE. Volúmenes de tránsito 2008. Bogotá: INVIAS, 2008.

CORREDOR, Gustavo. Experimento Vial de la AASHO y las Guías de Diseño AASHTO. Universidad Nacional de Ingeniería. Facultad de Tecnología de la Construcción. 4 p.

GARCIA ALADIN, Maria Fernanda. Catálogo de diseño de pavimentos rígidos de la PCA adaptado a las condiciones de tránsito colombianas. Popayán: UNICAUCA, 2002.

MEZA, Manuel. Diseño de Pavimentos ME-PDG [diapositivas]. Universidad del Sinú. Facultad de Ingeniería Civil. 2009.

NATIONAL COOPERATIVE HIGHWAY RESEARCH PROGRAM. Background, Scope and Overview. En: Guide for Mechanistic-Empirical Design of New and Rehabilitated Pavement Structures. Part 1: Introduction. Illinois, 2004

-----,-----, Part 2: Design Inputs. Illinois, 2004.

RICO RODRIGUEZ, Alfonso. JUAREZ BADILLO, Eulalio. Mecánica de suelos: fundamentos de la mecánica de suelos. México: Limusa, 1980.

RIVERO, María Cristina y NEIRA, Jeffrey Javier. Práctica empresarial convenio No 154 Gobernación de Santander y Universidad Industrial de Santander. Estudios y diseños definitivos ampliación a doble calzada de la vía Aeropuerto-Lebrija, rehabilitación de la vía Palenque-Café Madrid y ciclovia entre Girón y Lebrija. Trabajo de grado Ingeniero Civil. Bucaramanga: Universidad Industrial de Santander. Escuela de Ingeniería Civil. Facultad de Ingenierías Físico-Mecánicas, 2006.

ANEXOS

Anexo A. Serie histórica y composición del tránsito promedio diario semanal- Santander

MINISTERIO DE TRANSPORTE INSTITUTO NACIONAL DE VÍAS SUBDIRECCIÓN DE APOYO TÉCNICO		SERIE HISTÓRICA Y COMPOSICIÓN DE TRÁNSITO PROMEDIO DIARIO SEMANAL																							
ESTAC. No.	SECTOR	FECHA	LONG. KM.	REGIONAL Nº 22 SANTANDER																	CAMIONES DE CONTEO DE 2008				
				17-IV 1996	15-IV 1997	27-II 1998	27-XI 1999	17-X 2000	15-XII 2001	15-XII 2002	31-X 2003	24-XI 2004	2005	2006	2007	2008	DESVAL. ESTAND.	TOTAL	C - 2P	C - 2G	C - 3 - 4	C - 5	>C - 5		
126	BARBOSA-GUESA	12	4.087	3.763	4.149	4.500	4.110	2.980	4.318	4.650	4.874	5.414	3.758	6.091	8.862	759	13.311	3.040	5.454	1.181	1.117	2.519			
127	VADO REAL-OBA	28	2.213	2.036	2.220	2.277	2.010	2.842	2.154	2.192	2.300	2.835	2.589	3.182	4.111	625	11.048	3.866	3.364	872	766	3.180			
128	OBA-SOCCORRO	29	2.353	2.092	2.256	2.386	2.617	3.641	2.333	2.419	2.750	3.182	2.925	3.472	4.159	410	11.105	3.203	3.470	868	1.010	2.445			
129	SOCCORRO-BERLIN	7	3.443	3.209	3.430	3.582	3.160	4.226	3.485	3.920	4.006	4.422	4.009	4.470	5.485	504	12.290	4.208	3.481	1.039	945	2.617			
141	BERLIN-SAN GIL	17	4.068	4.072	3.985	4.373	3.985	4.681	4.018	4.391	4.489	4.604	4.366	5.324	6.435	665	12.414	2.674	5.410	970	803	2.507			
142	CHARALA-SAN GIL	37	1.583	1.897	1.253	1.449	1.259	1.477	1.182	866	1.350	1.102	1.395	*	*										
143	SAN GIL - LOS CUROS	70	3.552	2.456	2.521	2.656	2.245	3.391	2.565	2.890	2.799	3.528	3.172	3.895	4.623	416	11.822	3.376	3.857	1.080	778	2.731			
144	LOS CUROS-PIEDECUESTA	10	4.112	4.433	4.770	4.351	4.331	5.852	4.846	4.723	6.021	5.747	4.909	8.477	8.523	859	20.807	1.594	12.844	2.814	1.035	2.520			
145	LOS CUROS-MALAGA	124	197	138	236	246	137	187	192	180	211	244	152	351	568	29	563	223	338	2	0	0			
146	PIEDECUESTA-FLORIDABLANCA	12	23.516	22.981	24.369	27.303	25.654	27.721	28.035	28.672	25.139	27.935	33.599	37.475	37.730	3.250	27.066	5.253	12.365	5.038	1.842	2.738			
148	BARICHARA-SAN GIL	20	2.441	1.662	2.774	2.839	2.852	2.241	3.076	3.314	3.206	2.369	3.218	3.343	3.868	410	2.523	28.039	1.828	63	13	38			
151	VELEZ-BARBOSA	17	1.979	1.373	1.499	1.472	1.546	1.959	1.266	1.384	1.969	1.477	1.592	1.787	1.852	141	1.940	923	891	94	4	29			
152	PUNTE NACIONAL-BARBOSA	11	3.402	2.293	2.399	3.121	3.649	2.910	3.116	3.594	4.270	3.764	2.997	4.691	5.623	258	10.693	2.425	4.906	840	850	1.672			
153	VELEZ-LANAUZURI	57	458	147	347	220	1.585	989	428	404	654	467	467	515	479	57	650	288	338	21	2	1			
154	PUERTO OLAYA-PUERTO ARAUJO	69	715	564	991	1.395	1.147	1.969	1.486	1.443	1.943	1.434	2.968	1.416	295	28	627	255	267	81	20	14			
158	BUCARAMANGA-RIO NEGRO	16	3.475	3.834	3.619	4.189	3.695	4.571	4.041	4.178	3.584	4.697	5.435	4.944	4.909	199	10.949	3.046	3.907	740	381	2.235			
159	EL PLAYON-MALPADO	20	4.284	1.898	1.792	1.917	1.618	2.414	1.877	2.025	1.803	2.113	3.291	2.558	3.394	1.091	11.482	1.836	2.684	650	473	5.839			
161	MALPADO-SAN ALBERTO	31	1.822	1.817	1.672	1.839	1.579	2.372	1.765	2.067	1.841	2.180	3.045	2.537	2.906	81	8.277	1.754	2.630	653	454	2.786			
166	GIRON-TE DE AEROPUERTO	6	11.534	9.570	5.767	4.246	4.217	10.565	8.584	8.690	10.376	9.540	9.900	8.698	7.776	497	15.206	5.404	5.250	2.177	702	1.673			
167	SAN VICENTE-LA RENTA	42	229	185	184	217	169	280	186	190	102	163	173	*	*										
168	CONCEPCION-CERRITO	10	194	298	276	304	305	306	306	383	383	476	1.309	*											
169	CONCEPCION-MALAGA	12	185	1.143	890	708	2.192	588	1.429	789	1.200	986	1.910	1.141	238	1.957	719	1.015	124	11	88				

* NO SE INSTALO ESTACION DE CONTEO



MINISTERIO DE TRANSPORTE INSTITUTO NACIONAL DE VÍAS SUBDIRECCIÓN DE APOYO TÉCNICO		SERIE HISTÓRICA Y COMPOSICIÓN DE TRÁNSITO PROMEDIO DIARIO SEMANAL REGIONAL N° 22 SANTANDER															CAMIONES DE CONTEO DE 2008 TOTAL SEMANAL Y DISTRIBUCIÓN PORCENTUAL					
ESTAC. No.	SECTOR	FECHA	17-IV	15-IV	27-II	27-XI	17-X	15-XII	15-XII	31-X	24-XI	16-06-08	DESVIA	ESTAND.	ESTAND.	TOTAL	C - 2P	C - 2G	C - 3 - 4	C - 5	≥C - 5	
			1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006									2007
198	MIRANDA-MALAGA	6	564	862	1.030	917	1.777	1.024	1.101	899	4.874	5.414	3.758	6.091	6.862	13.311	930	1.458	271	56	114	
199	CAPITANEJO-MIRANDA	29	613	358	378	433	460	450	454	411	496		582	609	729	134	1.353	401	735	102	31	84
633	GUEPSA-VADO REAL	27	4.022	2.246	2.980	3.367	2.894	3.325	3.124	2.721	2.675	3.171	3.275	3.745	4.437	487	11.425	3.411	3.844	919	930	2.521
637	FLORIDA BLANCA-BUCARAMANGA	5	40.483	41.890	45.245	47.528	49.041	53.862	50.247	52.150	50.112	48.347	46.569	62.493	59.272	7.754	12.471	5.193	5.072	1.391	362	453
638	RIONEGRO-EL PLAYON	25	2.257	2.459	2.351	2.422	2.108	2.922	2.238	5.194	2.281	2.612	3.498	3.135	3.440	202	8.962	2.034	2.922	698	457	2.851
667	BUCARAMANGA-BERLIN	60	1.988	1.622	1.924	1.698	1.645	2.418	2.068	2.471	2.744	1.897	2.405	2.610	3.197	430	8.894	2.369	2.333	848	942	2.402
668	PTO ARAUJO-LANDAZURI	61	172	255	d	302	228	301	283	261	229	594	313	403	*							
682	LA RENTA-LA LIZAMA	43	2.367	2.463	2.479	2.823	2.280	3.087	2.726	2.628	3.039	2.624	2.780	3.309	3.407	298	8.144	1.716	3.568	910	486	1.464
710	SAN GIL-MOGOTES	33	653	729	672	642	656	596	661	783	683	690	718	943	869	71	1.102	376	711	13	1	1
725	LA RENTA-LEBRUJA	24	3.362	3.183	3.186	3.324	2.682	3.770	3.355	3.127	3.829	2.999	3.307	3.946	4.031	334	9.028	2.016	4.235	750	389	1.638
741	LEBRUJA-TE DE AEROPUERTO	3	5.742	5.672	6.077	6.424	3.746	7.146	6.505	6.160	6.126	5.643	7.162	6.991	7.801	626	14.904	4.682	5.652	1.051	796	2.253
886	20 DE JULIO - SAN ALBERTO	10	1.918	2.000	1.937	1.427	1.307	1.860	1.669	1.768	2.164	2.755	4.071	2.910	3.314	546	10.840	1.557	2.324	1.372	1.231	3.356
1000	PALENQUE-TE DE CAFE MADRID	12	12.072	12.958	18.379	7.941	6.433	21.732	18.512	21.107	18.526	19.138	24.050	24.083	24.337	2.925	37.461	10.616	16.392	3.495	1.857	4.901
1018	GIRON-BUCARAMANGA	9	32.462	39.568	29.291	29.345	28.442	31.818	28.336	27.237	29.145	25.055	31.412	28.543	22.877	2.958	20.822	10.671	5.773	1.672	711	1.795
1113	PUERTO ARAUJO-LA LIZAMA	87	1.979	2.044	1.753	1.985	1.947	2.532	2.107	1.955	2.639	3.246	2.754	3.159	3.984	940	14.446	1.940	3.376	2.118	1.672	5.340
1124	FLORIDABLANCA-PALENQUE	11	8.606	9.798	10.414	10.888	9.635	10.323	10.620	12.987	10.604	15.199	16.610	18.338	19.161	1.158	24.095	5.215	11.038	3.616	1.349	2.879
1140	LA LIZAMA-VEINTE DE JULIO	61	1.724	1.601	1.554	1.636	1.530	2.198	1.879	2.376	2.559	2.173	3.498	3.288	4.428	467	15.492	1.290	4.730	3.140	1.507	4.825
1143	LA LIZAMA-BARRANCABERMEJA	30	1.875	2.090	1.735	1.928	1.543	2.298	1.893	2.045	1.932	1.900	2.261	2.550	3.037	301	5.788	720	2.461	1.063	517	1.027
1150	PUERTO ARAUJO-PUERTO SERVIEZ	73		2.104	1.705	1.859	1.931	3.068	2.004	1.860	2.191	2.565	2.327	2.642	3.741	455	14.000	2.155	2.771	1.586	2.049	5.439
1189	MOGOTES - SAN JOAQUIN	29		31-6-83	32-7-81	36-6-58	27-6-67				57	70	104	55	77	19	3	3	0	0	0	0
											63-8-29						100,00%	0,00%	0,00%	0,00%	0,00%	

* NO SE INSTALO ESTACION DE CONTEO



Anexo B. Espectro de carga

Anexo B.1 Espectro de carga-eje simple

CARGA EJE SIMPLE	Ejes por 1000 camiones-TPDS					
	0 - 500	500 - 1.000	1.000 - 2.500	2.500 - 5.000	5.000 - 10.000	Mayor a 10.000
19.6	43	42				
21.56			39	37		
23.52					38	41
29.4	369	367				
32.34			334	320		
35.28					330	352
39.2	404	403				
43.12			379	368		
47.04					376	393
49	226	229				
53.9			257	268		
58.8					260	241
58.8	119	119				
64.68			132	138		
70.56					133	124
68.6	73	72				
75.46			69	67		
82.32					68	71
78.4	69	69				
86.24			63	61		
94.08					62	66
88.2	84	84				
97.02			76	73		
105.84					75	80
98	83	83				
107.8			76	72		
117.6					74	80
107.8	116	116				
118.58			105	101		
129.36					104	111
117.6	121	121				
129.36			110	105		

141.12					108	116
127.4	103	102				
140.14			93	89		
152.88					92	98
137.2	50	50				
150.92			46	44		
164.64					45	48
147	19	19				
161.7			17	17		
176.4					17	18

Anexo B.2 Espectro de carga-eje tandem

CARGA EJE TANDEM	Ejes por 1000 camiones-TPDS					
	0 - 500	500 - 1.000	1.000 - 2.500	2.500 - 5.000	5.000 - 10.000	Mayor a 10.000
KN						
58.8	2	2				
64.68			4	5		
70.56					5	3
78.4	8	8				
86.24			17	21		
94.08					19	13
98	6	6				
107.8			11	14		
117.6					13	9
117.6	6	6				
129.36			12	16		
141.12					14	10
137.2	7	7				
150.92			13	16		
164.64					15	11
156.8	7	7				
172.48			13	17		
188.16					15	11
176.4	9	9				
194.04			18	23		
211.68					21	15
196	14	15				

215.6			30	38		
235.2					34	24
215.6	27	29				
237.16			55	67		
258.72					60	42
235.2	20	21				
258.72			37	44		
282.24					39	28
254.8	6	7				
280.28			12	14		
305.76					12	8
274.4	1	2				
301.84			4	5		
329.28					4	2
294	1	1				
323.4			2	2		
352.8					2	1

Anexo B.3 Espectro de carga-eje tandem

CARGA EJE TRIDEM	Ejes por 1000 camiones-TPDS						
	KN	0 - 500	500 - 1.000	1.000 - 2.500	2.500 - 5.000	5.000 - 10.000	Mayor a 10.000
58.8		0	1				
64.68				1	2		
70.56						1	1
78.4		1	1				
86.24				2	3		
94.08						2	1
98		0	1				
107.8				1	1		
117.6						1	1
117.6		0	0				
129.36				1	1		
141.12						1	1
137.2		0	0				
150.92				1	1		
164.64						1	1

156.8	1	1				
172.48			2	2		
188.16					2	1
176.4	1	1				
194.04			2	2		
211.68					2	1
196	1	1				
215.6			3	3		
235.2					3	2
215.6	1	2				
237.16			5	5		
258.72					4	3
235.2	4	5				
258.72			11	13		
282.24					11	6
254.8	5	7				
280.28			15	17		
305.76					14	8
274.4	3	4				
301.84			9	10		
329.28					8	5
294	2	2				
323.4			5	6		
352.8					5	3
313.6	1	1				
344.96			3	3		
376.32					3	1
333.2	1	1				
366.52			2	3		
399.84					2	1
352.8	1	1				
388.08			2	2		
423.36					2	1
372.4	1	1				
409.64			2	2		
446.88					2	1

Anexo C. Código fuente

Anexo C.1 Información general

Anexo C.1.1 Formularios

Marco

```
Imports System.IO
Imports System.Xml.Serialization
Public Class Marco
    Public serializadorProyecto As New XmlSerializer(glInformacionProyecto.GetType)
    Public streamProyecto As Stream
    Private Sub Principal_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Intro.MdiParent = Me
        Intro.Show()
        InformacionPorDefecto()
        NuevoToolStripMenuItem.Enabled = True
        AbrirToolStripMenuItem.Enabled = True
        GuardarToolStripMenuItem.Enabled = False
        CerrarToolStripMenuItem.Enabled = False
        SalirToolStripMenuItem1.Enabled = True
        EdicionToolStripMenuItem.Enabled = False
    End Sub
    Private Sub SalirToolStripMenuItem1_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles SalirToolStripMenuItem1.Click
        Me.Close()
    End Sub
    Private Sub AyudaToolStripMenuItem4_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles AyudaToolStripMenuItem4.Click
        Process.Start(Ayuda.chm)
    End Sub
    Private Sub AcercaDeEsteProgramaToolStripMenuItem1_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
AcercaDeEsteProgramaToolStripMenuItem1.Click
        AcercDe.Show()
    End Sub
    Private Sub NuevoToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles NuevoToolStripMenuItem.Click
        Intro.Close()
        General.Show()
    End Sub
    Private Sub AbrirToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles AbrirToolStripMenuItem.Click
        If openProyecto.ShowDialog() = System.Windows.Forms.DialogResult.OK Then
            Try
                streamProyecto = openProyecto.OpenFile()
                Dim streamR As New StreamReader(streamProyecto)
                glInformacionProyecto = serializadorProyecto.Deserialize(streamR)
                streamR.Close()

                glInformacionGeneral = glInformacionProyecto.General
                glInformacionTrafico = glInformacionProyecto.Trafico
                glInformacionEstructura = glInformacionProyecto.Estructura
            End Try
        End If
    End Sub
End Class
```

```

        If (streamProyecto IsNot Nothing) Then
            streamProyecto.Close()
        End If
    Catch Ex As Exception
        MessageBox.Show("No se puede leer el archivo. Causa: " & Ex.Message)
    Finally
        If (streamProyecto IsNot Nothing) Then
            streamProyecto.Close()
        End If
    End Try
End If
End Sub
Private Sub ReiniciarToolStripMenuItem_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ReiniciarToolStripMenuItem.Click
    General.Close()
    General.Show()
End Sub
End Class

```

Límites

```

Public Class Limites
    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub
    Private Sub checkIRI_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles checkIRI.CheckedChanged
        If checkIRI.Checked = True Then
            txtConfIRI.Enabled = True
            txtLimiteIRI.Enabled = True
        Else
            txtConfIRI.Enabled = False
            txtLimiteIRI.Enabled = False
        End If
    End Sub
    Private Sub checkLong_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles checkLong.CheckedChanged
        If checkLong.Checked = True Then
            txtConfLong.Enabled = True
            txtLimiteLong.Enabled = True
        Else
            txtConfLong.Enabled = False
            txtLimiteLong.Enabled = False
        End If
    End Sub
    Private Sub checkCocodrilo_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles checkCocodrilo.CheckedChanged
        If checkCocodrilo.Checked = True Then
            txtConfCocodrilo.Enabled = True
            txtLimiteCocodrilo.Enabled = True
        Else
            txtConfCocodrilo.Enabled = False
            txtLimiteCocodrilo.Enabled = False
        End If
    End Sub
End Class

```

```

    Private Sub checkTemp_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles checkTemp.CheckedChanged
        If checkTemp.Checked = True Then
            txtConfTemp.Enabled = True
            txtLimiteTemp.Enabled = True
        Else
            txtConfTemp.Enabled = False
            txtLimiteTemp.Enabled = False
        End If
    End Sub
    Private Sub checkDeformacion_CheckedChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles checkDeformacion.CheckedChanged
        If checkDeformacion.Checked = True Then
            txtConfDeformacion.Enabled = True
            txtLimiteDeformacion.Enabled = True
        Else
            txtConfDeformacion.Enabled = False
            txtLimiteDeformacion.Enabled = False
        End If
    End Sub
    Private Sub checkDeformacionCA_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles checkDeformacionCA.CheckedChanged
        If checkDeformacionCA.Checked = True Then
            txtConfDeformacionCA.Enabled = True
            txtLimiteDeformacionCA.Enabled = True
        Else
            txtConfDeformacionCA.Enabled = False
            txtLimiteDeformacionCA.Enabled = False
        End If
    End Sub
    Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
        Me.Close()
    End Sub
    Private Sub Limites_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        If General.rbtnFlexible.Checked = True Then
            grpFlexible.Visible = True
            grpRigido.Visible = False
        ElseIf General.rbtnRigido.Checked = True Then
            grpFlexible.Visible = False
            grpRigido.Visible = True
        End If
    End Sub
End Class

```

Acerca de

```

Public Class AcercDe
    Private Sub btnVolver_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnVolver.Click
        Me.Close()
    End Sub
End Class

```

Intro

```
Public Class Intro
    Private Sub btnContinuar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnContinuar.Click
        Me.Close()
        General.Show()
    End Sub
End Class
```

General

```
Imports System.IO
Imports System.Xml.Serialization
Public Class General
    'GUARDAR Y CARGAR
    Public srlzGeneral As New XmlSerializer(glInformacionGeneral.GetType)
    Public strmGeneral As Stream
    Private Sub RecopilarInformacionGeneral()
        With glInformacionGeneral
            .Departamento = txtDepartamento.Text
            .EstacionFinal = mtxtEstacionFin.Text
            .EstacionInicial = mtxtEstacionIni.Text
            .FechaApertTrafico = dtpApertTraf.Text
            .FechaConstrBase = dtpConstrBase.Text
            .FechaConstrPav = dtpConstrPav.Text
            .Municipio = txtMunicipio.Text
            .Descripcion = txtDescripcion.Text
            If rbtnNuevoPav.Checked = True Then
                .TipoAnalisis = "Nuevo"
            Else
                .TipoAnalisis = "Sobrecarpeta"
            End If
            If rbtnFlexible.Checked = True Then
                .TipoPavimento = "Flexible"
            Else
                .TipoPavimento = "Rigido"
            End If
            .VidaDiseño = txtVida.Text
        End With
    End Sub
    Private Sub btnGuardarInfoGeneral_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnGuardarInfoGeneral.Click
        'EXTRAER INFO A CONTROLES
        RecopilarInformacionGeneral()
        If saveGeneral.ShowDialog() = DialogResult.OK Then
            strmGeneral = saveGeneral.OpenFile()
            Using strmW As StreamWriter = New StreamWriter(strmGeneral)
                srlzGeneral.Serialize(strmW, glInformacionGeneral)
                strmW.Close()
            End Using
            If (strmGeneral IsNot Nothing) Then
                strmGeneral.Close()
            End If
        End If
    End Sub
End Class
```

```

Private Sub btnCargarInfoGeneral_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles btnCargarInfoGeneral.Click
    If openGeneral.ShowDialog() = System.Windows.Forms.DialogResult.OK Then
        Try
            strmGeneral = openGeneral.OpenFile()
            Dim strmR As New StreamReader(strmGeneral)
            glInformacionGeneral = srlzGeneral.Deserialize(strmR)
            strmR.Close()
            'ASIGNAR INFO A CONTROLES
            With glInformacionGeneral
                txtDepartamento.Text = .Departamento
                mtxtEstacionFin.Text = .EstacionFinal
                mtxtEstacionIni.Text = .EstacionInicial
                dtpApertTraf.Text = .FechaApertTrafico
                dtpConstrBase.Text = .FechaConstrBase
                dtpConstrPav.Text = .FechaConstrPav
                txtMunicipio.Text = .Municipio
                If .TipoAnalisis = "Nuevo" Then
                    rbtnNuevoPav.Checked = True
                Else
                    rbtnSobrecarpeta.Checked = True
                End If
                If .TipoPavimento = "Flexible" Then
                    rbtnFlexible.Checked = True
                Else
                    rbtnRigido.Checked = True
                End If
                txtDescripcion.Text = .Descripcion
                txtVida.Text = .VidaDiseño
            End With
            If (strmGeneral IsNot Nothing) Then
                strmGeneral.Close()
            End If
        Catch Ex As Exception
            MessageBox.Show("No se puede leer el archivo. Causa: " & Ex.Message)
        Finally
            If (strmGeneral IsNot Nothing) Then
                strmGeneral.Close()
            End If
        End Try
    End If
End Sub
'INICIAL
Private Sub General_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    Me.MdiParent = Marco
    With glInformacionGeneral
        txtDepartamento.Text = .Departamento
        mtxtEstacionFin.Text = .EstacionFinal
        mtxtEstacionIni.Text = .EstacionInicial
        dtpApertTraf.Text = .FechaApertTrafico
        dtpConstrBase.Text = .FechaConstrBase
        dtpConstrPav.Text = .FechaConstrPav
        txtMunicipio.Text = .Municipio
    End With

```

```

        If .TipoAnálisis = "Nuevo" Then
            rbtnNuevoPav.Checked = True
        Else
            rbtnSobrecarpeta.Checked = True
        End If
        If .TipoPavimento = "Flexible" Then
            rbtnFlexible.Checked = True
        Else
            rbtnRigido.Checked = True
        End If
        txtDescripcion.Text = .Descripcion
        txtVida.Text = .VidaDiseño
    End With
End Sub
'Abrir variables de entrada
Private Sub btnTráfico_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnTráfico.Click
    RecopilarInformaciónGeneral()
    Tráfico.Show()
End Sub
Private Sub btnEstructura_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEstructura.Click
    RecopilarInformaciónGeneral()
    Estructura.Show()
End Sub
Private Sub btnLímites_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLímites.Click
    Límites.Show()
End Sub
'Validaciones
Private Sub txtVida_Validating(ByVal sender As Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles txtVida.Validating
    If IsNothing(txtVida) = False Then
        Try
            glInformaciónGeneral.VidaDiseño = CInt(txtVida.Text)
        Catch ex As Exception
            If txtVida.Text = "" Then
                e.Cancel = False
            Else
                MsgBox("El valor ingresado debe ser un número. Verifique el
valor e ingréselo de nuevo.", vbInformation, "Valor inválido")
                e.Cancel = True
            End If
        End Try
    End If
End Sub
Private Sub dtpConstrPav_Validating(ByVal sender As Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles dtpConstrPav.Validating
    If IsNothing(dtpConstrPav) = False Then
        Try
            glInformaciónGeneral.FechaConstrPav = CDate(dtpConstrPav.Text)
        Catch ex As Exception
            If dtpConstrPav.Text = "" Then
                e.Cancel = False
            End If
        End Try
    End If
End Sub

```

```

        Else
            MsgBox("La fecha ingresada no es válida. Verifique la fecha e
ingrésela de nuevo.", vbInformation, "Fecha inválida")
            e.Cancel = True
        End If
    End Try
End If
End Sub
Private Sub dtpApertTraf_Validating(ByVal sender As Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles dtpApertTraf.Validating
    If IsNothing(dtpApertTraf) = False Then
        Try
            glInformacionGeneral.FechaApertTrafico = CDate(dtpApertTraf.Text)
        Catch ex As Exception
            If dtpApertTraf.Text = "" Then
                e.Cancel = False
            Else
                MsgBox("La fecha ingresada no es válida. Verifique la fecha e
ingrésela de nuevo.", vbInformation, "Fecha inválida")
                e.Cancel = True
            End If
        End Try
    End If
End Sub
Private Sub dtpConstrBase_Validating(ByVal sender As Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles dtpConstrBase.Validating
    If IsNothing(dtpConstrBase) = False Then
        Try
            glInformacionGeneral.FechaConstrBase = CDate(dtpConstrBase.Text)
        Catch ex As Exception
            If dtpConstrBase.Text = "" Then
                e.Cancel = False
            Else
                MsgBox("La fecha ingresada no es válida. Verifique la fecha e
ingrésela de nuevo.", vbInformation, "Fecha inválida")
                e.Cancel = True
            End If
        End Try
    End If
End Sub
Private Sub rbtnNuevoPav_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles rbtnNuevoPav.CheckedChanged,
rbtnFlexible.CheckedChanged
    If rbtnNuevoPav.Checked = True Then
        glInformacionGeneral.TipoAnálisis = "Nuevo"
    End If
End Sub
Private Sub rbtnSobrecarpeta_CheckedChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles rbtnSobrecarpeta.CheckedChanged,
rbtnRigido.CheckedChanged
    If rbtnSobrecarpeta.Checked = True Then
        glInformacionGeneral.TipoAnálisis = "Sobrecarpeta"
    End If
End Sub

```

```

    Private Sub rbtnFlexible_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs)
        If rbtnFlexible.Checked = True Then
            glInformacionGeneral.TipoPavimento = "Flexible"
        End If
    End Sub
    Private Sub rbtnRigido_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs)
        If rbtnRigido.Checked = True Then
            glInformacionGeneral.TipoPavimento = "Rígido"
        End If
    End Sub
    Private Sub txtDescripcion_Validating(ByVal sender As Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles txtDescripcion.Validating
        If IsNothing(txtDescripcion) = True Then
            Try
                glInformacionGeneral.VidaDiseño = txtDescripcion.Text
            Catch ex As Exception
                If txtDescripcion.Text = "" Then
                    e.Cancel = False
                Else
                    e.Cancel = True
                End If
            End Try
        End If
    End Sub
    Private Sub txtDepartamento_Validating(ByVal sender As Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles txtDepartamento.Validating
        If IsNothing(txtDepartamento) = False Then
            Try
                glInformacionGeneral.Departamento = txtDepartamento.Text
            Catch ex As Exception
                If txtDepartamento.Text = "" Then
                    e.Cancel = False
                Else
                    e.Cancel = True
                End If
            End Try
        End If
    End Sub
    Private Sub txtMunicipio_Validating(ByVal sender As Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles txtMunicipio.Validating
        If IsNothing(txtMunicipio) = False Then
            Try
                glInformacionGeneral.Municipio = txtMunicipio.Text
            Catch ex As Exception
                If txtDepartamento.Text = "" Then
                    e.Cancel = False
                Else
                    e.Cancel = True
                End If
            End Try
        End If
    End Sub

```

```

Private Sub mtxtEstacionIni_Validating(ByVal sender As Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles mtxtEstacionIni.Validating
    If IsNothing(mtxtEstacionIni) = False Then
        Try
            glInformacionGeneral.EstacionInicial = mtxtEstacionIni.Text
        Catch ex As Exception
            If mtxtEstacionIni.Text = "" Then
                e.Cancel = False
            Else
                e.Cancel = True
            End If
        End Try
    End If
End Sub
Private Sub mtxtEstacionFin_Validating(ByVal sender As Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles mtxtEstacionFin.Validating
    If IsNothing(mtxtEstacionFin) = False Then
        Try
            glInformacionGeneral.EstacionFinal = mtxtEstacionFin.Text
        Catch ex As Exception
            If mtxtEstacionFin.Text = "" Then
                e.Cancel = False
            Else
                e.Cancel = True
            End If
        End Try
    End If
End Sub
End Class

```

Anexo C.1.2 Clases

Información del proyecto

```

Public Class ClaseProyecto
    Private miGeneral As ClaseGeneral
    Private miTrafico As ClaseTrafico
    Private miEstructura As ClaseEstructura
    Public Property General() As ClaseGeneral
        Get
            General = miGeneral
        End Get
        Set(ByVal Value As ClaseGeneral)
            miGeneral = Value
        End Set
    End Property
    Public Property Trafico() As ClaseTrafico
        Get
            Trafico = miTrafico
        End Get
        Set(ByVal Value As ClaseTrafico)
            miTrafico = Value
        End Set
    End Property

```

```

Public Property Estructura() As ClaseEstructura
    Get
        Estructura = miEstructura
    End Get

    Set(ByVal Value As ClaseEstructura)
        miEstructura = Value
    End Set
End Property
End Class

```

Información general

```

Public Class ClaseGeneral
    Private miVidaDiseño As Double
    Private miFechaConstrPav As Date
    Private miFechaConstrBase As Date
    Private miFechaApertTrafico As Date
    Private miTipoAnálisis As String
    Private miTipoPavimento As String
    Private miDescripcion As String
    Private miDepartamento As String
    Private miMunicipio As String
    Private miEstIni As String
    Private miEstFin As String
    Public Property VidaDiseño As Double
        Get
            VidaDiseño = miVidaDiseño
        End Get
        Set(ByVal Value As Double)
            miVidaDiseño = Value
        End Set
    End Property
    Public Property FechaConstrPav As Date
        Get
            FechaConstrPav = miFechaConstrPav
        End Get
        Set(ByVal Value As Date)
            miFechaConstrPav = Value
        End Set
    End Property
    Public Property FechaConstrBase As Date
        Get
            FechaConstrBase = miFechaConstrBase
        End Get
        Set(ByVal Value As Date)
            miFechaConstrBase = Value
        End Set
    End Property
    Public Property FechaApertTrafico As Date
        Get
            FechaApertTrafico = miFechaApertTrafico
        End Get

```

```

        Set(ByVal Value As Date)
            miFechaApertTrafico = Value
        End Set
    End Property
    Public Property TipoAnalisis As String
        Get
            TipoAnalisis = miTipoAnalisis
        End Get
        Set(ByVal Value As String)
            miTipoAnalisis = Value
        End Set
    End Property
    Public Property TipoPavimento As String
        Get
            TipoPavimento = miTipoPavimento
        End Get
        Set(ByVal Value As String)
            miTipoPavimento = Value
        End Set
    End Property
    Public Property Descripcion As String
        Get
            Descripcion = miDescripcion
        End Get
        Set(ByVal Value As String)
            miDescripcion = Value
        End Set
    End Property
    Public Property Departamento As String
        Get
            Departamento = miDepartamento
        End Get
        Set(ByVal Value As String)
            miDepartamento = Value
        End Set
    End Property
    Public Property Municipio As String
        Get
            Municipio = miMunicipio
        End Get
        Set(ByVal Value As String)
            miMunicipio = Value
        End Set
    End Property
    Public Property EstacionInicial As String
        Get
            EstacionInicial = miEstIni
        End Get
        Set(ByVal Value As String)
            miEstIni = Value
        End Set
    End Property

```

```

Public Property EstacionFinal As String
    Get
        EstacionFinal = miEstFin
    End Get
    Set(ByVal Value As String)
        miEstFin = Value
    End Set
End Property
End Class

```

Anexo C.1.3 Módulos

Variables Globales

```

Module ModuloVariablesGlobales
    Public glInformacionProyecto As New ClaseProyecto
    Public glInformacionGeneral As New ClaseGeneral
    Public glInformacionTrafico As New ClaseTrafico
    Public glInformacionEstructura As New ClaseEstructura
    'Variables adicionales de tráfico
    Public glConfigEje As New ClaseConfiguracionEje
    Public glBatalla As New ClaseBatalla
    Public glDistrClase As New ClaseDistrClase
    Public glDistrHoraria As New ClaseDistrHoraria
    Public glFacMensuales As New ClaseFactMensuales
    Public glFacCrecimiento As New ClaseFactCrecimiento
    Public glEspectrosDeCarga As New ClaseEspectroCarga
    'DataTables y DatSets iniciales para los espectros
    Public EspectroSimpleGeneralT1 As New DataTable
    Public EspectroSimpleGeneralT2 As New DataTable
    Public EspectroSimpleGeneralT3 As New DataTable
    Public EspectroSimpleGeneralT4 As New DataTable
    Public EspectroSimpleGeneralT5 As New DataTable
    Public EspectroSimpleGeneralT6 As New DataTable
    Public EspectroSimplePorClaseT1 As New DataTable
    Public EspectroSimplePorClaseT2 As New DataTable
    Public EspectroSimplePorClaseT3 As New DataTable
    Public EspectroSimplePorClaseT4 As New DataTable
    Public EspectroSimplePorClaseT5 As New DataTable
    Public EspectroSimplePorClaseT6 As New DataTable
    Public EspectroTandemGeneralT1 As New DataTable
    Public EspectroTandemGeneralT2 As New DataTable
    Public EspectroTandemGeneralT3 As New DataTable
    Public EspectroTandemGeneralT4 As New DataTable
    Public EspectroTandemGeneralT5 As New DataTable
    Public EspectroTandemGeneralT6 As New DataTable
    Public EspectroTandemPorClaseT1 As New DataTable
    Public EspectroTandemPorClaseT2 As New DataTable
    Public EspectroTandemPorClaseT3 As New DataTable
    Public EspectroTandemPorClaseT4 As New DataTable
    Public EspectroTandemPorClaseT5 As New DataTable
    Public EspectroTandemPorClaseT6 As New DataTable
    Public EspectroTridemGeneralT1 As New DataTable
    Public EspectroTridemGeneralT2 As New DataTable
    Public EspectroTridemGeneralT3 As New DataTable
    Public EspectroTridemGeneralT4 As New DataTable

```

```

Public EspectroTridemGeneralT5 As New DataTable
Public EspectroTridemGeneralT6 As New DataTable
Public EspectroTridemPorClaseT1 As New DataTable
Public EspectroTridemPorClaseT2 As New DataTable
Public EspectroTridemPorClaseT3 As New DataTable
Public EspectroTridemPorClaseT4 As New DataTable
Public EspectroTridemPorClaseT5 As New DataTable
Public EspectroTridemPorClaseT6 As New DataTable
Public dsSimpleGeneral As New DataSet
Public dsTandemGeneral As New DataSet
Public dsTridemGeneral As New DataSet
Public dsSimplePorClase As New DataSet
Public dsTandemPorClase As New DataSet
Public dsTridemPorClase As New DataSet
'Función multiplicar matrices
Public Function MultiplicarMatrices(ByVal MatrizA As Double(,), ByVal MatrizB As
Double(,)) As Double(,)
    Dim MatrizX(UBound(MatrizA, 1), UBound(MatrizB, 2)) As Double
    Dim fil1 = UBound(MatrizA, 1)
    Dim col1 = UBound(MatrizA, 2)
    Dim fil2 = UBound(MatrizB, 1)
    Dim col2 = UBound(MatrizB, 2)
    Dim sum As Double
    If col1 = fil2 Then
        For i = 0 To fil1
            For j = 0 To col2
                sum = 0
                For k = 0 To col1
                    sum += MatrizA(i, k) * MatrizB(k, j)
                Next
                MatrizX.SetValue(sum, i, j)
            Next
        Next
        Return MatrizX
    Else
        Return MatrizX
    End If
End Function
'Función Transponer
Public Function TransponerMatriz(ByVal Matriz As Double(,)) As Double(,)
    Dim MatrizTranspuesta(UBound(Matriz, 2), UBound(Matriz, 1)) As Double
    For i = 0 To UBound(Matriz, 2)
        For j = 0 To UBound(Matriz, 1)
            MatrizTranspuesta(i, j) = Matriz(j, i)
        Next
    Next
    Return MatrizTranspuesta
End Function
'Función para pasar datatable a una matriz de tipo double(,) de 2 dimensiones
Public Function DataTable2Matriz(ByVal Tabla As DataTable) As Double(,)
    Dim Matriz(,) As Double
    If IsNumeric(Tabla.Rows(0).Item(0)) Then
        ReDim Matriz(Tabla.Rows.Count - 1, Tabla.Columns.Count - 1)
        For i = 0 To Tabla.Rows.Count - 1
            For j = 0 To Tabla.Columns.Count - 1

```

```

        Matriz(i, j) = Cdbl(Tabla.Rows(i).Item(j).ToString)
    Next
Next
Else
    ReDim Matriz(Tabla.Rows.Count - 1, Tabla.Columns.Count - 2)
    For i = 0 To Tabla.Rows.Count - 1
        For j = 0 To Tabla.Columns.Count - 2
            Matriz(i, j) = Cdbl(Tabla.Rows(i).Item(j + 1).ToString)
        Next
    Next
End If
Return Matriz
End Function
'Función para pasar datatable de una sola dimensión a vector de tipo double ()
Public Function DataTable2Vector(ByVal Tabla As DataTable) As Double()
    Dim Vector(Tabla.Rows.Count - 1) As Double
    For i = 0 To Tabla.Rows.Count - 1
        Vector(i) = Cdbl(Tabla.Rows(i).Item(0).ToString)
    Next
    Return Vector
End Function
'Extraer Fila de datatable y guardarla en vector tipo double()
Public Function ExtraerFilaDataTable(ByVal Tabla As DataTable, ByVal Fila As
Integer) As Double()
    Dim Vector() As Double
    If IsNumeric(Tabla.Rows(0).Item(0)) Then
        ReDim Vector(Tabla.Columns.Count - 1)
        For i = 0 To Tabla.Columns.Count - 1
            Vector(i) = Cdbl(Tabla.Rows(Fila).Item(i).ToString)
        Next
    Else
        ReDim Vector(Tabla.Columns.Count - 2)
        For i = 0 To Tabla.Columns.Count - 2
            Vector(i) = Cdbl(Tabla.Rows(Fila).Item(i + 1).ToString)
        Next
    End If
    Return Vector
End Function
'Extraer Columna de datatable y guardarla en vector tipo double()
Public Function ExtraerColumnaDataTable(ByVal Tabla As DataTable, ByVal Columna
As Integer) As Double()
    Dim Vector(Tabla.Rows.Count - 1) As Double
    If IsNumeric(Tabla.Rows(0).Item(0)) Then
        For i = 0 To Tabla.Rows.Count - 1
            Vector(i) = Cdbl(Tabla.Rows(i).Item(Columna).ToString)
        Next
    Else
        For i = 0 To Tabla.Rows.Count - 1
            Vector(i) = Cdbl(Tabla.Rows(i).Item(Columna + 1).ToString)
        Next
    End If
    Return Vector
End Function

```

```

'Valores por defecto
Public Sub InformacionPorDefecto()
    'INFORMACIÓN GENERAL POR DEFECTO
    With glInformacionGeneral
        .EstacionInicial = "K 00+000"
        .EstacionFinal = "K 10+000"
        .Departamento = "Santander"
        .Municipio = "Bucaramanga"
        .TipoAnálisis = "Nuevo"
        .TipoPavimento = "Flexible"
        .Descripcion = ""
        .VidaDiseño = 10
    End With
    'INFORMACIÓN DE TRÁFICO POR DEFECTO
    With glInformacionTrafico
        'TPD Y FACTORES
        .TPDAT = 10000
        .NumCarril = 2
        .FactorDir = 50
        .FactorCarril = 90
        .VelOper = 55
        'VOLUMEN DE TRAFICO
        glDistrClase.Porcentajes = {58.34, 8.09, 19.03, 7.75, 0.0, 2.53, 4.26}
        .DistrClase = glDistrClase
        glDistrHoraria.Porcentajes = {0.91, 1.1, 0.96, 1.07, 1.92, 5.27, 4.5,
5.12, 5.33, 5.33, 4.66, 5.25, 4.65, 4.76, 4.67, 3.11, 4.87, 4.9, 5.96, 8.26, 6.9,
5.91, 2.83, 1.76}
        .DistrHoraria = glDistrHoraria
        Dim tablaDistrMes As New DataTable
        With tablaDistrMes
            .Columns.Add("MES")
            .Columns.Add("BUS")
            .Columns.Add("C2P")
            .Columns.Add("C2G")
            .Columns.Add("C3")
            .Columns.Add("C4")
            .Columns.Add("C5")
            .Columns.Add(">C5")
            .Rows.Add("ENERO", 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0)
            .Rows.Add("FEBRERO", 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0)
            .Rows.Add("MARZO", 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0)
            .Rows.Add("ABRIL", 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0)
            .Rows.Add("MAYO", 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0)
            .Rows.Add("JUNIO", 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0)
            .Rows.Add("JULIO", 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0)
            .Rows.Add("AGOSTO", 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0)
            .Rows.Add("SEPTIEMBRE", 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0)
            .Rows.Add("OCTUBRE", 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0)
            .Rows.Add("NOVIEMBRE", 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0)
            .Rows.Add("DICIEMBRE", 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0)
        End With
        glFacMensuales.Factores = tablaDistrMes
        .FactorMeses = glFacMensuales
        'ESPECTROS DE CARGA
        'DataTables
    End With
End Sub

```

```

'Simple/general
With EspectroSimpleGeneralT1
.Columns.Add("General")
.Rows.Add(2.29)
.Rows.Add(19.64)
.Rows.Add(21.5)
.Rows.Add(12.03)
.Rows.Add(6.33)
.Rows.Add(3.89)
.Rows.Add(3.67)
.Rows.Add(4.47)
.Rows.Add(4.42)
.Rows.Add(6.17)
.Rows.Add(6.44)
.Rows.Add(5.48)
.Rows.Add(2.66)
.Rows.Add(1.01)
.Rows.Add(0)
.Rows.Add(0)
.Rows.Add(0)
.Rows.Add(0)
End With
With EspectroSimpleGeneralT2
.Columns.Add("General")
.Rows.Add(2.24)
.Rows.Add(19.56)
.Rows.Add(21.48)
.Rows.Add(12.21)
.Rows.Add(6.34)
.Rows.Add(3.84)
.Rows.Add(3.68)
.Rows.Add(4.48)
.Rows.Add(4.42)
.Rows.Add(6.18)
.Rows.Add(6.45)
.Rows.Add(5.44)
.Rows.Add(2.67)
.Rows.Add(1.01)
.Rows.Add(0)
.Rows.Add(0)
.Rows.Add(0)
.Rows.Add(0)
End With
With EspectroSimpleGeneralT3
.Columns.Add("General")
.Rows.Add(0)
.Rows.Add(2.17)
.Rows.Add(18.6)
.Rows.Add(21.1)
.Rows.Add(14.31)
.Rows.Add(7.35)
.Rows.Add(3.84)
.Rows.Add(3.51)
.Rows.Add(4.23)
.Rows.Add(4.23)

```

```

        .Rows.Add(5.85)
        .Rows.Add(6.12)
        .Rows.Add(0)
        .Rows.Add(5.18)
        .Rows.Add(2.56)
        .Rows.Add(0.95)
        .Rows.Add(0)
        .Rows.Add(0)
End With
With EspectroSimpleGeneralT4
    .Columns.Add("General")
    .Rows.Add(0)
    .Rows.Add(2.1)
    .Rows.Add(18.18)
    .Rows.Add(20.91)
    .Rows.Add(15.23)
    .Rows.Add(7.84)
    .Rows.Add(3.81)
    .Rows.Add(3.47)
    .Rows.Add(4.15)
    .Rows.Add(4.09)
    .Rows.Add(5.74)
    .Rows.Add(5.97)
    .Rows.Add(0)
    .Rows.Add(5.06)
    .Rows.Add(2.48)
    .Rows.Add(0.97)
    .Rows.Add(0)
    .Rows.Add(0)
End With
With EspectroSimpleGeneralT5
    .Columns.Add("General")
    .Rows.Add(0)
    .Rows.Add(2.13)
    .Rows.Add(18.52)
    .Rows.Add(21.1)
    .Rows.Add(14.59)
    .Rows.Add(0)
    .Rows.Add(7.46)
    .Rows.Add(3.82)
    .Rows.Add(3.48)
    .Rows.Add(4.21)
    .Rows.Add(4.15)
    .Rows.Add(5.84)
    .Rows.Add(0)
    .Rows.Add(6.06)
    .Rows.Add(5.16)
    .Rows.Add(2.53)
    .Rows.Add(0.95)
    .Rows.Add(0)
End With
With EspectroSimpleGeneralT6
    .Columns.Add("General")
    .Rows.Add(0)
    .Rows.Add(2.23)

```

```

.Rows.Add(19.14)
.Rows.Add(21.37)
.Rows.Add(13.1)
.Rows.Add(0)
.Rows.Add(6.74)
.Rows.Add(3.86)
.Rows.Add(3.59)
.Rows.Add(4.35)
.Rows.Add(4.35)
.Rows.Add(6.04)
.Rows.Add(0)
.Rows.Add(6.31)
.Rows.Add(5.33)
.Rows.Add(2.61)
.Rows.Add(0.98)
.Rows.Add(0)
End With
'Simple/clase
With EspectroSimplePorClaseT1
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(2.29, 2.29, 2.29, 2.29, 2.29, 2.29, 2.29)
.Rows.Add(19.64, 19.64, 19.64, 19.64, 19.64, 19.64, 19.64)
.Rows.Add(21.5, 21.5, 21.5, 21.5, 21.5, 21.5, 21.5)
.Rows.Add(12.03, 12.03, 12.03, 12.03, 12.03, 12.03, 12.03)
.Rows.Add(6.33, 6.33, 6.33, 6.33, 6.33, 6.33, 6.33)
.Rows.Add(3.89, 3.89, 3.89, 3.89, 3.89, 3.89, 3.89)
.Rows.Add(3.67, 3.67, 3.67, 3.67, 3.67, 3.67, 3.67)
.Rows.Add(4.47, 4.47, 4.47, 4.47, 4.47, 4.47, 4.47)
.Rows.Add(4.42, 4.42, 4.42, 4.42, 4.42, 4.42, 4.42)
.Rows.Add(6.17, 6.17, 6.17, 6.17, 6.17, 6.17, 6.17)
.Rows.Add(6.44, 6.44, 6.44, 6.44, 6.44, 6.44, 6.44)
.Rows.Add(5.48, 5.48, 5.48, 5.48, 5.48, 5.48, 5.48)
.Rows.Add(2.66, 2.66, 2.66, 2.66, 2.66, 2.66, 2.66)
.Rows.Add(1.01, 1.01, 1.01, 1.01, 1.01, 1.01, 1.01)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
With EspectroSimplePorClaseT2
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(2.24, 2.24, 2.24, 2.24, 2.24, 2.24, 2.24)
.Rows.Add(19.56, 19.56, 19.56, 19.56, 19.56, 19.56, 19.56)

```

```

.Rows.Add(21.48, 21.48, 21.48, 21.48, 21.48, 21.48, 21.48)
.Rows.Add(12.21, 12.21, 12.21, 12.21, 12.21, 12.21, 12.21)
.Rows.Add(6.34, 6.34, 6.34, 6.34, 6.34, 6.34, 6.34)
.Rows.Add(3.84, 3.84, 3.84, 3.84, 3.84, 3.84, 3.84)
.Rows.Add(3.68, 3.68, 3.68, 3.68, 3.68, 3.68, 3.68)
.Rows.Add(4.48, 4.48, 4.48, 4.48, 4.48, 4.48, 4.48)
.Rows.Add(4.42, 4.42, 4.42, 4.42, 4.42, 4.42, 4.42)
.Rows.Add(6.18, 6.18, 6.18, 6.18, 6.18, 6.18, 6.18)
.Rows.Add(6.45, 6.45, 6.45, 6.45, 6.45, 6.45, 6.45)
.Rows.Add(5.44, 5.44, 5.44, 5.44, 5.44, 5.44, 5.44)
.Rows.Add(2.67, 2.67, 2.67, 2.67, 2.67, 2.67, 2.67)
.Rows.Add(1.01, 1.01, 1.01, 1.01, 1.01, 1.01, 1.01)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
With EspectroSimplePorClaseT3
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(2.17, 2.17, 2.17, 2.17, 2.17, 2.17, 2.17)
.Rows.Add(18.6, 18.6, 18.6, 18.6, 18.6, 18.6, 18.6)
.Rows.Add(21.1, 21.1, 21.1, 21.1, 21.1, 21.1, 21.1)
.Rows.Add(14.31, 14.31, 14.31, 14.31, 14.31, 14.31, 14.31)
.Rows.Add(7.35, 7.35, 7.35, 7.35, 7.35, 7.35, 7.35)
.Rows.Add(3.84, 3.84, 3.84, 3.84, 3.84, 3.84, 3.84)
.Rows.Add(3.51, 3.51, 3.51, 3.51, 3.51, 3.51, 3.51)
.Rows.Add(4.23, 4.23, 4.23, 4.23, 4.23, 4.23, 4.23)
.Rows.Add(4.23, 4.23, 4.23, 4.23, 4.23, 4.23, 4.23)
.Rows.Add(5.85, 5.85, 5.85, 5.85, 5.85, 5.85, 5.85)
.Rows.Add(6.12, 6.12, 6.12, 6.12, 6.12, 6.12, 6.12)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(5.18, 5.18, 5.18, 5.18, 5.18, 5.18, 5.18)
.Rows.Add(2.56, 2.56, 2.56, 2.56, 2.56, 2.56, 2.56)
.Rows.Add(0.95, 0.95, 0.95, 0.95, 0.95, 0.95, 0.95)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
With EspectroSimplePorClaseT4
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(2.1, 2.1, 2.1, 2.1, 2.1, 2.1, 2.1)
.Rows.Add(18.18, 18.18, 18.18, 18.18, 18.18, 18.18, 18.18)

```

```

.Rows.Add(20.91, 20.91, 20.91, 20.91, 20.91, 20.91, 20.91)
.Rows.Add(15.23, 15.23, 15.23, 15.23, 15.23, 15.23, 15.23)
.Rows.Add(7.84, 7.84, 7.84, 7.84, 7.84, 7.84, 7.84)
.Rows.Add(3.81, 3.81, 3.81, 3.81, 3.81, 3.81, 3.81)
.Rows.Add(3.47, 3.47, 3.47, 3.47, 3.47, 3.47, 3.47)
.Rows.Add(4.15, 4.15, 4.15, 4.15, 4.15, 4.15, 4.15)
.Rows.Add(4.09, 4.09, 4.09, 4.09, 4.09, 4.09, 4.09)
.Rows.Add(5.74, 5.74, 5.74, 5.74, 5.74, 5.74, 5.74)
.Rows.Add(5.97, 5.97, 5.97, 5.97, 5.97, 5.97, 5.97)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(5.06, 5.06, 5.06, 5.06, 5.06, 5.06, 5.06)
.Rows.Add(2.48, 2.48, 2.48, 2.48, 2.48, 2.48, 2.48)
.Rows.Add(0.97, 0.97, 0.97, 0.97, 0.97, 0.97, 0.97)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
With EspectroSimplePorClaseT5
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(2.13, 2.13, 2.13, 2.13, 2.13, 2.13, 2.13)
.Rows.Add(18.52, 18.52, 18.52, 18.52, 18.52, 18.52, 18.52)
.Rows.Add(21.1, 21.1, 21.1, 21.1, 21.1, 21.1, 21.1)
.Rows.Add(14.59, 14.59, 14.59, 14.59, 14.59, 14.59, 14.59)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(7.46, 7.46, 7.46, 7.46, 7.46, 7.46, 7.46)
.Rows.Add(3.82, 3.82, 3.82, 3.82, 3.82, 3.82, 3.82)
.Rows.Add(3.48, 3.48, 3.48, 3.48, 3.48, 3.48, 3.48)
.Rows.Add(4.21, 4.21, 4.21, 4.21, 4.21, 4.21, 4.21)
.Rows.Add(4.15, 4.15, 4.15, 4.15, 4.15, 4.15, 4.15)
.Rows.Add(5.84, 5.84, 5.84, 5.84, 5.84, 5.84, 5.84)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(6.06, 6.06, 6.06, 6.06, 6.06, 6.06, 6.06)
.Rows.Add(5.16, 5.16, 5.16, 5.16, 5.16, 5.16, 5.16)
.Rows.Add(2.53, 2.53, 2.53, 2.53, 2.53, 2.53, 2.53)
.Rows.Add(0.95, 0.95, 0.95, 0.95, 0.95, 0.95, 0.95)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
With EspectroSimplePorClaseT6
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(2.23, 2.23, 2.23, 2.23, 2.23, 2.23, 2.23)
.Rows.Add(19.14, 19.14, 19.14, 19.14, 19.14, 19.14, 19.14)
.Rows.Add(21.37, 21.37, 21.37, 21.37, 21.37, 21.37, 21.37)

```

```

.Rows.Add(13.1, 13.1, 13.1, 13.1, 13.1, 13.1, 13.1)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(6.74, 6.74, 6.74, 6.74, 6.74, 6.74, 6.74)
.Rows.Add(3.86, 3.86, 3.86, 3.86, 3.86, 3.86, 3.86)
.Rows.Add(3.59, 3.59, 3.59, 3.59, 3.59, 3.59, 3.59)
.Rows.Add(4.35, 4.35, 4.35, 4.35, 4.35, 4.35, 4.35)
.Rows.Add(4.35, 4.35, 4.35, 4.35, 4.35, 4.35, 4.35)
.Rows.Add(6.04, 6.04, 6.04, 6.04, 6.04, 6.04, 6.04)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(6.31, 6.31, 6.31, 6.31, 6.31, 6.31, 6.31)
.Rows.Add(5.33, 5.33, 5.33, 5.33, 5.33, 5.33, 5.33)
.Rows.Add(2.61, 2.61, 2.61, 2.61, 2.61, 2.61, 2.61)
.Rows.Add(0.98, 0.98, 0.98, 0.98, 0.98, 0.98, 0.98)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
'Tandem/general
With EspectroTandemGeneralT1
.Columns.Add("General")
.Rows.Add(0)
.Rows.Add(1.75)
.Rows.Add(7.02)
.Rows.Add(5.26)
.Rows.Add(5.26)
.Rows.Add(6.14)
.Rows.Add(6.14)
.Rows.Add(7.89)
.Rows.Add(12.28)
.Rows.Add(23.68)
.Rows.Add(17.54)
.Rows.Add(5.26)
.Rows.Add(0.89)
.Rows.Add(0.89)
.Rows.Add(0)
.Rows.Add(0)
.Rows.Add(0)
.Rows.Add(0)
End With
With EspectroTandemGeneralT2
.Columns.Add("General")
.Rows.Add(0)
.Rows.Add(1.67)
.Rows.Add(6.67)
.Rows.Add(5)
.Rows.Add(5)
.Rows.Add(5.83)
.Rows.Add(5.83)
.Rows.Add(7.5)
.Rows.Add(12.5)
.Rows.Add(24.17)
.Rows.Add(17.5)
.Rows.Add(5.83)
.Rows.Add(1.67)
.Rows.Add(0.83)
.Rows.Add(0)
.Rows.Add(0)

```

```

        .Rows.Add(0)
        .Rows.Add(0)
    End With
    With EspectroTandemGeneralT3
        .Columns.Add("General")
        .Rows.Add(0)
        .Rows.Add(1.75)
        .Rows.Add(7.46)
        .Rows.Add(4.82)
        .Rows.Add(5.26)
        .Rows.Add(0)
        .Rows.Add(5.71)
        .Rows.Add(5.71)
        .Rows.Add(7.89)
        .Rows.Add(13.16)
        .Rows.Add(24.12)
        .Rows.Add(16.23)
        .Rows.Add(5.26)
        .Rows.Add(1.75)
        .Rows.Add(0.88)
        .Rows.Add(0)
        .Rows.Add(0)
        .Rows.Add(0)
    End With
    With EspectroTandemGeneralT4
        .Columns.Add("General")
        .Rows.Add(0)
        .Rows.Add(1.77)
        .Rows.Add(7.45)
        .Rows.Add(4.96)
        .Rows.Add(5.67)
        .Rows.Add(0)
        .Rows.Add(5.67)
        .Rows.Add(6.03)
        .Rows.Add(8.16)
        .Rows.Add(13.48)
        .Rows.Add(23.76)
        .Rows.Add(15.6)
        .Rows.Add(4.96)
        .Rows.Add(1.77)
        .Rows.Add(0.72)
        .Rows.Add(0)
        .Rows.Add(0)
        .Rows.Add(0)
    End With
    With EspectroTandemGeneralT5
        .Columns.Add("General")
        .Rows.Add(0)
        .Rows.Add(0)
        .Rows.Add(1.98)
        .Rows.Add(7.51)
        .Rows.Add(5.14)
        .Rows.Add(5.53)
        .Rows.Add(5.93)
        .Rows.Add(5.93)
    End With

```

```

.Rows.Add(0)
.Rows.Add(8.3)
.Rows.Add(13.44)
.Rows.Add(23.72)
.Rows.Add(15.42)
.Rows.Add(4.74)
.Rows.Add(1.58)
.Rows.Add(0)
.Rows.Add(0.78)
.Rows.Add(0)
End With
With EspectroTandemGeneralT6
.Columns.Add("General")
.Rows.Add(0)
.Rows.Add(0)
.Rows.Add(1.69)
.Rows.Add(7.34)
.Rows.Add(5.08)
.Rows.Add(5.65)
.Rows.Add(6.22)
.Rows.Add(6.22)
.Rows.Add(0)
.Rows.Add(8.47)
.Rows.Add(13.56)
.Rows.Add(23.73)
.Rows.Add(15.82)
.Rows.Add(4.52)
.Rows.Add(1.13)
.Rows.Add(0)
.Rows.Add(0.57)
.Rows.Add(0)
End With
'Tandem/class
With EspectroTandemPorClaseT1
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(1.75, 1.75, 1.75, 1.75, 1.75, 1.75, 1.75)
.Rows.Add(7.02, 7.02, 7.02, 7.02, 7.02, 7.02, 7.02)
.Rows.Add(5.26, 5.26, 5.26, 5.26, 5.26, 5.26, 5.26)
.Rows.Add(5.26, 5.26, 5.26, 5.26, 5.26, 5.26, 5.26)
.Rows.Add(6.14, 6.14, 6.14, 6.14, 6.14, 6.14, 6.14)
.Rows.Add(6.14, 6.14, 6.14, 6.14, 6.14, 6.14, 6.14)
.Rows.Add(7.89, 7.89, 7.89, 7.89, 7.89, 7.89, 7.89)
.Rows.Add(12.28, 12.28, 12.28, 12.28, 12.28, 12.28, 12.28)
.Rows.Add(23.68, 23.68, 23.68, 23.68, 23.68, 23.68, 23.68)
.Rows.Add(17.54, 17.54, 17.54, 17.54, 17.54, 17.54, 17.54)
.Rows.Add(5.26, 5.26, 5.26, 5.26, 5.26, 5.26, 5.26)
.Rows.Add(0.89, 0.89, 0.89, 0.89, 0.89, 0.89, 0.89)
.Rows.Add(0.89, 0.89, 0.89, 0.89, 0.89, 0.89, 0.89)

```

```

.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
With EspectroTandemPorClaseT2
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(1.67, 1.67, 1.67, 1.67, 1.67, 1.67, 1.67)
.Rows.Add(6.67, 6.67, 6.67, 6.67, 6.67, 6.67, 6.67)
.Rows.Add(5, 5, 5, 5, 5, 5, 5)
.Rows.Add(5, 5, 5, 5, 5, 5, 5)
.Rows.Add(5.83, 5.83, 5.83, 5.83, 5.83, 5.83, 5.83)
.Rows.Add(5.83, 5.83, 5.83, 5.83, 5.83, 5.83, 5.83)
.Rows.Add(7.5, 7.5, 7.5, 7.5, 7.5, 7.5, 7.5)
.Rows.Add(12.5, 12.5, 12.5, 12.5, 12.5, 12.5, 12.5)
.Rows.Add(24.17, 24.17, 24.17, 24.17, 24.17, 24.17, 24.17)
.Rows.Add(17.5, 17.5, 17.5, 17.5, 17.5, 17.5, 17.5)
.Rows.Add(5.83, 5.83, 5.83, 5.83, 5.83, 5.83, 5.83)
.Rows.Add(1.67, 1.67, 1.67, 1.67, 1.67, 1.67, 1.67)
.Rows.Add(0.83, 0.83, 0.83, 0.83, 0.83, 0.83, 0.83)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
With EspectroTandemPorClaseT3
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(1.75, 1.75, 1.75, 1.75, 1.75, 1.75, 1.75)
.Rows.Add(7.46, 7.46, 7.46, 7.46, 7.46, 7.46, 7.46)
.Rows.Add(4.82, 4.82, 4.82, 4.82, 4.82, 4.82, 4.82)
.Rows.Add(5.26, 5.26, 5.26, 5.26, 5.26, 5.26, 5.26)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(5.71, 5.71, 5.71, 5.71, 5.71, 5.71, 5.71)
.Rows.Add(5.71, 5.71, 5.71, 5.71, 5.71, 5.71, 5.71)
.Rows.Add(7.89, 7.89, 7.89, 7.89, 7.89, 7.89, 7.89)
.Rows.Add(13.16, 13.16, 13.16, 13.16, 13.16, 13.16, 13.16)
.Rows.Add(24.12, 24.12, 24.12, 24.12, 24.12, 24.12, 24.12)
.Rows.Add(16.23, 16.23, 16.23, 16.23, 16.23, 16.23, 16.23)
.Rows.Add(5.26, 5.26, 5.26, 5.26, 5.26, 5.26, 5.26)
.Rows.Add(1.75, 1.75, 1.75, 1.75, 1.75, 1.75, 1.75)
.Rows.Add(0.88, 0.88, 0.88, 0.88, 0.88, 0.88, 0.88)

```

```

.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
With EspectroTandemPorClaseT4
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(1.77, 1.77, 1.77, 1.77, 1.77, 1.77, 1.77)
.Rows.Add(7.45, 7.45, 7.45, 7.45, 7.45, 7.45, 7.45)
.Rows.Add(4.96, 4.96, 4.96, 4.96, 4.96, 4.96, 4.96)
.Rows.Add(5.67, 5.67, 5.67, 5.67, 5.67, 5.67, 5.67)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(5.67, 5.67, 5.67, 5.67, 5.67, 5.67, 5.67)
.Rows.Add(6.03, 6.03, 6.03, 6.03, 6.03, 6.03, 6.03)
.Rows.Add(8.16, 8.16, 8.16, 8.16, 8.16, 8.16, 8.16)
.Rows.Add(13.48, 13.48, 13.48, 13.48, 13.48, 13.48, 13.48)
.Rows.Add(23.76, 23.76, 23.76, 23.76, 23.76, 23.76, 23.76)
.Rows.Add(15.6, 15.6, 15.6, 15.6, 15.6, 15.6, 15.6)
.Rows.Add(4.96, 4.96, 4.96, 4.96, 4.96, 4.96, 4.96)
.Rows.Add(1.77, 1.77, 1.77, 1.77, 1.77, 1.77, 1.77)
.Rows.Add(0.71, 0.71, 0.71, 0.71, 0.71, 0.71, 0.71)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
With EspectroTandemPorClaseT5
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(1.98, 1.98, 1.98, 1.98, 1.98, 1.98, 1.98)
.Rows.Add(7.51, 7.51, 7.51, 7.51, 7.51, 7.51, 7.51)
.Rows.Add(5.14, 5.14, 5.14, 5.14, 5.14, 5.14, 5.14)
.Rows.Add(5.53, 5.53, 5.53, 5.53, 5.53, 5.53, 5.53)
.Rows.Add(5.93, 5.93, 5.93, 5.93, 5.93, 5.93, 5.93)
.Rows.Add(5.93, 5.93, 5.93, 5.93, 5.93, 5.93, 5.93)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(8.3, 8.3, 8.3, 8.3, 8.3, 8.3, 8.3)
.Rows.Add(13.44, 13.44, 13.44, 13.44, 13.44, 13.44, 13.44)
.Rows.Add(23.72, 23.72, 23.72, 23.72, 23.72, 23.72, 23.72)
.Rows.Add(15.42, 15.42, 15.42, 15.42, 15.42, 15.42, 15.42)
.Rows.Add(4.74, 4.74, 4.74, 4.74, 4.74, 4.74, 4.74)
.Rows.Add(1.58, 1.58, 1.58, 1.58, 1.58, 1.58, 1.58)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)

```

```

        .Rows.Add(0.78, 0.78, 0.78, 0.78, 0.78, 0.78, 0.78)
        .Rows.Add(0, 0, 0, 0, 0, 0, 0)
    End With
    With EspectroTandemPorClaseT6
        .Columns.Add("BUS")
        .Columns.Add("C2P")
        .Columns.Add("C2G")
        .Columns.Add("C3")
        .Columns.Add("C4")
        .Columns.Add("C5")
        .Columns.Add("> C5")
        .Rows.Add(0, 0, 0, 0, 0, 0, 0)
        .Rows.Add(0, 0, 0, 0, 0, 0, 0)
        .Rows.Add(1.69, 1.69, 1.69, 1.69, 1.69, 1.69, 1.69)
        .Rows.Add(7.34, 7.34, 7.34, 7.34, 7.34, 7.34, 7.34)
        .Rows.Add(5.08, 5.08, 5.08, 5.08, 5.08, 5.08, 5.08)
        .Rows.Add(5.65, 5.65, 5.65, 5.65, 5.65, 5.65, 5.65)
        .Rows.Add(6.22, 6.22, 6.22, 6.22, 6.22, 6.22, 6.22)
        .Rows.Add(6.22, 6.22, 6.22, 6.22, 6.22, 6.22, 6.22)
        .Rows.Add(0, 0, 0, 0, 0, 0, 0)
        .Rows.Add(8.47, 8.47, 8.47, 8.47, 8.47, 8.47, 8.47)
        .Rows.Add(13.56, 13.56, 13.56, 13.56, 13.56, 13.56, 13.56)
        .Rows.Add(23.73, 23.73, 23.73, 23.73, 23.73, 23.73, 23.73)
        .Rows.Add(15.82, 15.82, 15.82, 15.82, 15.82, 15.82, 15.82)
        .Rows.Add(4.52, 4.52, 4.52, 4.52, 4.52, 4.52, 4.52)
        .Rows.Add(1.13, 1.13, 1.13, 1.13, 1.13, 1.13, 1.13)
        .Rows.Add(0, 0, 0, 0, 0, 0, 0)
        .Rows.Add(0.57, 0.57, 0.57, 0.57, 0.57, 0.57, 0.57)
        .Rows.Add(0, 0, 0, 0, 0, 0, 0)
    End With

    'Tridem/general
    With EspectroTridemGeneralT1
        .Columns.Add("General")
        .Rows.Add(0)
        .Rows.Add(0)
        .Rows.Add(4.35)
        .Rows.Add(0)
        .Rows.Add(0)
        .Rows.Add(0)
        .Rows.Add(4.35)
        .Rows.Add(4.35)
        .Rows.Add(4.35)
        .Rows.Add(4.35)
        .Rows.Add(4.35)
        .Rows.Add(17.39)
        .Rows.Add(21.73)
        .Rows.Add(13.03)
        .Rows.Add(8.7)
        .Rows.Add(4.35)
        .Rows.Add(4.35)
        .Rows.Add(4.35)
        .Rows.Add(4.35)
        .Rows.Add(0)
        .Rows.Add(0)
        .Rows.Add(0)
    End With

```

```

        .Rows.Add(0)
End With
With SpectroTridemGeneralT2
    .Columns.Add("General")
    .Rows.Add(0)
    .Rows.Add(3.33)
    .Rows.Add(3.33)
    .Rows.Add(3.33)
    .Rows.Add(0)
    .Rows.Add(0)
    .Rows.Add(3.33)
    .Rows.Add(3.33)
    .Rows.Add(3.33)
    .Rows.Add(6.68)
    .Rows.Add(16.68)
    .Rows.Add(23.33)
    .Rows.Add(13.33)
    .Rows.Add(6.68)
    .Rows.Add(3.33)
    .Rows.Add(3.33)
    .Rows.Add(3.33)
    .Rows.Add(3.33)
    .Rows.Add(0)
    .Rows.Add(0)
    .Rows.Add(0)
    .Rows.Add(0)
End With
With SpectroTridemGeneralT3
    .Columns.Add("General")
    .Rows.Add(0)
    .Rows.Add(1.49)
    .Rows.Add(2.99)
    .Rows.Add(1.49)
    .Rows.Add(1.49)
    .Rows.Add(0)
    .Rows.Add(1.49)
    .Rows.Add(2.99)
    .Rows.Add(2.99)
    .Rows.Add(4.47)
    .Rows.Add(7.46)
    .Rows.Add(16.42)
    .Rows.Add(22.39)
    .Rows.Add(13.43)
    .Rows.Add(7.46)
    .Rows.Add(4.47)
    .Rows.Add(2.99)
    .Rows.Add(2.99)
    .Rows.Add(2.99)
    .Rows.Add(0)
    .Rows.Add(0)
    .Rows.Add(0)
End With
With SpectroTridemGeneralT4
    .Columns.Add("General")
    .Rows.Add(0)

```

```

        .Rows.Add(2.63)
        .Rows.Add(3.95)
        .Rows.Add(1.32)
        .Rows.Add(1.32)
        .Rows.Add(0)
        .Rows.Add(1.32)
        .Rows.Add(2.63)
        .Rows.Add(2.63)
        .Rows.Add(3.95)
        .Rows.Add(6.57)
        .Rows.Add(17.11)
        .Rows.Add(22.37)
        .Rows.Add(13.15)
        .Rows.Add(7.89)
        .Rows.Add(3.95)
        .Rows.Add(3.95)
        .Rows.Add(2.63)
        .Rows.Add(2.63)
        .Rows.Add(0)
        .Rows.Add(0)
        .Rows.Add(0)
End With
With EspectroTridemGeneralT5
    .Columns.Add("General")
    .Rows.Add(0)
    .Rows.Add(0)
    .Rows.Add(1.56)
    .Rows.Add(3.13)
    .Rows.Add(1.56)
    .Rows.Add(1.56)
    .Rows.Add(1.56)
    .Rows.Add(3.13)
    .Rows.Add(0)
    .Rows.Add(3.13)
    .Rows.Add(4.69)
    .Rows.Add(6.24)
    .Rows.Add(17.18)
    .Rows.Add(21.88)
    .Rows.Add(12.49)
    .Rows.Add(0)
    .Rows.Add(7.81)
    .Rows.Add(4.69)
    .Rows.Add(3.13)
    .Rows.Add(3.13)
    .Rows.Add(3.13)
    .Rows.Add(0)
End With
With EspectroTridemGeneralT6
    .Columns.Add("General")
    .Rows.Add(0)
    .Rows.Add(0)
    .Rows.Add(2.63)
    .Rows.Add(2.63)
    .Rows.Add(2.63)
    .Rows.Add(2.63)

```

```

.Rows.Add(2.63)
.Rows.Add(2.63)
.Rows.Add(0)
.Rows.Add(2.63)
.Rows.Add(5.26)
.Rows.Add(7.9)
.Rows.Add(15.79)
.Rows.Add(21.05)
.Rows.Add(13.17)
.Rows.Add(0)
.Rows.Add(7.9)
.Rows.Add(2.63)
.Rows.Add(2.63)
.Rows.Add(2.63)
.Rows.Add(2.63)
.Rows.Add(0)
End With
'Tridem/class
With EspectroTridemPorClaseT1
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(4.35, 4.35, 4.35, 4.35, 4.35, 4.35, 4.35)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(4.35, 4.35, 4.35, 4.35, 4.35, 4.35, 4.35)
.Rows.Add(4.35, 4.35, 4.35, 4.35, 4.35, 4.35, 4.35)
.Rows.Add(4.35, 4.35, 4.35, 4.35, 4.35, 4.35, 4.35)
.Rows.Add(4.35, 4.35, 4.35, 4.35, 4.35, 4.35, 4.35)
.Rows.Add(17.39, 17.39, 17.39, 17.39, 17.39, 17.39, 17.39)
.Rows.Add(21.73, 21.73, 21.73, 21.73, 21.73, 21.73, 21.73)
.Rows.Add(13.03, 13.03, 13.03, 13.03, 13.03, 13.03, 13.03)
.Rows.Add(8.7, 8.7, 8.7, 8.7, 8.7, 8.7, 8.7)
.Rows.Add(4.35, 4.35, 4.35, 4.35, 4.35, 4.35, 4.35)
.Rows.Add(4.35, 4.35, 4.35, 4.35, 4.35, 4.35, 4.35)
.Rows.Add(4.35, 4.35, 4.35, 4.35, 4.35, 4.35, 4.35)
.Rows.Add(4.35, 4.35, 4.35, 4.35, 4.35, 4.35, 4.35)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
With EspectroTridemPorClaseT2
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")

```

```

.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(3.33, 3.33, 3.33, 3.33, 3.33, 3.33, 3.33)
.Rows.Add(3.33, 3.33, 3.33, 3.33, 3.33, 3.33, 3.33)
.Rows.Add(3.33, 3.33, 3.33, 3.33, 3.33, 3.33, 3.33)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(3.33, 3.33, 3.33, 3.33, 3.33, 3.33, 3.33)
.Rows.Add(3.33, 3.33, 3.33, 3.33, 3.33, 3.33, 3.33)
.Rows.Add(3.33, 3.33, 3.33, 3.33, 3.33, 3.33, 3.33)
.Rows.Add(6.66, 6.66, 6.66, 6.66, 6.66, 6.66, 6.66)
.Rows.Add(16.66, 16.66, 16.66, 16.66, 16.66, 16.66, 16.66)
.Rows.Add(23.33, 23.33, 23.33, 23.33, 23.33, 23.33, 23.33)
.Rows.Add(13.33, 13.33, 13.33, 13.33, 13.33, 13.33, 13.33)
.Rows.Add(6.66, 6.66, 6.66, 6.66, 6.66, 6.66, 6.66)
.Rows.Add(3.33, 3.33, 3.33, 3.33, 3.33, 3.33, 3.33)
.Rows.Add(3.33, 3.33, 3.33, 3.33, 3.33, 3.33, 3.33)
.Rows.Add(3.33, 3.33, 3.33, 3.33, 3.33, 3.33, 3.33)
.Rows.Add(3.33, 3.33, 3.33, 3.33, 3.33, 3.33, 3.33)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)

```

End With

With EspectroTridemPorClaseT3

```

.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(1.49, 1.49, 1.49, 1.49, 1.49, 1.49, 1.49)
.Rows.Add(2.99, 2.99, 2.99, 2.99, 2.99, 2.99, 2.99)
.Rows.Add(1.49, 1.49, 1.49, 1.49, 1.49, 1.49, 1.49)
.Rows.Add(1.49, 1.49, 1.49, 1.49, 1.49, 1.49, 1.49)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(1.49, 1.49, 1.49, 1.49, 1.49, 1.49, 1.49)
.Rows.Add(2.99, 2.99, 2.99, 2.99, 2.99, 2.99, 2.99)
.Rows.Add(2.99, 2.99, 2.99, 2.99, 2.99, 2.99, 2.99)
.Rows.Add(4.47, 4.47, 4.47, 4.47, 4.47, 4.47, 4.47)
.Rows.Add(7.46, 7.46, 7.46, 7.46, 7.46, 7.46, 7.46)
.Rows.Add(16.42, 16.42, 16.42, 16.42, 16.42, 16.42, 16.42)
.Rows.Add(22.39, 22.39, 22.39, 22.39, 22.39, 22.39, 22.39)
.Rows.Add(13.43, 13.43, 13.43, 13.43, 13.43, 13.43, 13.43)
.Rows.Add(7.46, 7.46, 7.46, 7.46, 7.46, 7.46, 7.46)
.Rows.Add(4.47, 4.47, 4.47, 4.47, 4.47, 4.47, 4.47)
.Rows.Add(2.99, 2.99, 2.99, 2.99, 2.99, 2.99, 2.99)
.Rows.Add(2.99, 2.99, 2.99, 2.99, 2.99, 2.99, 2.99)
.Rows.Add(2.99, 2.99, 2.99, 2.99, 2.99, 2.99, 2.99)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)

```

```

        .Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
With EspectroTridemPorClaseT4
    .Columns.Add("BUS")
    .Columns.Add("C2P")
    .Columns.Add("C2G")
    .Columns.Add("C3")
    .Columns.Add("C4")
    .Columns.Add("C5")
    .Columns.Add("> C5")
    .Rows.Add(0, 0, 0, 0, 0, 0, 0)
    .Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
    .Rows.Add(3.95, 3.95, 3.95, 3.95, 3.95, 3.95, 3.95)
    .Rows.Add(1.32, 1.32, 1.32, 1.32, 1.32, 1.32, 1.32)
    .Rows.Add(1.32, 1.32, 1.32, 1.32, 1.32, 1.32, 1.32)
    .Rows.Add(0, 0, 0, 0, 0, 0, 0)
    .Rows.Add(1.32, 1.32, 1.32, 1.32, 1.32, 1.32, 1.32)
    .Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
    .Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
    .Rows.Add(3.95, 3.95, 3.95, 3.95, 3.95, 3.95, 3.95)
    .Rows.Add(6.57, 6.57, 6.57, 6.57, 6.57, 6.57, 6.57)
    .Rows.Add(17.11, 17.11, 17.11, 17.11, 17.11, 17.11, 17.11)
    .Rows.Add(22.37, 22.37, 22.37, 22.37, 22.37, 22.37, 22.37)
    .Rows.Add(13.15, 13.15, 13.15, 13.15, 13.15, 13.15, 13.15)
    .Rows.Add(7.89, 7.89, 7.89, 7.89, 7.89, 7.89, 7.89)
    .Rows.Add(3.95, 3.95, 3.95, 3.95, 3.95, 3.95, 3.95)
    .Rows.Add(3.95, 3.95, 3.95, 3.95, 3.95, 3.95, 3.95)
    .Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
    .Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
    .Rows.Add(0, 0, 0, 0, 0, 0, 0)
    .Rows.Add(0, 0, 0, 0, 0, 0, 0)
    .Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
With EspectroTridemPorClaseT5
    .Columns.Add("BUS")
    .Columns.Add("C2P")
    .Columns.Add("C2G")
    .Columns.Add("C3")
    .Columns.Add("C4")
    .Columns.Add("C5")
    .Columns.Add("> C5")
    .Rows.Add(0, 0, 0, 0, 0, 0, 0)
    .Rows.Add(0, 0, 0, 0, 0, 0, 0)
    .Rows.Add(1.56, 1.56, 1.56, 1.56, 1.56, 1.56, 1.56)
    .Rows.Add(3.13, 3.13, 3.13, 3.13, 3.13, 3.13, 3.13)
    .Rows.Add(1.56, 1.56, 1.56, 1.56, 1.56, 1.56, 1.56)
    .Rows.Add(1.56, 1.56, 1.56, 1.56, 1.56, 1.56, 1.56)
    .Rows.Add(1.56, 1.56, 1.56, 1.56, 1.56, 1.56, 1.56)
    .Rows.Add(3.13, 3.13, 3.13, 3.13, 3.13, 3.13, 3.13)
    .Rows.Add(0, 0, 0, 0, 0, 0, 0)
    .Rows.Add(3.13, 3.13, 3.13, 3.13, 3.13, 3.13, 3.13)
    .Rows.Add(4.69, 4.69, 4.69, 4.69, 4.69, 4.69, 4.69)
    .Rows.Add(6.24, 6.24, 6.24, 6.24, 6.24, 6.24, 6.24)
    .Rows.Add(17.18, 17.18, 17.18, 17.18, 17.18, 17.18, 17.18)
    .Rows.Add(21.88, 21.88, 21.88, 21.88, 21.88, 21.88, 21.88)

```

```

.Rows.Add(12.49, 12.49, 12.49, 12.49, 12.49, 12.49, 12.49)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(7.81, 7.81, 7.81, 7.81, 7.81, 7.81, 7.81)
.Rows.Add(4.69, 4.69, 4.69, 4.69, 4.69, 4.69, 4.69)
.Rows.Add(3.13, 3.13, 3.13, 3.13, 3.13, 3.13, 3.13)
.Rows.Add(3.13, 3.13, 3.13, 3.13, 3.13, 3.13, 3.13)
.Rows.Add(3.13, 3.13, 3.13, 3.13, 3.13, 3.13, 3.13)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
With EspectroTridemPorClaseT6
.Columns.Add("BUS")
.Columns.Add("C2P")
.Columns.Add("C2G")
.Columns.Add("C3")
.Columns.Add("C4")
.Columns.Add("C5")
.Columns.Add("> C5")
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
.Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
.Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
.Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
.Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
.Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
.Rows.Add(5.26, 5.26, 5.26, 5.26, 5.26, 5.26, 5.26)
.Rows.Add(7.9, 7.9, 7.9, 7.9, 7.9, 7.9, 7.9)
.Rows.Add(15.79, 15.79, 15.79, 15.79, 15.79, 15.79, 15.79)
.Rows.Add(21.05, 21.05, 21.05, 21.05, 21.05, 21.05, 21.05)
.Rows.Add(13.17, 13.17, 13.17, 13.17, 13.17, 13.17, 13.17)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
.Rows.Add(7.9, 7.9, 7.9, 7.9, 7.9, 7.9, 7.9)
.Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
.Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
.Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
.Rows.Add(2.63, 2.63, 2.63, 2.63, 2.63, 2.63, 2.63)
.Rows.Add(0, 0, 0, 0, 0, 0, 0)
End With
'DataSets
dsSimpleGeneral.Tables.Add(EspectroSimpleGeneralT1)
dsSimpleGeneral.Tables.Add(EspectroSimpleGeneralT2)
dsSimpleGeneral.Tables.Add(EspectroSimpleGeneralT3)
dsSimpleGeneral.Tables.Add(EspectroSimpleGeneralT4)
dsSimpleGeneral.Tables.Add(EspectroSimpleGeneralT5)
dsSimpleGeneral.Tables.Add(EspectroSimpleGeneralT6)
dsSimplePorClase.Tables.Add(EspectroSimplePorClaseT1)
dsSimplePorClase.Tables.Add(EspectroSimplePorClaseT2)
dsSimplePorClase.Tables.Add(EspectroSimplePorClaseT3)
dsSimplePorClase.Tables.Add(EspectroSimplePorClaseT4)
dsSimplePorClase.Tables.Add(EspectroSimplePorClaseT5)
dsSimplePorClase.Tables.Add(EspectroSimplePorClaseT6)
dsTandemGeneral.Tables.Add(EspectroTandemGeneralT1)
dsTandemGeneral.Tables.Add(EspectroTandemGeneralT2)

```

```

dsTandemGeneral.Tables.Add(EspectroTandemGeneralT3)
dsTandemGeneral.Tables.Add(EspectroTandemGeneralT4)
dsTandemGeneral.Tables.Add(EspectroTandemGeneralT5)
dsTandemGeneral.Tables.Add(EspectroTandemGeneralT6)
dsTandemPorClase.Tables.Add(EspectroTandemPorClaseT1)
dsTandemPorClase.Tables.Add(EspectroTandemPorClaseT2)
dsTandemPorClase.Tables.Add(EspectroTandemPorClaseT3)
dsTandemPorClase.Tables.Add(EspectroTandemPorClaseT4)
dsTandemPorClase.Tables.Add(EspectroTandemPorClaseT5)
dsTandemPorClase.Tables.Add(EspectroTandemPorClaseT6)
dsTridemGeneral.Tables.Add(EspectroTridemGeneralT1)
dsTridemGeneral.Tables.Add(EspectroTridemGeneralT2)
dsTridemGeneral.Tables.Add(EspectroTridemGeneralT3)
dsTridemGeneral.Tables.Add(EspectroTridemGeneralT4)
dsTridemGeneral.Tables.Add(EspectroTridemGeneralT5)
dsTridemGeneral.Tables.Add(EspectroTridemGeneralT6)
dsTridemPorClase.Tables.Add(EspectroTridemPorClaseT1)
dsTridemPorClase.Tables.Add(EspectroTridemPorClaseT2)
dsTridemPorClase.Tables.Add(EspectroTridemPorClaseT3)
dsTridemPorClase.Tables.Add(EspectroTridemPorClaseT4)
dsTridemPorClase.Tables.Add(EspectroTridemPorClaseT5)
dsTridemPorClase.Tables.Add(EspectroTridemPorClaseT6)
With glEspectrosDeCarga
    .TipoEspectroSimple = "General"
    .TipoEspectroTandem = "General"
    .TipoEspectroTridem = "General"
    .EspectroSimpleGeneral = dsSimpleGeneral.Tables(0)
    .EspectroTandemGeneral = dsTandemGeneral.Tables(0)
    .EspectroTridemGeneral = dsTridemGeneral.Tables(0)
    .EspectroSimplePorClase = dsSimplePorClase.Tables(0)
    .EspectroTandemPorClase = dsTandemPorClase.Tables(0)
    .EspectroTridemPorClase = dsTridemPorClase.Tables(0)
End With
.EspectroCarga = glEspectrosDeCarga
'ENTRADAS GENERALES
.MediaHuella = 0.375
.DesvHuella = 0.2
.AnchoCarril = 3.65
glBatalla.MediamEspaciamiento = {3.75, 4.65, 5.75}
glBatalla.PorcentajeCamiones = {33, 33, 34}
.Batalla = glBatalla
With glConfigEje
    .AnchoEje = 2.6
    .DistGemelado = 0.303
    .PresionInflado = 820.8
    .EspaciamientoTandem = 1.3
    .EspaciamientoTridem = 1.3
End With
.ConfiguracionEje = glConfigEje
Dim tablaEjesClase As New DataTable
With tablaEjesClase
    .Columns.Add("CLASE", GetType(String))
    .Columns.Add("SIMPLE", GetType(String))
    .Columns.Add("TÁNDEM", GetType(String))
    .Columns.Add("TRÍDEM", GetType(String))

```

```

        .Rows.Add("BUSES", 2.0, 0.0, 0.0)
        .Rows.Add("C2P", 2.0, 0.0, 0.0)
        .Rows.Add("C2G", 2.0, 0.0, 0.0)
        .Rows.Add("C3", 1.0, 1.0, 0.0)
        .Rows.Add("C4", 2.0, 1.0, 0.0)
        .Rows.Add("C5", 1.0, 2.0, 0.0)
        .Rows.Add("> C5", 1.0, 1.0, 1.0)
    End With
    .NumEjesClase = tablaEjesClase
    'FACTORES DE CRECIMIENTO
    With glFacCrecimiento
        .TipoCrecimiento = "Compuesto"
        .UnicoValor = True
        .FactorGeneral = 3
        .FactorPorClase = {3, 3, 3, 3, 3, 3, 3}
    End With
    .FactCrecimiento = glFacCrecimiento
End With
End Sub
Public Sub ActualizarInfoProyecto()
    glInformacionProyecto.General = glInformacionGeneral
    glInformacionProyecto.Trafico = glInformacionTrafico
    glInformacionProyecto.Estructura = glInformacionEstructura
End Sub
End Module

```

Anexo C.2 Información de tráfico

Anexo C.2.1 Formularios

Tránsito

```

Imports System.IO
Imports System.Xml.Serialization
Imports Microsoft.Office.Interop.Excel
Public Class Transito
    Public srlzTrafico As New XmlSerializer(glInformacionTrafico.GetType)
    Public strmTrafico As Stream
    'EXTRAER INFORMACIÓN A CONTROLES
    Private Sub RecopilarInformacionTransito()
        With glInformacionTrafico
            .TPDAT = CDb1(txtTPDinicial.Text)
            .NumCarril = CDb1(txtCarriles.Text)
            .FactorDir = CDb1(txtFactorDireccional.Text)
            .FactorCarril = CDb1(txtPorcentajeCarril.Text)
            .VelOper = CDb1(txtVelocidad.Text)
        End With
    End Sub
    'REPORTE EN EXCEL
    Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
        RecopilarInformacionTransito()
        Dim excel As New Microsoft.Office.Interop.Excel.Application
        Dim libro As Microsoft.Office.Interop.Excel.Workbook = excel.Workbooks.Add()
        Dim hoja As New Microsoft.Office.Interop.Excel.Worksheet
        excel.Visible = False
    End Sub
End Class

```

```

excel.Worksheets.Add()
excel.Worksheets.Add()
excel.Worksheets.Add()
excel.Worksheets.Add()
'HOJA 1: ENTRADAS GENERALES
hoja = excel.Sheets(1)
With hoja
    'CAMBIAR NOMBRE
    .Name = "ENTRADAS GENERALES"
    'ANCHOS DE COLUMNA
    .Columns("A").ColumnWidth = 4
    .Columns("B").ColumnWidth = 4
    .Columns("C").ColumnWidth = 18
    .Columns("E").ColumnWidth = 4
    .Columns("F").ColumnWidth = 35
    .Columns("H").ColumnWidth = 4
    .Columns("I").ColumnWidth = 15
    .Columns("M").ColumnWidth = 4
    'COLOR DE FONDO TITULOS
    .Range("B2", "M21").Interior.Color = Color.White
    .Range("C3", "D3").Interior.Color = RGB(0, 20, 50)
    .Range("F3", "G3").Interior.Color = RGB(0, 20, 50)
    .Range("F10", "G10").Interior.Color = RGB(0, 20, 50)
    .Range("F15", "G15").Interior.Color = RGB(0, 20, 50)
    .Range("I3", "L3").Interior.Color = RGB(0, 20, 50)
    .Range("I14", "L14").Interior.Color = RGB(0, 20, 50)
    'COLOR DE FUENTE TITULOS
    .Range("C3", "D3").Font.Color = RGB(255, 255, 255)
    .Range("F3", "G3").Font.Color = RGB(255, 255, 255)
    .Range("F10", "G10").Font.Color = RGB(255, 255, 255)
    .Range("F15", "G15").Font.Color = RGB(255, 255, 255)
    .Range("I3", "L3").Font.Color = RGB(255, 255, 255)
    .Range("I14", "L14").Font.Color = RGB(255, 255, 255)
    'COMBINAR CELDAS
    .Range("C3", "D3").Merge()
    .Range("F3", "G3").Merge()
    .Range("F10", "G10").Merge()
    .Range("F15", "G15").Merge()
    .Range("I3", "L3").Merge()
    .Range("I4", "I5").Merge()
    .Range("J4", "L4").Merge()
    .Range("I14", "L14").Merge()
    'FUENTE EN NEGRITA
    .Range("C3", "D3").Font.Bold = True
    .Range("F3", "G3").Font.Bold = True
    .Range("F10", "G10").Font.Bold = True
    .Range("F15", "G15").Font.Bold = True
    .Range("I3", "L5").Font.Bold = True
    .Range("I14", "L15").Font.Bold = True
    'TEXTO CENTRADO
    .Range("C3", "D3").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
    .Range("F3", "G3").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter

```

```

        .Range("F4", "G8").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
        .Range("F10", "G10").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
        .Range("F11", "G13").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
        .Range("F15", "G15").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
        .Range("F16", "G20").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
        .Range("I3", "L12").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
        .Range("I4", "I5").VerticalAlignment =
Microsoft.Office.Interop.Excel.XlVAlign.xlVAlignCenter
        .Range("I14", "L17").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
        'BORDES
        .Range("B2", "M21").Borders(XlBordersIndex.xlEdgeTop).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Range("B2", "M21").Borders(XlBordersIndex.xlEdgeBottom).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Range("B2", "M21").Borders(XlBordersIndex.xlEdgeLeft).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Range("B2", "M21").Borders(XlBordersIndex.xlEdgeRight).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        For i = 3 To 12
            .Cells(i, 3).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
            .Cells(i, 4).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        Next
        For i = 3 To 8
            .Cells(i, 6).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
            .Cells(i, 7).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        Next
        For i = 10 To 13
            .Cells(i, 6).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
            .Cells(i, 7).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        Next
        For i = 15 To 20
            .Cells(i, 6).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
            .Cells(i, 7).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        Next
        For i = 3 To 12
            .Cells(i, 9).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
            .Cells(i, 10).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous

```

```

        .Cells(i, 11).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 12).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    Next
    For i = 14 To 17
        .Cells(i, 9).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 10).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 11).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 12).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    Next
    'TITULOS Y DATOS
    .Range("C3").Value = "INFORMACIÓN GENERAL"
    .Range("C4").Value = "Tipo de análisis"
    .Range("C5").Value = "Tipo de pavimento"
    .Range("C6").Value = "Descripción"
    .Range("C7").Value = "Departamento"
    .Range("C8").Value = "Municipio"
    .Range("C9").Value = "Estación inicial"
    .Range("C10").Value = "Estación final"
    .Range("C11").Value = "Vida de diseño"
    .Range("C12").Value = "Apertura de tráfico"
    .Range("D4").Value = glInformacionGeneral.TipoAnalisis
    .Range("D5").Value = glInformacionGeneral.TipoPavimento
    .Range("D6").Value = glInformacionGeneral.Descripcion
    .Range("D7").Value = glInformacionGeneral.Departamento
    .Range("D8").Value = glInformacionGeneral.Municipio
    .Range("D9").Value = glInformacionGeneral.EstacionInicial
    .Range("D10").Value = glInformacionGeneral.EstacionFinal
    .Range("D11").Value = glInformacionGeneral.VidaDiseño
    .Range("D12").Value = glInformacionGeneral.FechaApertTrafico.Date
    .Range("F3").Value = "TPDAT Y FACTORES"
    .Range("F4").Value = "TPDAT"
    .Range("F5").Value = "N° carriles"
    .Range("F6").Value = "Factor de carril"
    .Range("F7").Value = "Factor direccional"
    .Range("F8").Value = "Velocidad de operación [km/h]"
    .Range("F10").Value = "HUELLA Y CALZADA"
    .Range("F11").Value = "Ancho de carril [m]"
    .Range("F12").Value = "Distancia promedio a la huella [m]"
    .Range("F13").Value = "Desviación estándar de huella [m]"
    .Range("F15").Value = "CANTIDAD Y CONFIGURACIÓN DE EJES"
    .Range("F16").Value = "Ancho promedio de eje [m]"
    .Range("F17").Value = "Distancia de gemelado [m]"
    .Range("F18").Value = "Presión de inflado [kPa]"
    .Range("F19").Value = "Espaciamiento entre ejes tándem [m]"
    .Range("F20").Value = "Espaciamiento entre ejes trídem [m]"
    .Range("G4").Value = glInformacionTrafico.TPDAT
    .Range("G5").Value = glInformacionTrafico.NumCarril
    .Range("G6").Value = glInformacionTrafico.FactorCarril
    .Range("G7").Value = glInformacionTrafico.FactorDir

```

```

        .Range("G8").Value = glInformacionTrafico.VelOper
        .Range("G11").Value = glInformacionTrafico.AnchoCarril
        .Range("G12").Value = glInformacionTrafico.MedeaHuella
        .Range("G13").Value = glInformacionTrafico.DesvHuella
        .Range("G16").Value = glInformacionTrafico.ConfiguracionEje.AnchoEje
        .Range("G17").Value = glInformacionTrafico.ConfiguracionEje.DistGemelado
        .Range("G18").Value =
glInformacionTrafico.ConfiguracionEje.PresionInflado
        .Range("G19").Value =
glInformacionTrafico.ConfiguracionEje.EspaciamientoTandem
        .Range("G20").Value =
glInformacionTrafico.ConfiguracionEje.EspaciamientoTridem
        .Range("I3").Value = "NÚMERO DE EJES POR CLASE DE VEHÍCULO"
        .Range("I4").Value = "CLASE"
        .Range("J4").Value = "TIPO EJE"
        .Range("J5").Value = "Simple"
        .Range("K5").Value = "Tándem"
        .Range("L5").Value = "Trídem"
        For i = 0 To glInformacionTrafico.NumEjesClase.Rows.Count - 1
            For j = 0 To glInformacionTrafico.NumEjesClase.Columns.Count - 1
                .Cells(6 + i, 9 + j) =
glInformacionTrafico.NumEjesClase.Rows(i).Item(j)
            Next
        Next
        .Range("I14").Value = "DATOS DE BATALLA (O EMPATE)"
        .Range("I15").Value = "Propiedad"
        .Range("J15").Value = "Corto"
        .Range("K15").Value = "Medio"
        .Range("L15").Value = "Largo"
        .Range("I16").Value = "Distancia [m]"
        .Range("I17").Value = "Porcentajes [m]"
        For i = 0 To glInformacionTrafico.Batalla.MedeaEspaciamiento.Length - 1
            .Cells(16, 10 + i) =
glInformacionTrafico.Batalla.MedeaEspaciamiento(i)
        Next
        For i = 0 To glInformacionTrafico.Batalla.PorcentajeCamiones.Length - 1
            .Cells(17, 10 + i) =
glInformacionTrafico.Batalla.PorcentajeCamiones(i)
        Next
    End With
    'HOJA 2: VOLUMENES DE TRAFICO
    hoja = excel.Sheets(2)
    With hoja
        'CAMBIAR NOMBRE
        .Name = "VOLUMEN"
        'ANCHOS DE COLUMNA
        .Columns("A").ColumnWidth = 4
        .Columns("B").ColumnWidth = 4
        .Columns("C").ColumnWidth = 12
        .Columns("E").ColumnWidth = 4
        .Columns("F").ColumnWidth = 12
        .Columns("H").ColumnWidth = 4
        .Columns("I").ColumnWidth = 15
        .Columns("Q").ColumnWidth = 4
    End With

```

```

'COLOR DE FONDO TITULOS
    .Range("B2", "Q29").Interior.Color = Color.White
    .Range("C3", "D3").Interior.Color = RGB(0, 20, 50)
    .Range("F3", "G3").Interior.Color = RGB(0, 20, 50)
    .Range("I3", "P3").Interior.Color = RGB(0, 20, 50)
'COLOR DE FUENTE TITULOS
    .Range("C3", "D3").Font.Color = RGB(255, 255, 255)
    .Range("F3", "G3").Font.Color = RGB(255, 255, 255)
    .Range("I3", "P3").Font.Color = RGB(255, 255, 255)
'COMBINAR CELDAS
    .Range("C3", "D3").Merge()
    .Range("F3", "G3").Merge()
    .Range("I3", "P3").Merge()
    .Range("I4", "I5").Merge()
    .Range("J4", "P4").Merge()
'FUENTE EN NEGRITA
    .Range("C3", "D4").Font.Bold = True
    .Range("F3", "G4").Font.Bold = True
    .Range("I3", "P5").Font.Bold = True
'TEXTO CENTRADO
    .Range("C3", "D11").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
    .Range("F3", "G28").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
    .Range("I3", "P17").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
    .Range("I4", "I5").VerticalAlignment =
Microsoft.Office.Interop.Excel.XlVAlign.xlVAlignCenter
'BORDES
    .Range("B2", "Q29").Borders(XlBordersIndex.xlEdgeTop).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    .Range("B2", "Q29").Borders(XlBordersIndex.xlEdgeBottom).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    .Range("B2", "Q29").Borders(XlBordersIndex.xlEdgeLeft).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    .Range("B2", "Q29").Borders(XlBordersIndex.xlEdgeRight).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    For i = 3 To 11
        .Cells(i, 3).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 4).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    Next
    For i = 3 To 28
        .Cells(i, 6).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 7).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    Next
    For i = 3 To 17
        .Cells(i, 9).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 10).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous

```

```

        .Cells(i, 11).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 12).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 13).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 14).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 15).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 16).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    Next
    'TITULOS Y DATOS
    .Range("C3").Value = "DISTRIBUCIÓN POR CLASE"
    .Range("C4").Value = "CLASE"
    .Range("D4").Value = "%"
    .Range("C5").Value = "BUS"
    .Range("C6").Value = "C2P"
    .Range("C7").Value = "C2G"
    .Range("C8").Value = "C3"
    .Range("C9").Value = "C4"
    .Range("C10").Value = "C5"
    .Range("C11").Value = "> C5"
    For i = 0 To glInformacionTrafico.DistrClase.Porcentajes.Length - 1
        .Cells(5 + i, 4) = glInformacionTrafico.DistrClase.Porcentajes(i)
    Next
    .Range("F3").Value = "DISTRIBUCIÓN HORARIA"
    .Range("F4").Value = "HORA"
    .Range("G4").Value = "%"
    For i = 0 To glInformacionTrafico.DistrHoraria.Porcentajes.Length - 1
        .Cells(5 + i, 6) = CStr(i) + ":00"
        .Cells(5 + i, 7) = glInformacionTrafico.DistrHoraria.Porcentajes(i)
    Next
    .Range("I3").Value = "FACTORES MENSUALES"
    .Range("I4").Value = "MES"
    .Range("J4").Value = "CLASE VEHÍCULO"
    .Range("J5").Value = "BUS"
    .Range("K5").Value = "C2P"
    .Range("L5").Value = "C2G"
    .Range("M5").Value = "C3"
    .Range("N5").Value = "C4"
    .Range("O5").Value = "C5"
    .Range("P5").Value = "> C5"
    For i = 0 To glInformacionTrafico.FactorMeses.Factores.Rows.Count - 1
        For j = 0 To glInformacionTrafico.FactorMeses.Factores.Columns.Count
- 1
            .Cells(6 + i, 9 + j) =
glInformacionTrafico.FactorMeses.Factores.Rows(i).Item(j)
        Next
    Next
End With
'HOJA 3: ESPECTRO DE CARGAS
hoja = excel.Sheets(3)
With hoja

```

```

'CAMBIAR NOMBRE
.Name = "ESPECTRO DE CARGAS"
'ANCHOS DE COLUMNA
.Columns("A").ColumnWidth = 4
.Columns("B").ColumnWidth = 4
.Columns("C").ColumnWidth = 15
.Columns("K").ColumnWidth = 4
'COLOR DE FONDO TITULOS
.Range("B2", "K75").Interior.Color = Color.White
.Range("C3", "D3").Interior.Color = RGB(0, 20, 50)
.Range("C26", "D26").Interior.Color = RGB(0, 20, 50)
.Range("C49", "D49").Interior.Color = RGB(0, 20, 50)
'COLOR DE FUENTE TITULOS
.Range("C3", "D3").Font.Color = RGB(255, 255, 255)
.Range("C26", "D26").Font.Color = RGB(255, 255, 255)
.Range("C49", "D49").Font.Color = RGB(255, 255, 255)
'COMBINAR CELDAS
.Range("C3", "D3").Merge()
.Range("C26", "D26").Merge()
.Range("C49", "D49").Merge()
'FUENTE EN NEGRITA
.Range("C3", "C74").Font.Bold = True
.Range("D6", "J6").Font.Bold = True
.Range("D29", "J29").Font.Bold = True
.Range("D52", "J52").Font.Bold = True
'TEXTO CENTRADO
.Range("B2", "K75").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
'BORDES
.Range("B2", "K75").Borders(XlBordersIndex.xlEdgeTop).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
.Range("B2", "K75").Borders(XlBordersIndex.xlEdgeBottom).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
.Range("B2", "K75").Borders(XlBordersIndex.xlEdgeLeft).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
.Range("B2", "K75").Borders(XlBordersIndex.xlEdgeRight).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
.Range("C3", "D4").Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
.Range("C26", "D27").Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
.Range("C49", "D50").Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
For i = 6 To 24
.Cells(i, 3).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
.Cells(i, 4).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
Next
For i = 29 To 47
.Cells(i, 3).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
.Cells(i, 4).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
Next

```

```

        For i = 52 To 74
            .Cells(i, 3).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
            .Cells(i, 4).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        Next
        'TITULOS Y DATOS
        .Range("C3").Value = "EJE SIMPLE"
        .Range("C4").Value = "Tipo de espectro"
        .Range("C6").Value = "CARGA [KN]"
        .Range("C26").Value = "EJE TANDEM"
        .Range("C27").Value = "Tipo de espectro"
        .Range("C29").Value = "CARGA [KN]"
        .Range("C49").Value = "EJE TRIDEM"
        .Range("C50").Value = "Tipo de espectro"
        .Range("C52").Value = "CARGA [KN]"
        For i = 1 To 18
            .Cells(6 + i, 3) = CStr((i) * 10) + " - " + CStr((i + 1) * 10)
        Next
        For i = 1 To 18
            .Cells(29 + i, 3) = CStr(((2 * i) + 1) * 10) + " - " + CStr(((2 * (i
+ 1)) + 1) * 10)
        Next
        For i = 1 To 22
            .Cells(52 + i, 3) = CStr(((2 * i) + 1) * 10) + " - " + CStr(((2 * (i
+ 1)) + 1) * 10)
        Next
        'DATOS Y TITULOS DE LOS ESPECTROS
        'SIMPLE
        If glInformacionTrafico.EspectroCarga.TipoEspectroSimple = "Por clase"
Then
            .Range("D4").Value = "Por clase"
            .Range("D6").Value = "BUS"
            .Range("E6").Value = "C2P"
            .Range("F6").Value = "C2G"
            .Range("G6").Value = "C3"
            .Range("H6").Value = "C4"
            .Range("I6").Value = "C5"
            .Range("J6").Value = "> C5"
            For i = 0 To
glInformacionTrafico.EspectroCarga.EspectroSimplePorClase.Rows.Count - 1
                For j = 0 To
glInformacionTrafico.EspectroCarga.EspectroSimplePorClase.Columns.Count - 1
                    .Cells(7 + i, 4 + j) =
glInformacionTrafico.EspectroCarga.EspectroSimplePorClase.Rows(i).Item(j)
                Next
            Next
            For i = 6 To 24
                .Cells(i, 5).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
                .Cells(i, 6).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
                .Cells(i, 7).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous

```

```

        .Cells(i, 8).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 9).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        .Cells(i, 10).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    Next
    ElseIf glInformacionTrafico.EspectroCarga.TipoEspectroSimple = "General"
Then
        .Range("D4").Value = "General"
        .Range("D6").Value = "General"
        For i = 0 To
glInformacionTrafico.EspectroCarga.EspectroSimpleGeneral.Rows.Count - 1
            For j = 0 To
glInformacionTrafico.EspectroCarga.EspectroSimpleGeneral.Columns.Count - 1
                .Cells(7 + i, 4 + j) =
glInformacionTrafico.EspectroCarga.EspectroSimpleGeneral.Rows(i).Item(j)
            Next
        Next
    End If
'TANDEM
    If glInformacionTrafico.EspectroCarga.TipoEspectroTandem = "Por clase"
Then
        .Range("D27").Value = "Por clase"
        .Range("D29").Value = "BUS"
        .Range("E29").Value = "C2P"
        .Range("F29").Value = "C2G"
        .Range("G29").Value = "C3"
        .Range("H29").Value = "C4"
        .Range("I29").Value = "C5"
        .Range("J29").Value = "> C5"
        For i = 0 To
glInformacionTrafico.EspectroCarga.EspectroTandemPorClase.Rows.Count - 1
            For j = 0 To
glInformacionTrafico.EspectroCarga.EspectroTandemPorClase.Columns.Count - 1
                .Cells(30 + i, 4 + j) =
glInformacionTrafico.EspectroCarga.EspectroTandemPorClase.Rows(i).Item(j)
            Next
        Next
        For i = 29 To 47
            .Cells(i, 5).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
            .Cells(i, 6).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
            .Cells(i, 7).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
            .Cells(i, 8).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
            .Cells(i, 9).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
            .Cells(i, 10).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
        Next
    ElseIf glInformacionTrafico.EspectroCarga.TipoEspectroTandem = "General"
Then

```

```

        .Range("D27").Value = "General"
        .Range("D29").Value = "General"
    For i = 0 To
glInformacionTrafico.EspectroCarga.EspectroTandemGeneral.Rows.Count - 1
        For j = 0 To
glInformacionTrafico.EspectroCarga.EspectroTandemGeneral.Columns.Count - 1
            .Cells(30 + i, 4 + j) =
glInformacionTrafico.EspectroCarga.EspectroTandemGeneral.Rows(i).Item(j)
        Next
    Next
End If
'TRIDEM
If glInformacionTrafico.EspectroCarga.TipoEspectroTridem = "Por clase"
Then
    .Range("D50").Value = "Por clase"
    .Range("D52").Value = "BUS"
    .Range("E52").Value = "C2P"
    .Range("F52").Value = "C2G"
    .Range("G52").Value = "C3"
    .Range("H52").Value = "C4"
    .Range("I52").Value = "C5"
    .Range("J52").Value = "> C5"
    For i = 0 To
glInformacionTrafico.EspectroCarga.EspectroTridemPorClase.Rows.Count - 1
        For j = 0 To
glInformacionTrafico.EspectroCarga.EspectroTridemPorClase.Columns.Count - 1
            .Cells(53 + i, 4 + j) =
glInformacionTrafico.EspectroCarga.EspectroTridemPorClase.Rows(i).Item(j)
        Next
    Next
    For i = 52 To 74
        .Cells(i, 5).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous 'Asignar bordes
        .Cells(i, 6).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous 'Asignar bordes
        .Cells(i, 7).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous 'Asignar bordes
        .Cells(i, 8).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous 'Asignar bordes
        .Cells(i, 9).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous 'Asignar bordes
        .Cells(i, 10).Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous 'Asignar bordes
    Next
    ElseIf glInformacionTrafico.EspectroCarga.TipoEspectroTridem = "General"
Then
        .Range("D50").Value = "General"
        .Range("D52").Value = "General"
        For i = 0 To
glInformacionTrafico.EspectroCarga.EspectroTridemGeneral.Rows.Count - 1
            For j = 0 To
glInformacionTrafico.EspectroCarga.EspectroTridemGeneral.Columns.Count - 1
                .Cells(53 + i, 4 + j) =
glInformacionTrafico.EspectroCarga.EspectroTridemGeneral.Rows(i).Item(j)
            Next
        Next

```

```

        Next
    End If
End With
'HOJA 4: CRECIMIENTO
hoja = excel.Sheets(4)
With hoja
    'CAMBIAR NOMBRE
    .Name = "CRECIMIENTO"
    'ANCHOS DE COLUMNA
    .Columns("A").ColumnWidth = 4
    .Columns("B").ColumnWidth = 4
    .Columns("C").ColumnWidth = 14.5
    .Columns("E").ColumnWidth = 4
    'COLOR DE FONDO TITULOS
    .Range("B2", "E15").Interior.Color = Color.White
    .Range("C3", "D3").Interior.Color = RGB(0, 20, 50)
    'COLOR DE FUENTE TITULOS
    .Range("C3", "D3").Font.Color = RGB(255, 255, 255)
    'COMBINAR CELDAS
    .Range("C3", "D3").Merge()
    'FUENTE EN NEGRITA
    .Range("C3", "D3").Font.Bold = True
    .Range("C7", "D7").Font.Bold = True
    'TEXTO CENTRADO
    .Range("B2", "E15").HorizontalAlignment =
Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
    'BORDES
    .Range("B2", "E15").Borders(XlBordersIndex.xlEdgeTop).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    .Range("B2", "E15").Borders(XlBordersIndex.xlEdgeBottom).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    .Range("B2", "E15").Borders(XlBordersIndex.xlEdgeLeft).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    .Range("B2", "E15").Borders(XlBordersIndex.xlEdgeRight).LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    .Range("C3", "D5").Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous 'Asignar bordes
    .Range("C7", "D7").Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous 'Asignar bordes
    'TITULOS Y DATOS
    .Range("C3").Value = "TASA DE CRECIMIENTO"
    .Range("C4").Value = "Tipo"
    .Range("C5").Value = "¿Tasa por clase?"
    .Range("D4").Value =
glInformacionTrafico.FactCrecimiento.TipoCrecimiento
    If glInformacionTrafico.FactCrecimiento.UnicoValor = True Then
        .Range("D5").Value = "NO"
        .Range("C7").Value = "Tasa General"
        .Range("D7").Value =
glInformacionTrafico.FactCrecimiento.FactorGeneral
    ElseIf glInformacionTrafico.FactCrecimiento.UnicoValor = False Then
        .Range("D5").Value = "SI"
        .Range("C7").Value = "CLASE"
        .Range("D7").Value = "Tasa [%]"
        .Range("C8").Value = "BUS"
    End If
End With

```

```

        .Range("C9").Value = "C2P"
        .Range("C10").Value = "C2G"
        .Range("C11").Value = "C3"
        .Range("C12").Value = "C4"
        .Range("C13").Value = "C5"
        .Range("C14").Value = "> C5"
    For i = 0 To
glInformacionTrafico.FactCrecimiento.FactorPorClase.Length - 1
        .Cells(8 + i, 4) =
glInformacionTrafico.FactCrecimiento.FactorPorClase(i)
    Next
        .Range("C8", "D14").Borders.LineStyle =
Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
    End If
End With
'CALCULO DE TPDM, CANTIDAD VEHICULOS, CANTIDAD DE EJES POR TIPO Y CANTIDAD
DE EJES POR CARGA
Dim TPDATreducido As Double
Dim VidaDiseño As Integer = CInt(glInformacionGeneral.VidaDiseño)
Dim AñoInicial As Integer
Dim MesInicial As Integer
Dim PorcentajeClases As Double()
Dim PorcentajeHoras As Double()
Dim FactoresMeses As Double(,)
Dim NumeroEjesSimplesClase As Double()
Dim NumeroEjesTandemClase As Double()
Dim NumeroEjesTridemClase As Double()
Dim EspectroSimple(,) As Double
Dim EspectroTandem(,) As Double
Dim EspectroTridem(,) As Double
Dim EspectroTandemT(,) As Double
Dim EspectroTridemT(,) As Double
Dim EspectroSimpleT(,) As Double
Dim TasasCrecimiento() As Double
Dim TipoCrecimiento As String =
glInformacionTrafico.FactCrecimiento.TipoCrecimiento
Dim VectorAños() As Integer
Dim VectorMeses() As String
Dim VectorHoras() As String
ReDim VectorAños(VidaDiseño - 1)
ReDim VectorMeses(11)
ReDim VectorHoras(23)
With glInformacionTrafico
    TPDATreducido = .TPDAT * (.FactorDir / 100) * (.FactorCarril / 100)
    AñoInicial = glInformacionGeneral.FechaApertTrafico.Year
    For i = 0 To VidaDiseño - 1
        VectorAños(i) = AñoInicial + i
    Next
    For i = 0 To VectorMeses.Length - 1
        VectorMeses(i) = MonthName(i + 1)
    Next
    PorcentajeClases = .DistrClase.Porcentajes
    PorcentajeHoras = .DistrHoraria.Porcentajes

```

```

For i = 0 To 23
    VectorHoras(i) = CStr(i) & ":00"
Next
FactoresMeses = DataTable2Matriz(.FactorMeses.Factores)
NumeroEjesSimplesClase = ExtraerColumnaDataTable(.NumEjesClase, 0)
NumeroEjesTandemClase = ExtraerColumnaDataTable(.NumEjesClase, 1)
NumeroEjesTridemClase = ExtraerColumnaDataTable(.NumEjesClase, 2)
ReDim EspectroSimple(.EspectroCarga.EspectroSimpleGeneral.Rows.Count -
1, 6)
If .EspectroCarga.TipoEspectroSimple = "Por clase" Then
    EspectroSimple =
DataTable2Matriz(.EspectroCarga.EspectroSimplePorClase)
ElseIf .EspectroCarga.TipoEspectroSimple = "General" Then
    Dim EspectroSimpleGral() As Double
    EspectroSimpleGral =
DataTable2Vector(.EspectroCarga.EspectroSimpleGeneral)
    For i = 0 To EspectroSimpleGral.Length - 1
        For j = 0 To 6
            EspectroSimple(i, j) = EspectroSimpleGral(i)
        Next
    Next
End If
EspectroSimpleT = TransponerMatriz(EspectroSimple)
ReDim EspectroTandem(.EspectroCarga.EspectroTandemGeneral.Rows.Count -
1, 6)
If .EspectroCarga.TipoEspectroTandem = "Por clase" Then
    EspectroTandem =
DataTable2Matriz(.EspectroCarga.EspectroTandemPorClase)
ElseIf .EspectroCarga.TipoEspectroTandem = "General" Then
    Dim EspectroTandemGral() As Double
    EspectroTandemGral =
DataTable2Vector(.EspectroCarga.EspectroTandemGeneral)
    For i = 0 To EspectroTandemGral.Length - 1
        For j = 0 To 6
            EspectroTandem(i, j) = EspectroTandemGral(i)
        Next
    Next
End If
EspectroTandemT = TransponerMatriz(EspectroTandem)
ReDim EspectroTridem(.EspectroCarga.EspectroTridemGeneral.Rows.Count -
1, 6)
If .EspectroCarga.TipoEspectroTridem = "Por clase" Then
    EspectroTridem =
DataTable2Matriz(.EspectroCarga.EspectroTridemPorClase)
ElseIf .EspectroCarga.TipoEspectroTridem = "General" Then
    Dim EspectroTridemGral() As Double
    EspectroTridemGral =
DataTable2Vector(.EspectroCarga.EspectroTridemGeneral)
    For i = 0 To EspectroTridemGral.Length - 1
        For j = 0 To 6
            EspectroTridem(i, j) = EspectroTridemGral(i)
        Next
    Next
End If
EspectroTridemT = TransponerMatriz(EspectroTridem)

```

```

ReDim TasasCrecimiento(6)
If .FactCrecimiento.UnicoValor = True Then
    For i = 0 To 6
        TasasCrecimiento(i) = .FactCrecimiento.FactorGeneral
    Next
Else
    For i = 0 To 6
        TasasCrecimiento(i) = .FactCrecimiento.FactorPorClase(i)
    Next
End If
End With
Dim MatrizTPDM(VidaDiseño - 1, 11, 6)
Dim MatrizNumVehiculos(VidaDiseño - 1, 11, 23, 6) As Double
Dim MatrizNumEjeSim(VidaDiseño - 1, 11, 23, 6) As Double
Dim MatrizNumEjeTan(VidaDiseño - 1, 11, 23, 6) As Double
Dim MatrizNumEjeTri(VidaDiseño - 1, 11, 23, 6) As Double
Dim MatrizNumEjeSimCarga(VidaDiseño - 1, 11, 23, UBound(EspectroSimpleT, 2))
As Double
Dim MatrizNumEjeTanCarga(VidaDiseño - 1, 11, 23, UBound(EspectroTandemT, 2))
As Double
Dim MatrizNumEjeTriCarga(VidaDiseño - 1, 11, 23, UBound(EspectroTridemT, 2))
As Double
For i = 0 To VidaDiseño - 1
    For j = 0 To 11
        For k = 0 To 6
            If TipoCrecimiento = "Ninguno" Then
                MatrizTPDM(i, j, k) = TPDATreducido * (PorcentajeClases(k) /
100) * FactoresMeses(j, k)
            ElseIf TipoCrecimiento = "Lineal" Then
                MatrizTPDM(i, j, k) = TPDATreducido + (i * (TPDATreducido *
TasasCrecimiento(k) / 100)) * (PorcentajeClases(k) / 100) * FactoresMeses(j, k)
            ElseIf TipoCrecimiento = "Compuesto" Then
                MatrizTPDM(i, j, k) = TPDATreducido * Math.Pow((1 +
TasasCrecimiento(k) / 100), i) * (PorcentajeClases(k) / 100) * FactoresMeses(j, k)
            End If
        Next
    Next
Next
For i = 0 To VidaDiseño - 1
    For j = 0 To 11
        For k = 0 To 23
            For l = 0 To 6
                If TipoCrecimiento = "Ninguno" Then
                    MatrizNumVehiculos(i, j, k, l) = TPDATreducido *
(PorcentajeClases(l) / 100) * (PorcentajeHoras(k) / 100) * FactoresMeses(j, l)
                ElseIf TipoCrecimiento = "Lineal" Then
                    MatrizNumVehiculos(i, j, k, l) = TPDATreducido + (i *
(TPDATreducido * TasasCrecimiento(k) / 100)) * (PorcentajeClases(l) / 100) *
(PorcentajeHoras(k) / 100) * FactoresMeses(j, l)
                ElseIf TipoCrecimiento = "Compuesto" Then
                    MatrizNumVehiculos(i, j, k, l) = TPDATreducido *
Math.Pow((1 + TasasCrecimiento(k) / 100), i) * (PorcentajeClases(l) / 100) *
(PorcentajeHoras(k) / 100) * FactoresMeses(j, l)
                End If
            End If
        Next
    Next
Next

```

```

        MatrizNumEjeSim(i, j, k, l) = MatrizNumVehiculos(i, j, k, l)
* NumeroEjesSimplesClase(l)
        MatrizNumEjeTan(i, j, k, l) = MatrizNumVehiculos(i, j, k, l)
* NumeroEjesTandemClase(l)
        MatrizNumEjeTri(i, j, k, l) = MatrizNumVehiculos(i, j, k, l)
* NumeroEjesTridemClase(l)
        Next
    Next
Next
Dim temp As Double = 0
For i = 0 To VidaDiseño - 1
    For j = 0 To 11
        For k = 0 To 23
            For l = 0 To UBound(EspectroSimpleT, 2)
                temp = 0
                For m = 0 To 6
                    temp += MatrizNumEjeSim(i, j, k, m) * EspectroSimpleT(m,
1)
                Next
                MatrizNumEjeSimCarga(i, j, k, l) = temp
            Next
        Next
    Next
Next
For i = 0 To VidaDiseño - 1
    For j = 0 To 11
        For k = 0 To 23
            For l = 0 To UBound(EspectroTandemT, 2)
                temp = 0
                For m = 0 To 6
                    temp += MatrizNumEjeTan(i, j, k, m) * EspectroTandemT(m,
1)
                Next
                MatrizNumEjeTanCarga(i, j, k, l) = temp
            Next
        Next
    Next
Next
For i = 0 To VidaDiseño - 1
    For j = 0 To 11
        For k = 0 To 23
            For l = 0 To UBound(EspectroTridemT, 2)
                temp = 0
                For m = 0 To 6
                    temp += MatrizNumEjeTri(i, j, k, m) * EspectroTridemT(m,
1)
                Next
                MatrizNumEjeTriCarga(i, j, k, l) = temp
            Next
        Next
    Next
Next
'HOJA 5: PROYECCIÓN TPDM
hoja = excel.Sheets(5)

```

```

With hoja
'CAMBIAR NOMBRE
.Name = "PROY. TPDM"
'ANCHOS DE COLUMNA
.Columns("A").ColumnWidth = 4
.Columns("C").ColumnWidth = 11
'COLOR DE FONDO TITULOS
.Range("D2", "J2").Interior.Color = RGB(0, 20, 50)
'COLOR DE FUENTE TITULOS
.Range("D2", "J2").Font.Color = RGB(255, 255, 255)
'FUENTE EN NEGRITA
.Range("B2", "J4").Font.Bold = True
'COMBINAR CELDAS
.Range("D2", "J2").Merge()
.Range("B2", "B4").Merge()
.Range("C2", "C4").Merge()
.Range("D3", "J3").Merge()
'TEXTO CENTRADO
.Range(.Cells(2, 2), .Cells(4 + VidaDiseño * 12,
10)).HorizontalAlignment = Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
.Range("B3", "B4").VerticalAlignment =
Microsoft.Office.Interop.Excel.XlVAlign.xlVAlignCenter
.Range("C3", "C4").VerticalAlignment =
Microsoft.Office.Interop.Excel.XlVAlign.xlVAlignCenter
'BORDES
.Range(.Cells(2, 2), .Cells(4 + VidaDiseño * 12, 10)).Borders.LineStyle
= Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
'TITULOS Y DATOS
.Range("D2").Value = "TPD MENSUAL PROYECTADO"
.Range("B2").Value = "AÑO"
.Range("C2").Value = "MES"
.Range("D3").Value = "CLASE"
.Range("D4").Value = "BUS"
.Range("E4").Value = "C2P"
.Range("F4").Value = "C2G"
.Range("G4").Value = "C3"
.Range("H4").Value = "C4"
.Range("I4").Value = "C5"
.Range("J4").Value = "> C5"
Dim Matriz12Meses7Clases(11, 6) As Double
For i = 0 To VidaDiseño - 1
    For j = 0 To 11
        .Cells((5 + j) + (i * 12), 2) = VectorAños(i)
        .Cells((5 + j) + (i * 12), 3) = VectorMeses(j)
        For k = 0 To 6
            Matriz12Meses7Clases(j, k) = MatrizTPDM(i, j, k)
        Next
    Next
    .Range(.Cells(5 + (12 * i), 4), .Cells(5 + (12 * i), 4)).Resize(12,
7).Value = Matriz12Meses7Clases
Next
End With
'HOJA 6: PROYECCIÓN TPDH
hoja = excel.Sheets(6)
With hoja

```

```

'CAMBIAR NOMBRE
.Name = "PROY. TPDH Y N° DE EJES"
'ANCHOS DE COLUMNA
.Columns("A").ColumnWidth = 4
.Columns("C").ColumnWidth = 11
'COLOR DE FONDO TITULOS
.Range("E2", "AF2").Interior.Color = RGB(0, 20, 50)
'COLOR DE FUENTE TITULOS
.Range("E2", "AF2").Font.Color = RGB(255, 255, 255)
'FUENTE EN NEGRITA
.Range("B2", "AF4").Font.Bold = True
'COMBINAR CELDAS
.Range("E2", "K2").Merge()
.Range("L2", "AF2").Merge()
.Range("B2", "B4").Merge()
.Range("C2", "C4").Merge()
.Range("D2", "D4").Merge()
.Range("E3", "K3").Merge()
.Range("L3", "R3").Merge()
.Range("S3", "Y3").Merge()
.Range("Z3", "AF3").Merge()
'TEXTO CENTRADO
.Range(.Cells(2, 2), .Cells(4 + VidaDiseño * 12 * 24,
32)).HorizontalAlignment = Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
.Range("B2", "B4").VerticalAlignment =
Microsoft.Office.Interop.Excel.XlVAlign.xlVAlignCenter
.Range("C2", "C4").VerticalAlignment =
Microsoft.Office.Interop.Excel.XlVAlign.xlVAlignCenter
.Range("D2", "D4").VerticalAlignment =
Microsoft.Office.Interop.Excel.XlVAlign.xlVAlignCenter
'BORDES
.Range(.Cells(2, 2), .Cells(4 + VidaDiseño * 12 * 24,
32)).Borders.LineStyle = Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
'TITULOS Y DATOS
.Range("E2").Value = "TPD HORARIO PROYECTADO"
.Range("L2").Value = "NUMERO DE EJES PROYECTADO"
.Range("B2").Value = "AÑO"
.Range("C2").Value = "MES"
.Range("D2").Value = "HORA"
.Range("E3").Value = "CLASE"
.Range("L3").Value = "EJES SIMPLES"
.Range("S3").Value = "EJES TANDEM"
.Range("Z3").Value = "EJES TRIDEM"
.Range("E4").Value = "BUS"
.Range("F4").Value = "C2P"
.Range("G4").Value = "C2G"
.Range("H4").Value = "C3"
.Range("I4").Value = "C4"
.Range("J4").Value = "C5"
.Range("K4").Value = "> C5"
.Range("L4").Value = "BUS"
.Range("M4").Value = "C2P"
.Range("N4").Value = "C2G"
.Range("O4").Value = "C3"
.Range("P4").Value = "C4"

```

```

.Range("Q4").Value = "C5"
.Range("R4").Value = "> C5"
.Range("S4").Value = "BUS"
.Range("T4").Value = "C2P"
.Range("U4").Value = "C2G"
.Range("V4").Value = "C3"
.Range("W4").Value = "C4"
.Range("X4").Value = "C5"
.Range("Y4").Value = "> C5"
.Range("Z4").Value = "BUS"
.Range("AA4").Value = "C2P"
.Range("AB4").Value = "C2G"
.Range("AC4").Value = "C3"
.Range("AD4").Value = "C4"
.Range("AE4").Value = "C5"
.Range("AF4").Value = "> C5"
Dim Matriz24Horas7ClasesNumV(23, 6) As Double
Dim Matriz24Horas7ClasesNumESim(23, 6) As Double
Dim Matriz24Horas7ClasesNumETan(23, 6) As Double
Dim Matriz24Horas7ClasesNumETri(23, 6) As Double
For i = 0 To VidaDiseño - 1
    For j = 0 To 11
        For k = 0 To 23
            .Cells((24 * 12 * i) + (j * 24) + (5 + k), 2) =
VectorAños(i)
            .Cells((24 * 12 * i) + (j * 24) + (5 + k), 3) =
VectorMeses(j)
            .Cells((24 * 12 * i) + (j * 24) + (5 + k), 4) =
VectorHoras(k)
            For l = 0 To 6
                Matriz24Horas7ClasesNumV(k, l) = MatrizNumVehiculos(i,
j, k, l)
                Matriz24Horas7ClasesNumESim(k, l) = MatrizNumEjeSim(i,
j, k, l)
                Matriz24Horas7ClasesNumETan(k, l) = MatrizNumEjeTan(i,
j, k, l)
                Matriz24Horas7ClasesNumETri(k, l) = MatrizNumEjeTri(i,
j, k, l)
            Next
        Next
        .Range(.Cells((24 * 12 * i) + (j * 24) + 5, 5), .Cells((24 * 12
* i) + (j * 24) + 5, 5)).Resize(24, 7).Value = Matriz24Horas7ClasesNumV
        .Range(.Cells((24 * 12 * i) + (j * 24) + 5, 12), .Cells((24 * 12
* i) + (j * 24) + 5, 12)).Resize(24, 7).Value = Matriz24Horas7ClasesNumV
        .Range(.Cells((24 * 12 * i) + (j * 24) + 5, 19), .Cells((24 * 12
* i) + (j * 24) + 5, 19)).Resize(24, 7).Value = Matriz24Horas7ClasesNumV
        .Range(.Cells((24 * 12 * i) + (j * 24) + 5, 26), .Cells((24 * 12
* i) + (j * 24) + 5, 26)).Resize(24, 7).Value = Matriz24Horas7ClasesNumV
    Next
Next
End With
'HOJA 7: PROYECCIÓN ESPECTROS
hoja = excel.Sheets(7)
With hoja

```

```

'CAMBIAR NOMBRE
.Name = "PROY. ESPECTROS"
'ANCHOS DE COLUMNA
.Columns("A").ColumnWidth = 4
.Columns("C").ColumnWidth = 11
'COLOR DE FONDO TITULOS
.Range("E2", "BJ2").Interior.Color = RGB(0, 20, 50)
'COLOR DE FUENTE TITULOS
.Range("E2", "BJ2").Font.Color = RGB(255, 255, 255)
'FUENTE EN NEGRITA
.Range("B2", "BJ4").Font.Bold = True
'COMBINAR CELDAS
.Range("E2", "BJ2").Merge()
.Range("B2", "B4").Merge()
.Range("C2", "C4").Merge()
.Range("D2", "D4").Merge()
.Range("E3", "V3").Merge()
.Range("W3", "AN3").Merge()
.Range("A03", "BJ3").Merge()
'TEXTO CENTRADO
.Range(.Cells(2, 2), .Cells(4 + VidaDiseño * 12 * 24,
62)).HorizontalAlignment = Microsoft.Office.Interop.Excel.XlHAlign.xlHAlignCenter
.Range("B2", "B4").VerticalAlignment =
Microsoft.Office.Interop.Excel.XlVAlign.xlVAlignCenter
.Range("C2", "C4").VerticalAlignment =
Microsoft.Office.Interop.Excel.XlVAlign.xlVAlignCenter
.Range("D2", "D4").VerticalAlignment =
Microsoft.Office.Interop.Excel.XlVAlign.xlVAlignCenter
'BORDES
.Range(.Cells(2, 2), .Cells(4 + VidaDiseño * 12 * 24,
62)).Borders.LineStyle = Microsoft.Office.Interop.Excel.XlLineStyle.xlContinuous
'TITULOS Y DATOS
.Range("E2").Value = "ESPECTRO DE CARGA PROYECTADO"
.Range("B2").Value = "AÑO"
.Range("C2").Value = "MES"
.Range("D2").Value = "HORA"
.Range("E3").Value = "EJES SIMPLES"
.Range("W3").Value = "EJES TANDEM"
.Range("A03").Value = "EJES TRIDEM"
For i = 1 To 18
    .Cells(4, 4 + i) = CStr((i) * 10) + " - " + CStr((i + 1) * 10)
Next
For i = 1 To 18
    .Cells(4, 22 + i) = CStr(((2 * i) + 1) * 10) + " - " + CStr(((2 * (i
+ 1)) + 1) * 10)
Next
For i = 1 To 22
    .Cells(4, 40 + i) = CStr(((2 * i) + 1) * 10) + " - " + CStr(((2 * (i
+ 1)) + 1) * 10)
Next
Dim Matriz24Horas18CargasSim(23, 17) As Double
Dim Matriz24Horas18CargasTan(23, 17) As Double
For i = 0 To VidaDiseño - 1
    For j = 0 To 11
        For k = 0 To 23

```

```

        .Cells((24 * 12 * i) + (j * 24) + (5 + k), 2) =
VectorAños(i)
        .Cells((24 * 12 * i) + (j * 24) + (5 + k), 3) =
VectorMeses(j)
        .Cells((24 * 12 * i) + (j * 24) + (5 + k), 4) =
VectorHoras(k)
        For l = 0 To 17
            Matriz24Horas18CargasSim(k, l) = MatrizNumEjeSimCarga(i,
j, k, l)
            Matriz24Horas18CargasTan(k, l) = MatrizNumEjeTanCarga(i,
j, k, l)
        Next
    Next
    .Range(.Cells((24 * 12 * i) + (j * 24) + 5, 5), .Cells((24 * 12
* i) + (j * 24) + 5, 5)).Resize(24, 18).Value = Matriz24Horas18CargasSim
    .Range(.Cells((24 * 12 * i) + (j * 24) + 5, 23), .Cells((24 * 12
* i) + (j * 24) + 5, 23)).Resize(24, 18).Value = Matriz24Horas18CargasTan
    Next
    Next
    Dim Matriz24Horas22CargasTri(23, 21) As Double
    For i = 0 To VidaDiseño - 1
        For j = 0 To 11
            For k = 0 To 23
                For l = 0 To 21
                    Matriz24Horas22CargasTri(k, l) = MatrizNumEjeTriCarga(i,
j, k, l)
                Next
            Next
            .Range(.Cells((24 * 12 * i) + (j * 24) + 5, 41), .Cells((24 * 12
* i) + (j * 24) + 5, 41)).Resize(24, 22).Value = Matriz24Horas22CargasTri
        Next
    Next
    End With
    excel.Visible = True
    excel = Nothing
    libro = Nothing
    hoja = Nothing
End Sub
Private Sub btnGuardarInfoGeneral_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnGuardarInfoTransito.Click
    'EXTRAER INFORMACIÓN A CONTROLES
    RecopilarInformacionTransito()
    If saveTrafico.ShowDialog() = DialogResult.OK Then
        strmTrafico = saveTrafico.OpenFile()
        Using strmW As StreamWriter = New StreamWriter(strmTrafico)
            srlzTrafico.Serialize(strmW, glInformacionTrafico)
            strmW.Close()
        End Using
        If (strmTrafico IsNot Nothing) Then
            strmTrafico.Close()
        End If
    End If
End Sub

```

```

Private Sub btnCargarInfoGeneral_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles btnCargarInfoTransito.Click
    If openTrafico.ShowDialog() = System.Windows.Forms.DialogResult.OK Then
        Try
            strmTrafico = openTrafico.OpenFile()
            Dim strmR As New StreamReader(strmTrafico)
            glInformacionTrafico = srlzTrafico.Deserialize(strmR)
            strmR.Close()
            'ASIGNAR INFORMACIÓN A CONTROLES
            With glInformacionTrafico
                txtTPDinicial.Text = CStr(.TPDAT)
                txtCarriles.Text = CStr(.NumCarril)
                txtFactorDireccional.Text = CStr(.FactorDir)
                txtPorcentajeCarril.Text = CStr(.FactorCarril)
                txtVelocidad.Text = CStr(.VelOper)
                VolCarga.txtBus.Text = CStr(.DistrClase.Porcentajes(0))
                VolCarga.txtC2P.Text = CStr(.DistrClase.Porcentajes(1))
                VolCarga.txtC2G.Text = CStr(.DistrClase.Porcentajes(2))
                VolCarga.txtC3.Text = CStr(.DistrClase.Porcentajes(3))
                VolCarga.txtC4.Text = CStr(.DistrClase.Porcentajes(4))
                VolCarga.txtC5.Text = CStr(.DistrClase.Porcentajes(5))
                VolCarga.txtMayorC5.Text = CStr(.DistrClase.Porcentajes(6))
                VolCarga.txt12am.Text = CStr(.DistrHoraria.Porcentajes(0))
                VolCarga.txt1am.Text = CStr(.DistrHoraria.Porcentajes(1))
                VolCarga.txt2am.Text = CStr(.DistrHoraria.Porcentajes(2))
                VolCarga.txt3am.Text = CStr(.DistrHoraria.Porcentajes(3))
                VolCarga.txt4am.Text = CStr(.DistrHoraria.Porcentajes(4))
                VolCarga.txt5am.Text = CStr(.DistrHoraria.Porcentajes(5))
                VolCarga.txt6am.Text = CStr(.DistrHoraria.Porcentajes(6))
                VolCarga.txt7am.Text = CStr(.DistrHoraria.Porcentajes(7))
                VolCarga.txt8am.Text = CStr(.DistrHoraria.Porcentajes(8))
                VolCarga.txt9am.Text = CStr(.DistrHoraria.Porcentajes(9))
                VolCarga.txt10am.Text = CStr(.DistrHoraria.Porcentajes(10))
                VolCarga.txt11am.Text = CStr(.DistrHoraria.Porcentajes(11))
                VolCarga.txt12m.Text = CStr(.DistrHoraria.Porcentajes(12))
                VolCarga.txt1pm.Text = CStr(.DistrHoraria.Porcentajes(13))
                VolCarga.txt2pm.Text = CStr(.DistrHoraria.Porcentajes(14))
                VolCarga.txt3pm.Text = CStr(.DistrHoraria.Porcentajes(15))
                VolCarga.txt4pm.Text = CStr(.DistrHoraria.Porcentajes(16))
                VolCarga.txt5pm.Text = CStr(.DistrHoraria.Porcentajes(17))
                VolCarga.txt6pm.Text = CStr(.DistrHoraria.Porcentajes(18))
                VolCarga.txt7pm.Text = CStr(.DistrHoraria.Porcentajes(19))
                VolCarga.txt8pm.Text = CStr(.DistrHoraria.Porcentajes(20))
                VolCarga.txt9pm.Text = CStr(.DistrHoraria.Porcentajes(21))
                VolCarga.txt10pm.Text = CStr(.DistrHoraria.Porcentajes(22))
                VolCarga.txt11pm.Text = CStr(.DistrHoraria.Porcentajes(23))
                VolCarga.dgvDistrMes.DataSource = .FactorMeses.Factores
                If .EspectroCarga.TipoEspectroSimple = "General" Then
                    DistrCarga.rbbtnGeneralSimple.Checked = True
                Else
                    DistrCarga.rbbtnClasesSimple.Checked = True
                End If
                If .EspectroCarga.TipoEspectroTandem = "General" Then
                    DistrCarga.rbbtnGeneralTandem.Checked = True
                Else

```

```

        DistrCarga.rbtnClaseTandem.Checked = True
    End If
    If .EspectroCarga.TipoEspectroTridem = "General" Then
        DistrCarga.rbtnGeneralTridem.Checked = True
    Else
        DistrCarga.rbtnClaseTridem.Checked = True
    End If
    DistrCarga.dgvGeneralSimple.DataSource =
.EspectroCarga.EspectroSimpleGeneral
    DistrCarga.dgvGeneralTandem.DataSource =
.EspectroCarga.EspectroTandemGeneral
    DistrCarga.dgvGeneralTridem.DataSource =
.EspectroCarga.EspectroTridemGeneral
    DistrCarga.dgvClaseSimple.DataSource =
.EspectroCarga.EspectroSimplePorClase
    DistrCarga.dgvClaseTandem.DataSource =
.EspectroCarga.EspectroTandemPorClase
    DistrCarga.dgvClaseTridem.DataSource =
.EspectroCarga.EspectroTridemPorClase
    EntGrls.txtDistHuella.Text = CStr(.MediaHuella)
    EntGrls.txtDesvEst.Text = CStr(.DesvHuella)
    EntGrls.txtAnchoCarril.Text = CStr(.AnchoCarril)
    EntGrls.txtDistPromCorto.Text =
CStr(.Batalla.MediaEspaciamiento(0))
    EntGrls.txtDistPromMedio.Text =
CStr(.Batalla.MediaEspaciamiento(1))
    EntGrls.txtDistPromLargo.Text =
CStr(.Batalla.MediaEspaciamiento(2))
    EntGrls.txtPorcentajeCorto.Text =
CStr(.Batalla.PorcentajeCamiones(0))
    EntGrls.txtPorcentajeMedio.Text =
CStr(.Batalla.PorcentajeCamiones(1))
    EntGrls.txtPorcentajeLargo.Text =
CStr(.Batalla.PorcentajeCamiones(2))
    EntGrls.txtAnchoPromEje.Text = CStr(.ConfiguracionEje.AnchoEje)
    EntGrls.txtDistRuedaDoble.Text =
CStr(.ConfiguracionEje.DistGemelado)
    EntGrls.txtPresionInflado.Text =
CStr(.ConfiguracionEje.PresionInflado)
    EntGrls.txtEspTandem.Text =
CStr(.ConfiguracionEje.EspaciamientoTandem)
    EntGrls.txtEspTridem.Text =
CStr(.ConfiguracionEje.EspaciamientoTridem)
    EntGrls.dgvEjesClase.DataSource = .NumEjesClase
    If .FactCrecimiento.TipoCrecimiento = "Ninguno" Then
        FacCrec.rbtnNinguno.Checked = True
    ElseIf .FactCrecimiento.TipoCrecimiento = "Lineal" Then
        FacCrec.rbtnLineal.Checked = True
    ElseIf .FactCrecimiento.TipoCrecimiento = "Compuesto" Then
        FacCrec.rbtnCompuesto.Checked = True
    End If
    If .FactCrecimiento.UnicoValor = False Then
        FacCrec.checkPorClase.Checked = True

```

```

Else
    FacCrec.checkPorClase.Checked = False
End If
FacCrec.txtFactorGeneral.Text =
CStr(.FactCrecimiento.FactorGeneral)
FacCrec.txtBus.Text = CStr(.FactCrecimiento.FactorPorClase(0))
FacCrec.txtC2P.Text = CStr(.FactCrecimiento.FactorPorClase(1))
FacCrec.txtC2G.Text = CStr(.FactCrecimiento.FactorPorClase(2))
FacCrec.txtC3.Text = CStr(.FactCrecimiento.FactorPorClase(3))
FacCrec.txtC4.Text = CStr(.FactCrecimiento.FactorPorClase(4))
FacCrec.txtC5.Text = CStr(.FactCrecimiento.FactorPorClase(5))
FacCrec.txtMayorC5.Text =
CStr(.FactCrecimiento.FactorPorClase(6))
End With
If (strmTrafico IsNot Nothing) Then
    strmTrafico.Close()
End If
Catch Ex As Exception
    MessageBox.Show("No se puede leer el archivo. Causa: " & Ex.Message)
Finally
    If (strmTrafico IsNot Nothing) Then
        strmTrafico.Close()
    End If
End Try
End If
End Sub
Private Sub Transito_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    With glInformacionTrafico
        txtTPDinicial.Text = CStr(.TPDAT)
        txtCarriles.Text = CStr(.NumCarril)
        txtFactorDireccional.Text = CStr(.FactorDir)
        txtPorcentajeCarril.Text = CStr(.FactorCarril)
        txtVelocidad.Text = CStr(.VelOper)
    End With
End Sub
'ABRIR FORMS
Private Sub btnVolumen_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnVolumenTrafico.Click
    VolCarga.Show()
End Sub
Private Sub btnGeneral_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEntradaGeneral.Click
    EntGrIs.Show()
End Sub
Private Sub btnDistribucionCarga_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnDistrCargaEje.Click
    DistrCarga.Show()
End Sub
Private Sub btnCrecimiento_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCrecimiento.Click
    FacCrec.Show()
End Sub

```

```

    Private Sub btnCalcularTPDAC_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCalcularTPDAC.Click
        CalcularTPDAC.Show()
    End Sub
    Private Sub btnFactorCarril_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnFactorCarril.Click
        AyudaFactorCarril.Show()
    End Sub
    Private Sub btnFactorDireccional_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnFactorDireccional.Click
        AyudaFactorDireccional.Show()
    End Sub
    Private Sub btnVelocidad_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnVelocidad.Click
        AyudaVelocidadOperacion.Show()
    End Sub
    'BOTONES ACEPTAR/CANCELAR
    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub
    Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
        RecopilarInformacionTransito()
        Me.Close()
    End Sub
End Class
'EXPORTAR A ME-PDG
Private Sub btnMEPDG_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnMEPDG.Click
    Dim ruta As String
    If folderMEPDG.ShowDialog() = DialogResult.OK Then
        ruta = folderMEPDG.SelectedPath
    Else
        GoTo salida
    End If
    With glInformacionTrafico
        .TPDAT = CDb1(txtTPDinicial.Text)
        .NumCarril = CDb1(txtCarriles.Text)
        .FactorDir = CDb1(txtFactorDireccional.Text)
        .FactorCarril = CDb1(txtPorcentajeCarril.Text)
        .VelOper = CDb1(txtVelocidad.Text)
    End With
    Dim objStreamWriter As StreamWriter
    'ARCHIVO TRAFFIC.TXT
    objStreamWriter = New StreamWriter(ruta + "\Traffic.txt")
    objStreamWriter.WriteLine(CStr(glInformacionTrafico.TPDAT))
    objStreamWriter.WriteLine(CStr(glInformacionTrafico.NumCarril))
    objStreamWriter.WriteLine(CStr(glInformacionTrafico.FactorDir))
    objStreamWriter.WriteLine(CStr(glInformacionTrafico.FactorCarril))
    objStreamWriter.WriteLine(CStr(glInformacionTrafico.VelOper / 1.6))
    objStreamWriter.WriteLine("# Lines with # signs in them are optional and
for user information only.")
    objStreamWriter.WriteLine("#Line 1 - Initial two-way AADTT")
    objStreamWriter.WriteLine("#Line 2 - Number of lanes in the design
direction")

```

```

objStreamWriter.WriteLine("#Line 3 - Percent of trucks in the design
direction")
objStreamWriter.WriteLine("#Line 4 - Percent of trucks in the design
lane")
objStreamWriter.WriteLine("#Line 5 - Operational speed")
objStreamWriter.Close()
'ARCHIVO GENERALTRAFFIC.TXT
objStreamWriter = New StreamWriter(ruta + "\GeneralTraffic.txt")
objStreamWriter.WriteLine(CStr(.MediaHuella / 0.0254) + ",#MeanWheel
Location (inches from marking)")
objStreamWriter.WriteLine(CStr(.DesvHuella / 0.0254) + ",#Traffic Wander
(in)")
objStreamWriter.WriteLine(CStr(.AnchoCarril / 0.3048) + ",#Design lane
width feet")
objStreamWriter.WriteLine(CStr(.ConfiguracionEje.AnchoEje / 0.3048) +
",,#Average Axle Width(edge to edge) Outside dimensions (ft)")
objStreamWriter.WriteLine(CStr(.ConfiguracionEje.DistGemelado / 0.0254)
+ ",#Dual Tire Spacing (in)")
objStreamWriter.WriteLine(CStr(.ConfiguracionEje.PresionInflado *
0.14504) + ",#Tire Pressure")
objStreamWriter.WriteLine(CStr(.ConfiguracionEje.EspaciamientoTandem /
0.0254) + ",#Tandem axle spacing (in)")
objStreamWriter.WriteLine(CStr(.ConfiguracionEje.EspaciamientoTridem /
0.0254) + ",#Tridem axle spacing(in)")
objStreamWriter.WriteLine(CStr(0) + ",#Quad axle spacing(in)")
objStreamWriter.WriteLine(CStr(.Batalla.MediaEspaciamiento(0) / 0.3048)
+ ",#Short axle spacing(ft)")
objStreamWriter.WriteLine(CStr(.Batalla.PorcentajeCamiones(0)) +
",#Percent trucks with short axle spacing")
objStreamWriter.WriteLine(CStr(.Batalla.MediaEspaciamiento(1) / 0.3048)
+ ",#Medium axle spacing(ft)")
objStreamWriter.WriteLine(CStr(.Batalla.PorcentajeCamiones(1)) +
",#Percent of trucks with medium axle spacing")
objStreamWriter.WriteLine(CStr(.Batalla.MediaEspaciamiento(2) / 0.3048)
+ ",#Long axle spacing(ft)")
objStreamWriter.WriteLine(CStr(.Batalla.PorcentajeCamiones(2)) +
",#Percent of trucks with long axle spacing")
objStreamWriter.WriteLine("# Entries with # signs are optional and for
user information only.")
objStreamWriter.WriteLine("# User may enter this file with simple a
column of number.")
objStreamWriter.WriteLine("# Example")
objStreamWriter.WriteLine("#" + CStr(.MediaHuella))
objStreamWriter.WriteLine("#" + CStr(.DesvHuella))
objStreamWriter.WriteLine("#" + CStr(.AnchoCarril))
objStreamWriter.WriteLine("...")
objStreamWriter.Close()
'ARCHIVO VEHICLECLASSDISTRIBUTION.TXT
objStreamWriter = New StreamWriter(ruta +
"\_VehicleClassDistribution.txt")
objStreamWriter.WriteLine(CStr(.DistrClase.Porcentajes(0)) + ",Class 4")
objStreamWriter.WriteLine(CStr(.DistrClase.Porcentajes(0)) + ",Class 4")
objStreamWriter.WriteLine(CStr(.DistrClase.Porcentajes(1) +
.DistrClase.Porcentajes(2)) + ",Class 5")
objStreamWriter.WriteLine(CStr(0) + ",Class 6")

```

```

objStreamWriter.WriteLine(CStr(.DistrClase.Porcentajes(4)) + ",Class 7")
objStreamWriter.WriteLine(CStr(.DistrClase.Porcentajes(3)) + ",Class 8")
objStreamWriter.WriteLine(CStr(.DistrClase.Porcentajes(5)) + ",Class 9")
objStreamWriter.WriteLine(CStr(.DistrClase.Porcentajes(6)) + ",Class
10")
objStreamWriter.WriteLine(CStr(0) + ",Class 11")
objStreamWriter.WriteLine(CStr(0) + ",Class 12")
objStreamWriter.WriteLine(CStr(0) + ",Class 13")
objStreamWriter.WriteLine("# Lines with # signs in them are optional and
for user info")
objStreamWriter.WriteLine("#Line 2-11: Percent trucks in Class, Class
#(optional)")
objStreamWriter.WriteLine("#The input class number is optional. Minimum
requirements")
objStreamWriter.Close()
'ARCHIVO HOURLYTRAFFICPERC.TXT
objStreamWriter = New StreamWriter(ruta + "\_HourlyTrafficPerc.txt")
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(0)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(1)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(2)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(3)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(4)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(5)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(6)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(7)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(8)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(9)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(10)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(11)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(12)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(13)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(14)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(15)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(16)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(17)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(18)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(19)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(20)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(21)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(22)))
objStreamWriter.WriteLine(CStr(.DistrHoraria.Porcentajes(23)))
objStreamWriter.Close()
'ARCHIVO MONTHLYADJUSTMENTFACTORS.TXT
objStreamWriter = New StreamWriter(ruta +
"\MonthlyAdjustmentFactor.txt")
objStreamWriter.WriteLine("Month,Class 4,Class 5,Class 6,Class 7,Class
8,Class 9,Class 10,Class 11,Class 12,Class 13")
objStreamWriter.WriteLine("January," +
CStr(.FactorMeses.Factores.Rows(0).Item(1)) + "," +
CStr(.FactorMeses.Factores.Rows(0).Item(2)) + .FactorMeses.Factores.Rows(0).Item(3))
+ "," + CStr(0) + "," + CStr(.FactorMeses.Factores.Rows(0).Item(5)) + "," +
CStr(.FactorMeses.Factores.Rows(0).Item(4)) + "," +
CStr(.FactorMeses.Factores.Rows(0).Item(6)) + "," +
CStr(.FactorMeses.Factores.Rows(0).Item(7)) + "," + CStr(0) + "," + CStr(0) + "," +
CStr(0))

```



```

CStr(.FactorMeses.Factores.Rows(7).Item(7)) + "," + CStr(0) + "," + CStr(0) + "," +
CStr(0))
    objStreamWriter.WriteLine("September," +
CStr(.FactorMeses.Factores.Rows(8).Item(1)) + "," +
CStr(.FactorMeses.Factores.Rows(8).Item(2) + .FactorMeses.Factores.Rows(8).Item(3))
+ "," + CStr(0) + "," + CStr(.FactorMeses.Factores.Rows(8).Item(5)) + "," +
CStr(.FactorMeses.Factores.Rows(8).Item(4)) + "," +
CStr(.FactorMeses.Factores.Rows(8).Item(6)) + "," +
CStr(.FactorMeses.Factores.Rows(8).Item(7)) + "," + CStr(0) + "," + CStr(0) + "," +
CStr(0))
    objStreamWriter.WriteLine("October," +
CStr(.FactorMeses.Factores.Rows(9).Item(1)) + "," +
CStr(.FactorMeses.Factores.Rows(9).Item(2) + .FactorMeses.Factores.Rows(9).Item(3))
+ "," + CStr(0) + "," + CStr(.FactorMeses.Factores.Rows(9).Item(5)) + "," +
CStr(.FactorMeses.Factores.Rows(9).Item(4)) + "," +
CStr(.FactorMeses.Factores.Rows(9).Item(6)) + "," +
CStr(.FactorMeses.Factores.Rows(9).Item(7)) + "," + CStr(0) + "," + CStr(0) + "," +
CStr(0))
    objStreamWriter.WriteLine("November," +
CStr(.FactorMeses.Factores.Rows(10).Item(1)) + "," +
CStr(.FactorMeses.Factores.Rows(10).Item(2) +
.FactorMeses.Factores.Rows(10).Item(3)) + "," + CStr(0) + "," +
CStr(.FactorMeses.Factores.Rows(10).Item(5)) + "," +
CStr(.FactorMeses.Factores.Rows(10).Item(4)) + "," +
CStr(.FactorMeses.Factores.Rows(10).Item(6)) + "," +
CStr(.FactorMeses.Factores.Rows(10).Item(7)) + "," + CStr(0) + "," + CStr(0) + "," +
CStr(0))
    objStreamWriter.WriteLine("December," +
CStr(.FactorMeses.Factores.Rows(11).Item(1)) + "," +
CStr(.FactorMeses.Factores.Rows(11).Item(2) +
.FactorMeses.Factores.Rows(11).Item(3)) + "," + CStr(0) + "," +
CStr(.FactorMeses.Factores.Rows(11).Item(5)) + "," +
CStr(.FactorMeses.Factores.Rows(11).Item(4)) + "," +
CStr(.FactorMeses.Factores.Rows(11).Item(6)) + "," +
CStr(.FactorMeses.Factores.Rows(11).Item(7)) + "," + CStr(0) + "," + CStr(0) + "," +
CStr(0))
    objStreamWriter.Close()
    'ARCHIVO AXLESPERTRUCK.TXT
    objStreamWriter = New StreamWriter(ruta + "\AxlesPerTruck.txt")
    objStreamWriter.WriteLine(" ,Single,Tandem,Tridem,Quad")
    objStreamWriter.WriteLine("Class 4," +
CStr(.NumEjesClase.Rows(0).Item(1)) + "," + CStr(.NumEjesClase.Rows(0).Item(2)) +
",," + CStr(.NumEjesClase.Rows(0).Item(3)) + "," + CStr(0))
    objStreamWriter.WriteLine("Class 5," +
CStr(.NumEjesClase.Rows(1).Item(1) + .NumEjesClase.Rows(2).Item(1)) + "," +
CStr(.NumEjesClase.Rows(1).Item(2) + .NumEjesClase.Rows(2).Item(2)) + "," +
CStr(.NumEjesClase.Rows(1).Item(3) + .NumEjesClase.Rows(2).Item(3)) + "," + CStr(0))
    objStreamWriter.WriteLine("Class 6," + CStr(0) + "," + CStr(0) + "," +
CStr(0) + "," + CStr(0))
    objStreamWriter.WriteLine("Class 7," +
CStr(.NumEjesClase.Rows(4).Item(1)) + "," + CStr(.NumEjesClase.Rows(4).Item(2)) +
",," + CStr(.NumEjesClase.Rows(4).Item(3)) + "," + CStr(0))
    objStreamWriter.WriteLine("Class 8," +
CStr(.NumEjesClase.Rows(3).Item(1)) + "," + CStr(.NumEjesClase.Rows(3).Item(2)) +
",," + CStr(.NumEjesClase.Rows(3).Item(3)) + "," + CStr(0))

```

```

        objStreamWriter.WriteLine("Class 9," +
CStr(.NumEjesClase.Rows(5).Item(1)) + "," + CStr(.NumEjesClase.Rows(5).Item(2)) +
"," + CStr(.NumEjesClase.Rows(5).Item(3)) + "," + CStr(0))
        objStreamWriter.WriteLine("Class 10," +
CStr(.NumEjesClase.Rows(6).Item(1)) + "," + CStr(.NumEjesClase.Rows(6).Item(2)) +
"," + CStr(.NumEjesClase.Rows(6).Item(3)) + "," + CStr(0))
        objStreamWriter.WriteLine("Class 11," + CStr(0) + "," + CStr(0) + "," +
CStr(0) + "," + CStr(0))
        objStreamWriter.WriteLine("Class 12," + CStr(0) + "," + CStr(0) + "," +
CStr(0) + "," + CStr(0))
        objStreamWriter.WriteLine("Class 13," + CStr(0) + "," + CStr(0) + "," +
CStr(0) + "," + CStr(0))
        objStreamWriter.Close()
        'ARCHIVO TRAFFICGROWTH.TXT
        objStreamWriter = New StreamWriter(ruta + "\TrafficGrowth.txt")
        If .FactCrecimiento.TipoCrecimiento = "Ninguno" Then
            objStreamWriter.WriteLine(CStr(0))
            objStreamWriter.WriteLine("No")
            objStreamWriter.WriteLine(CStr(0))
            objStreamWriter.WriteLine("# Lines with # signs in them are optional
and for user information only.")
            objStreamWriter.WriteLine("# Line 1 - (0) Composite vehicle class
growth - (1) Vehicle-class specific growth")
            objStreamWriter.WriteLine("# Line 2 - Input growth rate (one of the
following): No, Linear or Compound")
            objStreamWriter.WriteLine("# Line 3 - Growth Rate - Number (%)
growth rate")
            objStreamWriter.Close()
        Else
            If .FactCrecimiento.UnicoValor = True Then
                objStreamWriter.WriteLine(CStr(0))

                objStreamWriter.WriteLine(CStr(.FactCrecimiento.TipoCrecimiento))
                objStreamWriter.WriteLine(CStr(.FactCrecimiento.FactorGeneral))
                objStreamWriter.WriteLine("# Lines with # signs in them are
optional and for user information only.")
                objStreamWriter.WriteLine("# Line 1 - (0) Composite vehicle
class growth - (1) Vehicle-class specific growth")
                objStreamWriter.WriteLine("# Line 2 - Input growth rate (one of
the following): No, Linear or Compound")
                objStreamWriter.WriteLine("# Line 3 - Growth Rate - Number (%)
growth rate")
                objStreamWriter.Close()
            Else
                objStreamWriter.WriteLine(CStr(1))
                objStreamWriter.WriteLine("4," +
CStr(.FactCrecimiento.FactorPorClase(0)) + "," +
CStr(.FactCrecimiento.TipoCrecimiento))
                objStreamWriter.WriteLine("5," +
CStr((.FactCrecimiento.FactorPorClase(1) + .FactCrecimiento.FactorPorClase(2)) / 2)
+ "," + CStr(.FactCrecimiento.TipoCrecimiento))
                objStreamWriter.WriteLine("6," + CStr(0) + "," +
CStr(.FactCrecimiento.TipoCrecimiento))

```

```

        objStreamWriter.WriteLine("7," +
CStr(.FactCrecimiento.FactorPorClase(4)) + "," +
CStr(.FactCrecimiento.TipoCrecimiento))
        objStreamWriter.WriteLine("8," +
CStr(.FactCrecimiento.FactorPorClase(3)) + "," +
CStr(.FactCrecimiento.TipoCrecimiento))
        objStreamWriter.WriteLine("9," +
CStr(.FactCrecimiento.FactorPorClase(5)) + "," +
CStr(.FactCrecimiento.TipoCrecimiento))
        objStreamWriter.WriteLine("10," +
CStr(.FactCrecimiento.FactorPorClase(6)) + "," +
CStr(.FactCrecimiento.TipoCrecimiento))
        objStreamWriter.WriteLine("11," + CStr(0) + "," +
CStr(.FactCrecimiento.TipoCrecimiento))
        objStreamWriter.WriteLine("12," + CStr(0) + "," +
CStr(.FactCrecimiento.TipoCrecimiento))
        objStreamWriter.WriteLine("13," + CStr(0) + "," +
CStr(.FactCrecimiento.TipoCrecimiento))
        objStreamWriter.WriteLine("# Lines with # signs in them are
optional and for user information only.")
        objStreamWriter.WriteLine("# Line 1 - (0) Composite vehicle
class growth - (1) Vehicle-class specific growth")
        objStreamWriter.WriteLine("# Line 2 - Input growth rate (one of
the following): No, Linear or Compound")
        objStreamWriter.WriteLine("# Line 3 - Growth Rate - Number (%)
growth rate")
        objStreamWriter.Close()
    End If
End If
'ARCHIVO ESPECTROS
'CALCULO DE TPDM, CANTIDAD VEHICULOS, CANTIDAD DE EJES POR TIPO Y
CANTIDAD DE EJES POR CARGA
Dim TPDATreducido As Double
Dim VidaDiseño As Integer = CInt(glInformacionGeneral.VidaDiseño)
Dim PorcentajeClases As Double()
Dim FactoresMeses As Double(,)
Dim NumeroEjesSimplesClase As Double()
Dim NumeroEjesTandemClase As Double()
Dim NumeroEjesTridemClase As Double()
Dim EspectroSimple(,) As Double
Dim EspectroTandem(,) As Double
Dim EspectroTridem(,) As Double
Dim EspectroTandemT(,) As Double
Dim EspectroTridemT(,) As Double
Dim EspectroSimpleT(,) As Double
Dim TipoCrecimiento As String =
glInformacionTrafico.FactCrecimiento.TipoCrecimiento
Dim VectorAños() As Integer
Dim VectorMeses() As String
Dim VectorHoras() As String
ReDim VectorAños(VidaDiseño - 1)
ReDim VectorMeses(11)
ReDim VectorHoras(23)
With glInformacionTrafico
    TPDATreducido = .TPDAT * (.FactorDir / 100) * (.FactorCarril / 100)

```

```

'Distribución por clase
PorcentajeClases = .DistrClase.Porcentajes
'Factores mensuales
FactoresMeses = DataTable2Matriz(.FactorMeses.Factores)
'Ejes simples por clase
NumeroEjesSimplesClase = ExtraerColumnaDataTable(.NumEjesClase, 0)
'Ejes tandem por clase
NumeroEjesTandemClase = ExtraerColumnaDataTable(.NumEjesClase, 1)
'Ejes tridem por clase
NumeroEjesTridemClase = ExtraerColumnaDataTable(.NumEjesClase, 2)
'Espectro eje simple
ReDim EspectroSimple(.EspectroCarga.EspectroSimpleGeneral.Rows.Count
- 1, 6)
    If .EspectroCarga.TipoEspectroSimple = "Por clase" Then
        EspectroSimple =
DataTable2Matriz(.EspectroCarga.EspectroSimplePorClase)
    ElseIf .EspectroCarga.TipoEspectroSimple = "General" Then
        Dim EspectroSimpleGral() As Double
        EspectroSimpleGral =
DataTable2Vector(.EspectroCarga.EspectroSimpleGeneral)
        For i = 0 To EspectroSimpleGral.Length - 1
            For j = 0 To 6
                EspectroSimple(i, j) = EspectroSimpleGral(i)
            Next
        Next
    End If
    EspectroSimpleT = TransponerMatriz(EspectroSimple)
'Espectro eje tandem
ReDim EspectroTandem(.EspectroCarga.EspectroTandemGeneral.Rows.Count
- 1, 6)
    If .EspectroCarga.TipoEspectroTandem = "Por clase" Then
        EspectroTandem =
DataTable2Matriz(.EspectroCarga.EspectroTandemPorClase)
    ElseIf .EspectroCarga.TipoEspectroTandem = "General" Then
        Dim EspectroTandemGral() As Double
        EspectroTandemGral =
DataTable2Vector(.EspectroCarga.EspectroTandemGeneral)
        For i = 0 To EspectroTandemGral.Length - 1
            For j = 0 To 6
                EspectroTandem(i, j) = EspectroTandemGral(i)
            Next
        Next
    End If
    EspectroTandemT = TransponerMatriz(EspectroTandem)
'Espectro eje tridem
ReDim EspectroTridem(.EspectroCarga.EspectroTridemGeneral.Rows.Count
- 1, 6)
    If .EspectroCarga.TipoEspectroTridem = "Por clase" Then
        EspectroTridem =
DataTable2Matriz(.EspectroCarga.EspectroTridemPorClase)
    ElseIf .EspectroCarga.TipoEspectroTridem = "General" Then
        Dim EspectroTridemGral() As Double
        EspectroTridemGral =
DataTable2Vector(.EspectroCarga.EspectroTridemGeneral)
        For i = 0 To EspectroTridemGral.Length - 1

```

```

        For j = 0 To 6
            EspectroTridem(i, j) = EspectroTridemGral(i)
        Next
    Next
End If
EspectroTridemT = TransponerMatriz(EspectroTridem)
End With
'Matrices para el archivo de conexión con ME-PDG
Dim MatrizNumEjeSimCarga(11, 6, UBound(EspectroSimple, 1)) As Double
Dim MatrizNumEjeTanCarga(11, 6, UBound(EspectroTandem, 1)) As Double
Dim MatrizNumEjeTriCarga(11, 6, UBound(EspectroTridem, 1)) As Double
For i = 0 To 11
    For j = 0 To 6
        For k = 0 To UBound(EspectroSimple, 1)
            MatrizNumEjeSimCarga(i, j, k) = TPDATreducido *
PorcentajeClases(j) * FactoresMeses(i, j) * NumeroEjesSimplesClase(j) *
EspectroSimple(k, j)
            MatrizNumEjeSimCarga(i, j, k) =
Math.Round(MatrizNumEjeSimCarga(i, j, k), 2)
        Next
    Next

    For j = 0 To 6
        For k = 0 To UBound(EspectroTandem, 1)
            MatrizNumEjeTanCarga(i, j, k) = TPDATreducido *
PorcentajeClases(j) * FactoresMeses(i, j) * NumeroEjesTandemClase(j) *
EspectroTandem(k, j)
            MatrizNumEjeTanCarga(i, j, k) =
Math.Round(MatrizNumEjeTanCarga(i, j, k), 2)
        Next
    Next

    For j = 0 To 6
        For k = 0 To UBound(EspectroTridem, 1)
            MatrizNumEjeTriCarga(i, j, k) = TPDATreducido *
PorcentajeClases(j) * FactoresMeses(i, j) * NumeroEjesTridemClase(j) *
EspectroTridem(k, j)
            MatrizNumEjeTriCarga(i, j, k) =
Math.Round(MatrizNumEjeTriCarga(i, j, k), 2)
        Next
    Next
Next
Dim cadena As String
objStreamWriter = New StreamWriter(ruta + "\single.alf")
Dim MonthsArray() As String = {"January", "February", "March", "April",
"May", "June", "July", "August", "September", "October", "November", "December"}
For i = 0 To 11
    For j = 0 To 6
        cadena = ""
        For k = 0 To UBound(EspectroSimple, 1) - 1
            cadena = cadena + CStr(MatrizNumEjeSimCarga(i, j, k)) + ","
        Next
        cadena = cadena + CStr(MatrizNumEjeSimCarga(i, j,
UBound(EspectroSimple, 1)))
        objStreamWriter.WriteLine(MonthsArray(i) + "," + cadena)
    Next
Next

```

```

        Next
        objStreamWriter = New StreamWriter(ruta + "\tandem.alf")
        Dim MonthsArray() As String = {"January", "February", "March", "April",
"May", "June", "July", "August", "September", "October", "November", "December"}
        For i = 0 To 11
            For j = 0 To 6
                cadena = ""
                For k = 0 To UBound(EspectroSimple, 1) - 1
                    cadena = cadena + CStr(MatrizNumEjeSimCarga(i, j, k)) + ","
                Next
                cadena = cadena + CStr(MatrizNumEjeTanCarga(i, j,
UBound(EspectroTandem, 1)))
                objStreamWriter.WriteLine(MonthsArray(i) + "," + cadena)
            Next
        Next
        objStreamWriter = New StreamWriter(ruta + "\tridem.alf")
        Dim MonthsArray() As String = {"January", "February", "March", "April",
"May", "June", "July", "August", "September", "October", "November", "December"}
        For i = 0 To 11
            For j = 0 To 6
                cadena = ""
                For k = 0 To UBound(EspectroSimple, 1) - 1
                    cadena = cadena + CStr(MatrizNumEjeTriCarga(i, j, k)) + ","
                Next
                cadena = cadena + CStr(MatrizNumEjeTriCarga(i, j,
UBound(EspectroTridem, 1)))
                objStreamWriter.WriteLine(MonthsArray(i) + "," + cadena)
            Next
        Next
    End With
End With
salida:
End Sub

```

Factor direccional

```

Public Class AyudaFactorDireccional
    Private Sub btnVolver_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnVolver.Click
        Me.Close()
    End Sub
End Class

```

Factor carril

```

Public Class AyudaFactorCarril
    Private Sub btnVolver_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnVolver.Click
        Me.Close()
    End Sub
End Class

```

Velocidad de operación

```

Public Class AyudaVelocidadOperacion

```

```

    Private Sub btnVolver_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnVolver.Click
        Me.Close()
    End Sub
End Class

```

Cálculo de TPDAC

```

Public Class CalcularTPDAC
    Private Sub btnCalcular_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCalcular.Click
        Dim TPDA As Double
        Dim PorcentajeBuses As Double
        Dim porcentajeCamiones As Double
        Dim TPDAC As Double
        Try
            TPDA = CDb1(txtTPDA.Text)
            PorcentajeBuses = CDb1(txtPorcentajeBuses.Text)
            porcentajeCamiones = CDb1(txtPorcentajeCamiones.Text)
            TPDAC = TPDA * ((PorcentajeBuses + porcentajeCamiones) / 100)
            txtTPDACcalc.Text = CStr(TPDAC)
        Catch ex As Exception
            MsgBox("Los valores ingresados para el cálculo no son numéricos o hay
campos vacíos. Verifique e intente nuevamente", vbInformation, "Valores
incorrectos")
        End Try
    End Sub
    Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
        Transito.txtTPDinicial.Text = txtTPDACcalc.Text
        Me.Close()
    End Sub
    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

```

AQUI VA LO OTRO

```

'ARCHIVO ESPECTROS
'CALCULO DE TPDM, CANTIDAD VEHICULOS, CANTIDAD DE EJES POR TIPO Y
Dim TPDATreducido As Double
    Dim VidaDiseño As Integer = CInt(glInformacionGeneral.VidaDiseño)
    Dim PorcentajeClases As Double()
    Dim FactoresMeses As Double(,)
    Dim NumeroEjesSimplesClase As Double()
    Dim NumeroEjesTandemClase As Double()
    Dim NumeroEjesTridemClase As Double()
    Dim EspectroSimple(,) As Double
    Dim EspectroTandem(,) As Double
    Dim EspectroTridem(,) As Double
    Dim EspectroTandemT(,) As Double
    Dim EspectroTridemT(,) As Double
    Dim EspectroSimpleT(,) As Double
    Dim TipoCrecimiento As String =
glInformacionTrafico.FactCrecimiento.TipoCrecimiento
    Dim VectorAños() As Integer

```

```

Dim VectorMeses() As String
Dim VectorHoras() As String
ReDim VectorAños(VidaDiseño - 1)
ReDim VectorMeses(11)
ReDim VectorHoras(23)
With glInformacionTrafico

    TPDATreducido = .TPDAT * (.FactorDir / 100) * (.FactorCarril / 100)
    'Distribución por clase
    PorcentajeClases = .DistrClase.Porcentajes
    'Factores mensuales
    FactoresMeses = DataTable2Matriz(.FactorMeses.Factores)
    'Ejes simples por clase
    NumeroEjesSimplesClase = ExtraerColumnaDataTable(.NumEjesClase, 0)
    'Ejes tandem por clase
    NumeroEjesTandemClase = ExtraerColumnaDataTable(.NumEjesClase, 1)
    'Ejes tridem por clase
    NumeroEjesTridemClase = ExtraerColumnaDataTable(.NumEjesClase, 2)
    'Espectro eje simple
    ReDim EspectroSimple(.EspectroCarga.EspectroSimpleGeneral.Rows.Count
- 1, 6)
    If .EspectroCarga.TipoEspectroSimple = "Por clase" Then
        EspectroSimple =
DataTable2Matriz(.EspectroCarga.EspectroSimplePorClase)
    ElseIf .EspectroCarga.TipoEspectroSimple = "General" Then
        Dim EspectroSimpleGral() As Double
        EspectroSimpleGral =
DataTable2Vector(.EspectroCarga.EspectroSimpleGeneral)
        For i = 0 To EspectroSimpleGral.Length - 1
            For j = 0 To 6
                EspectroSimple(i, j) = EspectroSimpleGral(i)
            Next
        Next
    End If
    EspectroSimpleT = TransponerMatriz(EspectroSimple)
    'Espectro eje tandem
    ReDim EspectroTandem(.EspectroCarga.EspectroTandemGeneral.Rows.Count
- 1, 6)
    If .EspectroCarga.TipoEspectroTandem = "Por clase" Then
        EspectroTandem =
DataTable2Matriz(.EspectroCarga.EspectroTandemPorClase)
    ElseIf .EspectroCarga.TipoEspectroTandem = "General" Then
        Dim EspectroTandemGral() As Double
        EspectroTandemGral =
DataTable2Vector(.EspectroCarga.EspectroTandemGeneral)
        For i = 0 To EspectroTandemGral.Length - 1
            For j = 0 To 6
                EspectroTandem(i, j) = EspectroTandemGral(i)
            Next
        Next
    End If
    EspectroTandemT = TransponerMatriz(EspectroTandem)
    'Espectro eje tridem
    ReDim EspectroTridem(.EspectroCarga.EspectroTridemGeneral.Rows.Count
- 1, 6)

```

```

        If .EspectroCarga.TipoEspectroTridem = "Por clase" Then
            EspectroTridem =
DataTable2Matriz(.EspectroCarga.EspectroTridemPorClase)
        ElseIf .EspectroCarga.TipoEspectroTridem = "General" Then
            Dim EspectroTridemGral() As Double
            EspectroTridemGral =
DataTable2Vector(.EspectroCarga.EspectroTridemGeneral)
            For i = 0 To EspectroTridemGral.Length - 1
                For j = 0 To 6
                    EspectroTridem(i, j) = EspectroTridemGral(i)
                Next
            Next
        End If
        EspectroTridemT = TransponerMatriz(EspectroTridem)
    End With
    'Matrices para el archivo de conexión con ME-PDG
    Dim MatrizNumEjeSimCarga(11, 6, UBound(EspectroSimple, 1)) As Double
    Dim MatrizNumEjeTanCarga(11, 6, UBound(EspectroTandem, 1)) As Double
    Dim MatrizNumEjeTriCarga(11, 6, UBound(EspectroTridem, 1)) As Double

    For i = 0 To 11
        For j = 0 To 6
            For k = 0 To UBound(EspectroSimple, 1)
                MatrizNumEjeSimCarga(i, j, k) = TPDATreducido *
PorcentajeClases(j) * FactoresMeses(i, j) * NumeroEjesSimplesClase(j) *
EspectroSimple(k, j)
                MatrizNumEjeSimCarga(i, j, k) =
Math.Round(MatrizNumEjeSimCarga(i, j, k), 2)
            Next
        Next
        For j = 0 To 6
            For k = 0 To UBound(EspectroTandem, 1)
                MatrizNumEjeTanCarga(i, j, k) = TPDATreducido *
PorcentajeClases(j) * FactoresMeses(i, j) * NumeroEjesTandemClase(j) *
EspectroTandem(k, j)
                MatrizNumEjeTanCarga(i, j, k) =
Math.Round(MatrizNumEjeTanCarga(i, j, k), 2)
            Next
        Next
        For j = 0 To 6
            For k = 0 To UBound(EspectroTridem, 1)
                MatrizNumEjeTriCarga(i, j, k) = TPDATreducido *
PorcentajeClases(j) * FactoresMeses(i, j) * NumeroEjesTridemClase(j) *
EspectroTridem(k, j)
                MatrizNumEjeTriCarga(i, j, k) =
Math.Round(MatrizNumEjeTriCarga(i, j, k), 2)
            Next
        Next
        Next
        Dim cadena As String
        objStreamWriter = New StreamWriter(ruta + "\single.alf")
        Dim MonthsArray() As String = {"January", "February", "March", "April",
"May", "June", "July", "August", "September", "October", "November", "December"}
        For i = 0 To 11
            For j = 0 To 6

```

```

        cadena = ""
        For k = 0 To UBound(EspectroSimple, 1) - 1
            cadena = cadena + CStr(MatrizNumEjeSimCarga(i, j, k)) + ","
        Next
        cadena = cadena + CStr(MatrizNumEjeSimCarga(i, j,
UBound(EspectroSimple, 1)))
        objStreamWriter.WriteLine(MonthsArray(i) + "," + cadena)
    Next
Next
objStreamWriter = New StreamWriter(ruta + "\tandem.alf")
Dim MonthsArray() As String = {"January", "February", "March", "April",
"May", "June", "July", "August", "September", "October", "November", "December"}
For i = 0 To 11
    For j = 0 To 6
        cadena = ""
        For k = 0 To UBound(EspectroSimple, 1) - 1
            cadena = cadena + CStr(MatrizNumEjeSimCarga(i, j, k)) + ","
        Next
        cadena = cadena + CStr(MatrizNumEjeTanCarga(i, j,
UBound(EspectroTandem, 1)))
        objStreamWriter.WriteLine(MonthsArray(i) + "," + cadena)
    Next
Next
objStreamWriter = New StreamWriter(ruta + "\tridem.alf")
Dim MonthsArray() As String = {"January", "February", "March", "April",
"May", "June", "July", "August", "September", "October", "November", "December"}
For i = 0 To 11
    For j = 0 To 6
        cadena = ""
        For k = 0 To UBound(EspectroSimple, 1) - 1
            cadena = cadena + CStr(MatrizNumEjeTriCarga(i, j, k)) + ","
        Next
        cadena = cadena + CStr(MatrizNumEjeTriCarga(i, j,
UBound(EspectroTridem, 1)))
        objStreamWriter.WriteLine(MonthsArray(i) + "," + cadena)
    Next
Next
End With
End Class

```

Entradas generales

```

Public Class EntGr1s
    Private Sub EntGr1s_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        With glInformacionTrafico
            txtDistHuella.Text = CStr(.MediaHuella)
            txtDesvEst.Text = CStr(.DesvHuella)
            txtAnchoCarril.Text = CStr(.AnchoCarril)
            txtDistPromCorto.Text = CStr(.Batalla.MediaEspaciamiento(0))
            txtDistPromMedio.Text = CStr(.Batalla.MediaEspaciamiento(1))
            txtDistPromLargo.Text = CStr(.Batalla.MediaEspaciamiento(2))
            txtPorcentajeCorto.Text = CStr(.Batalla.PorcentajeCamiones(0))
            txtPorcentajeMedio.Text = CStr(.Batalla.PorcentajeCamiones(1))
            txtPorcentajeLargo.Text = CStr(.Batalla.PorcentajeCamiones(2))
            txtAnchoPromEje.Text = CStr(.ConfiguracionEje.AnchoEje)
        End With
    End Sub
End Class

```

```

txtDistRuedaDoble.Text = CStr(.ConfiguracionEje.DistGemelado)
txtPresionInflado.Text = CStr(.ConfiguracionEje.PresionInflado)
txtEspTandem.Text = CStr(.ConfiguracionEje.EspaciamientoTandem)
txtEspTridem.Text = CStr(.ConfiguracionEje.EspaciamientoTridem)
dgvEjesClase.DataSource = glInformacionTrafico.NumEjesClase
End With
With dgvEjesClase
.Columns("CLASE").Width = 80
.Columns("SIMPLE").Width = 60
.Columns("TÁNDEM").Width = 60
.Columns("TRÍDEM").Width = 60
For i = 0 To .Columns.Count - 1
.Columns.Item(i).SortMode = DataGridViewColumnSortMode.NotSortable
Next i
Dim fuenteTitulos As New Font(.DefaultCellStyle.Font.FontFamily, 7,
FontStyle.Bold)
.ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
.ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
.ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
.Columns(0).DefaultCellStyle = .ColumnHeadersDefaultCellStyle
.Columns(0).ReadOnly = True
.RowHeadersVisible = False
Dim row As DataGridViewRow
For Each row In .Rows
row.Height = 20
Next
.EditMode = DataGridViewEditMode.EditOnEnter
.DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
.AllowUserToResizeColumns = False
.AllowUserToResizeRows = False
.AllowUserToAddRows = False
.AllowUserToDeleteRows = False
.AllowUserToOrderColumns = False
.MultiSelect = False
End With
End Sub
Private Sub btnDistHuella_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnDistHuella.Click
AyudaHuella.Show()
End Sub
Private Sub btnAnchoCarril_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAnchoCarril.Click
AyudaAnchoCalzada.Show()
End Sub
'BOTONES ACEPTAR/CANCELAR
Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
Me.Close()
End Sub
Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
Dim sumaPorcentajesCamiones As Double = CDb1(txtPorcentajeCorto.Text) +
CDbl(txtPorcentajeMedio.Text) + CDb1(txtPorcentajeLargo.Text)
sumaPorcentajesCamiones = Math.Round(sumaPorcentajesCamiones, 2)

```

```

If sumaPorcentajesCamiones >= 99.999 And sumaPorcentajesCamiones <= 100.001
Then
    With glInformacionTrafico
        .MediaHuellla = CDb1(txtDistHuellla.Text)
        .DesvHuellla = CDb1(txtDesvEst.Text)
        .AnchoCarril = CDb1(txtAnchoCarril.Text)
        .NumEjesClase = dgvEjesClase.DataSource
        .ConfiguracionEje = glConfigEje
        .ConfiguracionEje.AnchoEje = CDb1(txtAnchoPromEje.Text)
        .ConfiguracionEje.DistGemelado = CDb1(txtDistRuedaDoble.Text)
        .ConfiguracionEje.PresionInflado = CDb1(txtPresionInflado.Text)
        .ConfiguracionEje.EspaciamientoTandem = CDb1(txtEspTandem.Text)
        .ConfiguracionEje.EspaciamientoTridem = CDb1(txtEspTridem.Text)
        .Batalla = glBatalla
        .Batalla.MediaEspaciamiento = {CDb1(txtDistPromCorto.Text),
CDb1(txtDistPromMedio.Text), CDb1(txtDistPromLargo.Text)}
        .Batalla.PorcentajeCamiones = {CDb1(txtPorcentajeCorto.Text),
CDb1(txtPorcentajeMedio.Text), CDb1(txtPorcentajeLargo.Text)}
    End With
    Me.Close()
Else
    MsgBox("La suma de los porcentajes de camiones de la pestaña Batalla
debe ser igual a 100%, verifique los valores antes de continuar.", vbInformation,
"Total de porcentajes")
    End If
End Sub
End Class

```

Huella

```

Public Class AyudaHuellla
    Private Sub btnVolver_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnVolver.Click
        Me.Close()
    End Sub
End Class

```

Calzada

```

Public Class AyudaAnchoCalzada
    Private Sub btnVolver_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnVolver.Click
        Me.Close()
    End Sub
End Class

```

Volúmenes de carga

```

Imports System.IO
Imports System.Xml.Serialization
Public Class VolCarga
    Public DistribucionClase As New ClaseDistrClase
    Public DistribucionHoraria As New ClaseDistrHoraria
    Public FactoresMensuales As New ClaseFactMensuales
    Public srlzDistrClase As New XmlSerializer(DistribucionClase.GetType)
    Public srlzDistrHoraria As New XmlSerializer(DistribucionHoraria.GetType)

```

```

Public srlzFacMensuales As New XmlSerializer(FactoresMensuales.GetType)
Public SumaDistrClase As New Double
Public sumaDistrHoraria As New Double
Public strmDistrClase As Stream
Public strmDistrHoraria As Stream
Public strmFacMensuales As Stream

'Inicio
Private Sub VolumenCarga_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    With glInformacionTrafico
        txtBus.Text = CStr(.DistrClase.Porcentajes(0))
        txtC2P.Text = CStr(.DistrClase.Porcentajes(1))
        txtC2G.Text = CStr(.DistrClase.Porcentajes(2))
        txtC3.Text = CStr(.DistrClase.Porcentajes(3))
        txtC4.Text = CStr(.DistrClase.Porcentajes(4))
        txtC5.Text = CStr(.DistrClase.Porcentajes(5))
        txtMayorC5.Text = CStr(.DistrClase.Porcentajes(6))
        txt12am.Text = CStr(.DistrHoraria.Porcentajes(0))
        txt1am.Text = CStr(.DistrHoraria.Porcentajes(1))
        txt2am.Text = CStr(.DistrHoraria.Porcentajes(2))
        txt3am.Text = CStr(.DistrHoraria.Porcentajes(3))
        txt4am.Text = CStr(.DistrHoraria.Porcentajes(4))
        txt5am.Text = CStr(.DistrHoraria.Porcentajes(5))
        txt6am.Text = CStr(.DistrHoraria.Porcentajes(6))
        txt7am.Text = CStr(.DistrHoraria.Porcentajes(7))
        txt8am.Text = CStr(.DistrHoraria.Porcentajes(8))
        txt9am.Text = CStr(.DistrHoraria.Porcentajes(9))
        txt10am.Text = CStr(.DistrHoraria.Porcentajes(10))
        txt11am.Text = CStr(.DistrHoraria.Porcentajes(11))
        txt12m.Text = CStr(.DistrHoraria.Porcentajes(12))
        txt1pm.Text = CStr(.DistrHoraria.Porcentajes(13))
        txt2pm.Text = CStr(.DistrHoraria.Porcentajes(14))
        txt3pm.Text = CStr(.DistrHoraria.Porcentajes(15))
        txt4pm.Text = CStr(.DistrHoraria.Porcentajes(16))
        txt5pm.Text = CStr(.DistrHoraria.Porcentajes(17))
        txt6pm.Text = CStr(.DistrHoraria.Porcentajes(18))
        txt7pm.Text = CStr(.DistrHoraria.Porcentajes(19))
        txt8pm.Text = CStr(.DistrHoraria.Porcentajes(20))
        txt9pm.Text = CStr(.DistrHoraria.Porcentajes(21))
        txt10pm.Text = CStr(.DistrHoraria.Porcentajes(22))
        txt11pm.Text = CStr(.DistrHoraria.Porcentajes(23))
        dgvDistrMes.DataSource = .FactorMeses.Factores
    End With
    With dgvDistrMes
        .Columns("MES").Width = 80
        .Columns("BUS").Width = 35
        .Columns("C2P").Width = 35
        .Columns("C2G").Width = 35
        .Columns("C3").Width = 35
        .Columns("C4").Width = 35
        .Columns("C5").Width = 35
        .Columns(">C5").Width = 35
    End With
    Dim row As DataGridViewRow

```

```

        For Each row In .Rows
            row.Height = 20
        Next
        For i = 0 To .Columns.Count - 1
            .Columns.Item(i).SortMode = DataGridViewColumnSortMode.NotSortable
        Next i
        Dim fuenteTitulos As New Font(.DefaultCellStyle.Font.FontFamily, 7,
FontStyle.Bold)
        .ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
        .ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
        .ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
        .Columns(0).DefaultCellStyle = .ColumnHeadersDefaultCellStyle
        .RowHeadersVisible = False
        .Rows(0).ReadOnly = True
        .EditMode = DataGridViewEditMode.EditOnEnter
        .DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
        .AllowUserToResizeColumns = False
        .AllowUserToResizeRows = False
        .AllowUserToAddRows = False
        .AllowUserToDeleteRows = False
        .AllowUserToOrderColumns = False
        .MultiSelect = False
    End With
End Sub
'Guardar/cargar distribución por clase
Private Sub btnGuardarDistrClase_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnGuardarDistrClase.Click
    If SumaDistrClase >= 99.999 And SumaDistrClase <= 100.001 Then
        'EXTRAER INFORMACIÓN A CONTROLES
        With DistribucionClase
            .Porcentajes = {Cdbl(txtBus.Text), Cdbl(txtC2P.Text),
Cdbl(txtC2G.Text), Cdbl(txtC3.Text), Cdbl(txtC4.Text), Cdbl(txtC5.Text),
Cdbl(txtMayorC5.Text)}
        End With
        If saveDistrClase.ShowDialog() = DialogResult.OK Then
            strmDistrClase = saveDistrClase.OpenFile()
            Using strmW As StreamWriter = New StreamWriter(strmDistrClase)
                srlzDistrClase.Serialize(strmW, DistribucionClase)
                strmW.Close()
            End Using
            If (strmDistrClase IsNot Nothing) Then
                strmDistrClase.Close()
            End If
        End If
    Else
        MsgBox("La suma de los porcentajes de las clases de vehículo debe ser
igual a 100%, verifique los valores antes de continuar.", vbInformation, "Total de
porcentajes")
    End If
End Sub
Private Sub btnCargarDistrClase_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles btnCargarDistrClase.Click
    If openDistrClase.ShowDialog() = System.Windows.Forms.DialogResult.OK Then
        Try

```

```

strmDistrClase = openDistrClase.OpenFile()
Dim strmR As New StreamReader(strmDistrClase)
DistribucionClase = srlzDistrClase.Deserialize(strmR)
strmR.Close()
'ASIGNAR INFORMACIÓN A CONTROLES
With DistribucionClase
    txtBus.Text = CStr(DistribucionClase.Porcentajes(0))
    txtC2P.Text = CStr(DistribucionClase.Porcentajes(1))
    txtC2G.Text = CStr(DistribucionClase.Porcentajes(2))
    txtC3.Text = CStr(DistribucionClase.Porcentajes(3))
    txtC4.Text = CStr(DistribucionClase.Porcentajes(4))
    txtC5.Text = CStr(DistribucionClase.Porcentajes(5))
    txtMayorC5.Text = CStr(DistribucionClase.Porcentajes(6))
End With
If (strmDistrClase IsNot Nothing) Then
    strmDistrClase.Close()
End If
Catch Ex As Exception
    MessageBox.Show("No se puede leer el archivo. Causa: " & Ex.Message)
Finally
    If (strmDistrClase IsNot Nothing) Then
        strmDistrClase.Close()
    End If
End Try
End If
End Sub
'Guardar/cargar distribución horaria
Private Sub btnGuardarDistrHoraria_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnGuardarDistrHoraria.Click
    'EXTRAER INFORMACIÓN A CONTROLES
    If sumaDistrHoraria >= 99.999 And sumaDistrHoraria < 100.001 Then
        With DistribucionHoraria
            .Porcentajes = {Cdbl(txt12am.Text), Cdbl(txt1am.Text),
Cdbl(txt2am.Text), Cdbl(txt3am.Text), Cdbl(txt4am.Text), Cdbl(txt5am.Text),
Cdbl(txt6am.Text), Cdbl(txt7am.Text), Cdbl(txt8am.Text), Cdbl(txt9am.Text),
Cdbl(txt10am.Text), Cdbl(txt11am.Text), Cdbl(txt12m.Text), Cdbl(txt1pm.Text),
Cdbl(txt2pm.Text), Cdbl(txt3pm.Text), Cdbl(txt4pm.Text), Cdbl(txt5pm.Text),
Cdbl(txt6pm.Text), Cdbl(txt7pm.Text), Cdbl(txt8pm.Text), Cdbl(txt9pm.Text),
Cdbl(txt10pm.Text), Cdbl(txt11pm.Text)}
        End With
        If saveDistrHoraria.ShowDialog() = DialogResult.OK Then
            strmDistrHoraria = saveDistrHoraria.OpenFile()
            Using strmW As StreamWriter = New StreamWriter(strmDistrHoraria)
                srlzDistrHoraria.Serialize(strmW, DistribucionHoraria)
                strmW.Close()
            End Using
            If (strmDistrHoraria IsNot Nothing) Then
                strmDistrHoraria.Close()
            End If
        End If
    Else
        MsgBox("La suma de los porcentajes de todas las horas del día debe ser
igual a 100%, verifique los valores antes de continuar.", vbInformation, "Total de
porcentajes")
    End If
End If

```

```

End Sub
Private Sub btnCargarDistrHoraria_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles btnCargarDistrHoraria.Click
    If openDistrHoraria.ShowDialog() = System.Windows.Forms.DialogResult.OK Then
        Try
            strmDistrHoraria = openDistrHoraria.OpenFile()
            Dim strmR As New StreamReader(strmDistrHoraria)

            FactoresMensuales = srlzDistrHoraria.Deserialize(strmR)
            strmR.Close()
            'ASIGNAR INFORMACIÓN A CONTROLES
            With DistribucionHoraria
                txt12am.Text = CStr(.Porcentajes(0))
                txt1am.Text = CStr(.Porcentajes(1))
                txt2am.Text = CStr(.Porcentajes(2))
                txt3am.Text = CStr(.Porcentajes(3))
                txt4am.Text = CStr(.Porcentajes(4))
                txt5am.Text = CStr(.Porcentajes(5))
                txt6am.Text = CStr(.Porcentajes(6))
                txt7am.Text = CStr(.Porcentajes(7))
                txt8am.Text = CStr(.Porcentajes(8))
                txt9am.Text = CStr(.Porcentajes(9))
                txt10am.Text = CStr(.Porcentajes(10))
                txt11am.Text = CStr(.Porcentajes(11))
                txt12m.Text = CStr(.Porcentajes(12))
                txt1pm.Text = CStr(.Porcentajes(13))
                txt2pm.Text = CStr(.Porcentajes(14))
                txt3pm.Text = CStr(.Porcentajes(15))
                txt4pm.Text = CStr(.Porcentajes(16))
                txt5pm.Text = CStr(.Porcentajes(17))
                txt6pm.Text = CStr(.Porcentajes(18))
                txt7pm.Text = CStr(.Porcentajes(19))
                txt8pm.Text = CStr(.Porcentajes(20))
                txt9pm.Text = CStr(.Porcentajes(21))
                txt10pm.Text = CStr(.Porcentajes(22))
                txt11pm.Text = CStr(.Porcentajes(23))
            End With
            If (strmDistrHoraria IsNot Nothing) Then
                strmDistrHoraria.Close()
            End If
        Catch Ex As Exception
            MessageBox.Show("No se puede leer el archivo. Causa: " & Ex.Message)
        Finally
            If (strmDistrHoraria IsNot Nothing) Then
                strmDistrHoraria.Close()
            End If
        End Try
    End If
End Sub
'Guardar/cargar factores mensuales
Private Sub btnGuardarFacMensuales_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnGuardarFactMensuales.Click
    'EXTRAER INFORMACIÓN A CONTROLES
    With FactoresMensuales
        .Factores = dgvDistrMes.DataSource
    End With
End Sub

```

```

End With
If saveFacMensuales.ShowDialog() = DialogResult.OK Then
    strmFacMensuales = saveFacMensuales.OpenFile()
    Using strmW As StreamWriter = New StreamWriter(strmFacMensuales)
        srlzFacMensuales.Serialize(strmW, FactoresMensuales)
        strmW.Close()
    End Using
    If (strmFacMensuales IsNot Nothing) Then
        strmFacMensuales.Close()
    End If
End If
End Sub
Private Sub btnCargarFacMensuales_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles btnCargarFactMensuales.Click
    If openFacMensuales.ShowDialog() = System.Windows.Forms.DialogResult.OK Then
        Try
            strmFacMensuales = openFacMensuales.OpenFile()
            Dim strmR As New StreamReader(strmFacMensuales)
            FactoresMensuales = srlzFacMensuales.Deserialize(strmR)
            strmR.Close()
            'ASIGNAR INFORMACIÓN A CONTROLES
            With FactoresMensuales
                dgvDistrMes.DataSource = FactoresMensuales.Factores
            End With
            If (strmFacMensuales IsNot Nothing) Then
                strmFacMensuales.Close()
            End If
        Catch ex As Exception
            MessageBox.Show("No se puede leer el archivo. Causa: " & ex.Message)
        Finally
            If (strmFacMensuales IsNot Nothing) Then
                strmFacMensuales.Close()
            End If
        End Try
    End If
End Sub
'Verificación de porcentajes
Private Sub txtDistribucionClase_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtBus.TextChanged, txtC2G.TextChanged,
txtC2P.TextChanged, txtC3.TextChanged, txtC4.TextChanged, txtC5.TextChanged,
txtMayorC5.TextChanged
    Try
        SumaDistrClase = CDb1(txtBus.Text) + CDb1(txtC2G.Text) +
CDb1(txtC2P.Text) + CDb1(txtC3.Text) + CDb1(txtC4.Text) + CDb1(txtC5.Text) +
CDb1(txtMayorC5.Text)
        SumaDistrClase = Math.Round(SumaDistrClase, 2)
        lblTotalClases.Text = CStr(SumaDistrClase)
        If SumaDistrClase > 99.999 And SumaDistrClase < 100.001 Then
            lblTotalClases.ForeColor = Color.Black
        Else
            lblTotalClases.ForeColor = Color.DarkRed
        End If
    Catch ex As Exception
    End Try
End Sub

```

```

Private Sub txtDistribucionHoraria_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txt12am.TextChanged, txt1am.TextChanged,
txt2am.TextChanged, txt3am.TextChanged, txt4am.TextChanged, txt5am.TextChanged,
txt6am.TextChanged, txt7am.TextChanged, txt8am.TextChanged, txt9am.TextChanged,
txt10am.TextChanged, txt11am.TextChanged, txt12m.TextChanged, txt1pm.TextChanged,
txt2pm.TextChanged, txt3pm.TextChanged, txt4pm.TextChanged, txt5pm.TextChanged,
txt6pm.TextChanged, txt7pm.TextChanged, txt8pm.TextChanged, txt9pm.TextChanged,
txt10pm.TextChanged, txt11pm.TextChanged
Try
    sumaDistrHoraria = CDb1(txt12am.Text) + CDb1(txt1am.Text) +
CDbl(txt2am.Text) + CDb1(txt3am.Text) + CDb1(txt4am.Text) + CDb1(txt5am.Text) +
CDbl(txt6am.Text) + CDb1(txt7am.Text) + CDb1(txt8am.Text) + CDb1(txt9am.Text) +
CDbl(txt10am.Text) + CDb1(txt11am.Text) + CDb1(txt12m.Text) + CDb1(txt1pm.Text) +
CDbl(txt2pm.Text) + CDb1(txt3pm.Text) + CDb1(txt4pm.Text) + CDb1(txt5pm.Text) +
CDbl(txt6pm.Text) + CDb1(txt7pm.Text) + CDb1(txt8pm.Text) + CDb1(txt9pm.Text) +
CDbl(txt10pm.Text) + CDb1(txt11pm.Text)
    sumaDistrHoraria = Math.Round(sumaDistrHoraria, 2)
    lblTotalHoraria.Text = CStr(sumaDistrHoraria)
    If sumaDistrHoraria > 99.999 And sumaDistrHoraria < 100.001 Then
        lblTotalHoraria.ForeColor = Color.Black
    Else
        lblTotalHoraria.ForeColor = Color.DarkRed
    End If
Catch ex As Exception
End Try
End Sub
'Aceptar/Cancelar
Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub
Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    If SumaDistrClase <= 99.999 Or SumaDistrClase >= 100.001 Or sumaDistrHoraria
<= 99.999 Or sumaDistrHoraria >= 100.001 Then
        MsgBox("El valor de los porcentajes no es correcto. Verifique los
valores porcentuales de las distribuciones por clase y horarias.", vbInformation,
"Porcentajes")
    Else
        With glInformacionTrafico
            .DistrClase = glDistrClase
            .DistrClase.Porcentajes = {CDbl(txtBus.Text), CDb1(txtC2P.Text),
CDbl(txtC2G.Text), CDb1(txtC3.Text), CDb1(txtC4.Text), CDb1(txtC5.Text),
CDbl(txtMayorC5.Text)}
            .DistrHoraria = glDistrHoraria
            .DistrHoraria.Porcentajes = {CDbl(txt12am.Text), CDb1(txt1am.Text),
CDbl(txt2am.Text), CDb1(txt3am.Text), CDb1(txt4am.Text), CDb1(txt5am.Text),
CDbl(txt6am.Text), CDb1(txt7am.Text), CDb1(txt8am.Text), CDb1(txt9am.Text),
CDbl(txt10am.Text), CDb1(txt11am.Text), CDb1(txt12m.Text), CDb1(txt1pm.Text),
CDbl(txt2pm.Text), CDb1(txt3pm.Text), CDb1(txt4pm.Text), CDb1(txt5pm.Text),
CDbl(txt6pm.Text), CDb1(txt7pm.Text), CDb1(txt8pm.Text), CDb1(txt9pm.Text),
CDbl(txt10pm.Text), CDb1(txt11pm.Text)}
            .FactorMeses = glFacMensuales
            .FactorMeses.Factores = dgvDistrMes.DataSource
        End With
    End If
End Sub

```

```

        Me.Close()
    End If
End Sub
End Class

```

Espectro de cargas

```

Imports System.IO
Imports System.Xml.Serialization
Public Class DistrCarga
    Public EspectrosCarga As New ClaseEspectroCarga
    Public srlzEspectrosCarga As New XmlSerializer(EspectrosCarga.GetType)
    Public sumaSimpleGeneral As Double = 0
    Public sumaSimpleClaseBUS As Double = 0
    Public sumaSimpleClaseC2P As Double = 0
    Public sumaSimpleClaseC2G As Double = 0
    Public sumaSimpleClaseC3 As Double = 0
    Public sumaSimpleClaseC4 As Double = 0
    Public sumaSimpleClaseC5 As Double = 0
    Public sumaSimpleClaseMayorC5 As Double = 0
    Public sumaTandemGeneral As Double = 0
    Public sumaTandemClaseBUS As Double = 0
    Public sumaTandemClaseC2P As Double = 0
    Public sumaTandemClaseC2G As Double = 0
    Public sumaTandemClaseC3 As Double = 0
    Public sumaTandemClaseC4 As Double = 0
    Public sumaTandemClaseC5 As Double = 0
    Public sumaTandemClaseMayorC5 As Double = 0
    Public sumaTridemGeneral As Double = 0
    Public sumaTridemClaseBUS As Double = 0
    Public sumaTridemClaseC2P As Double = 0
    Public sumaTridemClaseC2G As Double = 0
    Public sumaTridemClaseC3 As Double = 0
    Public sumaTridemClaseC4 As Double = 0
    Public sumaTridemClaseC5 As Double = 0
    Public sumaTridemClaseMayorC5 As Double = 0
    Public strmEspectrosCarga As Stream
    Sub SumarPorcentajes()

        sumaSimpleGeneral = 0
        sumaSimpleClaseBUS = 0
        sumaSimpleClaseC2P = 0
        sumaSimpleClaseC2G = 0
        sumaSimpleClaseC3 = 0
        sumaSimpleClaseC4 = 0
        sumaSimpleClaseC5 = 0
        sumaSimpleClaseMayorC5 = 0
        sumaTandemGeneral = 0
        sumaTandemClaseBUS = 0
        sumaTandemClaseC2P = 0
        sumaTandemClaseC2G = 0
        sumaTandemClaseC3 = 0
        sumaTandemClaseC4 = 0
        sumaTandemClaseC5 = 0
        sumaTandemClaseMayorC5 = 0
        sumaTridemGeneral = 0
    End Sub
End Class

```

```

sumaTridemClaseBUS = 0
sumaTridemClaseC2P = 0
sumaTridemClaseC2G = 0
sumaTridemClaseC3 = 0
sumaTridemClaseC4 = 0
sumaTridemClaseC5 = 0
sumaTridemClaseMayorC5 = 0

For Each fila As DataGridViewRow In dgvGeneralSimple.Rows
    If fila.Cells("General").Value Is Nothing Then
        fila.Cells("General").Value = 0
    End If
    sumaSimpleGeneral += CDb1(fila.Cells("General").Value)
Next

For Each fila As DataGridViewRow In dgvClaseSimple.Rows
    If fila.Cells("BUS").Value Is Nothing Then
        fila.Cells("BUS").Value = 0
    ElseIf fila.Cells("C2P").Value Is Nothing Then
        fila.Cells("C2P").Value = 0
    ElseIf fila.Cells("C2G").Value Is Nothing Then
        fila.Cells("C2G").Value = 0
    ElseIf fila.Cells("C3").Value Is Nothing Then
        fila.Cells("C3").Value = 0
    ElseIf fila.Cells("C4").Value Is Nothing Then
        fila.Cells("C4").Value = 0
    ElseIf fila.Cells("C5").Value Is Nothing Then
        fila.Cells("C5").Value = 0
    ElseIf fila.Cells("> C5").Value Is Nothing Then
        fila.Cells("> C5").Value = 0
    End If
    sumaSimpleClaseBUS += CDb1(fila.Cells("BUS").Value)
    sumaSimpleClaseC2P += CDb1(fila.Cells("C2P").Value)
    sumaSimpleClaseC2G += CDb1(fila.Cells("C2G").Value)
    sumaSimpleClaseC3 += CDb1(fila.Cells("C3").Value)
    sumaSimpleClaseC4 += CDb1(fila.Cells("C4").Value)
    sumaSimpleClaseC5 += CDb1(fila.Cells("C5").Value)
    sumaSimpleClaseMayorC5 += CDb1(fila.Cells("> C5").Value)
Next

For Each fila As DataGridViewRow In dgvGeneralTandem.Rows
    If fila.Cells("General").Value Is Nothing Then
        fila.Cells("General").Value = 0
    End If
    sumaTandemGeneral += CDb1(fila.Cells("General").Value)
Next

For Each fila As DataGridViewRow In dgvClaseTandem.Rows
    If fila.Cells("BUS").Value Is Nothing Then
        fila.Cells("BUS").Value = 0
    ElseIf fila.Cells("C2P").Value Is Nothing Then
        fila.Cells("C2P").Value = 0
    ElseIf fila.Cells("C2G").Value Is Nothing Then
        fila.Cells("C2G").Value = 0
    ElseIf fila.Cells("C3").Value Is Nothing Then

```

```

        fila.Cells("C3").Value = 0
    ElseIf fila.Cells("C4").Value Is Nothing Then
        fila.Cells("C4").Value = 0
    ElseIf fila.Cells("C5").Value Is Nothing Then
        fila.Cells("C5").Value = 0
    ElseIf fila.Cells("> C5").Value Is Nothing Then
        fila.Cells("> C5").Value = 0
    End If
    sumaTandemClaseBUS += Cdbl(fila.Cells("BUS").Value)
    sumaTandemClaseC2P += Cdbl(fila.Cells("C2P").Value)
    sumaTandemClaseC2G += Cdbl(fila.Cells("C2G").Value)
    sumaTandemClaseC3 += Cdbl(fila.Cells("C3").Value)
    sumaTandemClaseC4 += Cdbl(fila.Cells("C4").Value)
    sumaTandemClaseC5 += Cdbl(fila.Cells("C5").Value)
    sumaTandemClaseMayorC5 += Cdbl(fila.Cells("> C5").Value)
Next

For Each fila As DataGridViewRow In dgvGeneralTridem.Rows
    If fila.Cells("General").Value Is Nothing Then
        fila.Cells("General").Value = 0
    End If
    sumaTridemGeneral += Cdbl(fila.Cells("General").Value)
Next

For Each fila As DataGridViewRow In dgvClaseTridem.Rows
    If fila.Cells("BUS").Value Is Nothing Then
        fila.Cells("BUS").Value = 0
    ElseIf fila.Cells("C2P").Value Is Nothing Then
        fila.Cells("C2P").Value = 0
    ElseIf fila.Cells("C2G").Value Is Nothing Then
        fila.Cells("C2G").Value = 0
    ElseIf fila.Cells("C3").Value Is Nothing Then
        fila.Cells("C3").Value = 0
    ElseIf fila.Cells("C4").Value Is Nothing Then
        fila.Cells("C4").Value = 0
    ElseIf fila.Cells("C5").Value Is Nothing Then
        fila.Cells("C5").Value = 0
    ElseIf fila.Cells("> C5").Value Is Nothing Then
        fila.Cells("> C5").Value = 0
    End If
    sumaTridemClaseBUS += Cdbl(fila.Cells("BUS").Value)
    sumaTridemClaseC2P += Cdbl(fila.Cells("C2P").Value)
    sumaTridemClaseC2G += Cdbl(fila.Cells("C2G").Value)
    sumaTridemClaseC3 += Cdbl(fila.Cells("C3").Value)
    sumaTridemClaseC4 += Cdbl(fila.Cells("C4").Value)
    sumaTridemClaseC5 += Cdbl(fila.Cells("C5").Value)
    sumaTridemClaseMayorC5 += Cdbl(fila.Cells("> C5").Value)
Next

sumaSimpleGeneral = Math.Round(sumaSimpleGeneral, 2)
sumaSimpleClaseBUS = Math.Round(sumaSimpleClaseBUS, 2)
sumaSimpleClaseC2P = Math.Round(sumaSimpleClaseC2P, 2)
sumaSimpleClaseC2G = Math.Round(sumaSimpleClaseC2G, 2)
sumaSimpleClaseC3 = Math.Round(sumaSimpleClaseC3, 2)
sumaSimpleClaseC4 = Math.Round(sumaSimpleClaseC4, 2)

```

```

sumaSimpleClaseC5 = Math.Round(sumaSimpleClaseC5, 2)
sumaSimpleClaseMayorC5 = Math.Round(sumaSimpleClaseMayorC5, 2)
sumaTandemGeneral = Math.Round(sumaTandemGeneral, 2)
sumaTandemClaseBUS = Math.Round(sumaTandemClaseBUS, 2)
sumaTandemClaseC2P = Math.Round(sumaTandemClaseC2P, 2)
sumaTandemClaseC2G = Math.Round(sumaTandemClaseC2G, 2)
sumaTandemClaseC3 = Math.Round(sumaTandemClaseC3, 2)
sumaTandemClaseC4 = Math.Round(sumaTandemClaseC4, 2)
sumaTandemClaseC5 = Math.Round(sumaTandemClaseC5, 2)
sumaTandemClaseMayorC5 = Math.Round(sumaTandemClaseMayorC5, 2)
sumaTridemGeneral = Math.Round(sumaTridemGeneral, 2)
sumaTridemClaseBUS = Math.Round(sumaTridemClaseBUS, 2)
sumaTridemClaseC2P = Math.Round(sumaTridemClaseC2P, 2)
sumaTridemClaseC2G = Math.Round(sumaTridemClaseC2G, 2)
sumaTridemClaseC3 = Math.Round(sumaTridemClaseC3, 2)
sumaTridemClaseC4 = Math.Round(sumaTridemClaseC4, 2)
sumaTridemClaseC5 = Math.Round(sumaTridemClaseC5, 2)
sumaTridemClaseMayorC5 = Math.Round(sumaTridemClaseMayorC5, 2)

End Sub
Private Sub btnGuardarEspectros_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnGuardarEspectros.Click
    SumarPorcentajes()
    If rbtnGeneralSimple.Checked = True Then
        If sumaSimpleGeneral <= 99.999 Or sumaSimpleGeneral >= 100.001 Then
            MsgBox("Total porcentajes (General) = " + CStr(sumaSimpleGeneral) +
vbLf + "La suma de los porcentajes del espectro de cargas para el eje simple no es
igual a 100. Por favor corrija los valores antes poder continuar.", vbInformation,
"Total porcentajes")
            GoTo salida1
        End If
    ElseIf rbtnClasesSimple.Checked = True Then
        If sumaSimpleClaseBUS <= 99.999 Or sumaSimpleClaseBUS >= 100.001 Or
sumaSimpleClaseC2G < 99.999 Or sumaSimpleClaseC2G > 100.001 Or sumaSimpleClaseC2P <
99.999 Or sumaSimpleClaseC2P > 100.001 Or sumaSimpleClaseC3 < 99.999 Or
sumaSimpleClaseC3 > 100.001 Or sumaSimpleClaseC4 < 99.999 Or sumaSimpleClaseC4 >
100.001 Or sumaSimpleClaseC5 < 99.999 Or sumaSimpleClaseC5 > 100.001 Or
sumaSimpleClaseMayorC5 < 99.999 Or sumaSimpleClaseMayorC5 > 100.001 Then
            MsgBox("Total porcentajes (BUS) = " + CStr(sumaSimpleClaseBUS) +
vbLf + "Total porcentajes (C2P) = " + CStr(sumaSimpleClaseC2P) + vbLf + "Total
porcentajes (C2G) = " + CStr(sumaSimpleClaseC2G) + vbLf + "Total porcentajes (C3) =
" + CStr(sumaSimpleClaseC3) + vbLf + "Total porcentajes (C4) = " +
CStr(sumaSimpleClaseC4) + vbLf + "Total porcentajes (C5) = " +
CStr(sumaSimpleClaseC5) + vbLf + "Total porcentajes (>C5) = " +
CStr(sumaSimpleClaseMayorC5) + vbLf + "La suma de los porcentajes del espectro de
cargas para el eje simple no es igual a 100. Por favor corrija los valores antes
poder continuar.", vbInformation, "Total porcentajes")
            GoTo salida1
        End If
    End If
    If rbtnGeneralTandem.Checked = True Then
        If sumaTandemGeneral <= 99.999 Or sumaTandemGeneral >= 100.001 Then
            MsgBox("Total porcentajes (General) = " + CStr(sumaTandemGeneral) +
vbLf + "La suma de los porcentajes del espectro de cargas para el eje Tandem no es

```

```

igual a 100. Por favor corrija los valores antes poder continuar.", vbInformation,
"Total porcentajes")
    GoTo salida1
End If
ElseIf rbtnClaseTandem.Checked = True Then
    If sumaTandemClaseBUS <= 99.999 Or sumaTandemClaseBUS >= 100.001 Or
sumaTandemClaseC2G < 99.999 Or sumaTandemClaseC2G > 100.001 Or sumaTandemClaseC2P <
99.999 Or sumaTandemClaseC2P > 100.001 Or sumaTandemClaseC3 < 99.999 Or
sumaTandemClaseC3 > 100.001 Or sumaTandemClaseC4 < 99.999 Or sumaTandemClaseC4 >
100.001 Or sumaTandemClaseC5 < 99.999 Or sumaTandemClaseC5 > 100.001 Or
sumaTandemClaseMayorC5 < 99.999 Or sumaTandemClaseMayorC5 > 100.001 Then
        MsgBox("Total porcentajes (BUS) = " + CStr(sumaTandemClaseBUS) +
vbLf + "Total porcentajes (C2P) = " + CStr(sumaTandemClaseC2P) + vbLf + "Total
porcentajes (C2G) = " + CStr(sumaTandemClaseC2G) + vbLf + "Total porcentajes (C3) =
" + CStr(sumaTandemClaseC3) + vbLf + "Total porcentajes (C4) = " +
CStr(sumaTandemClaseC4) + vbLf + "Total porcentajes (C5) = " +
CStr(sumaTandemClaseC5) + vbLf + "Total porcentajes (>C5) = " +
CStr(sumaTandemClaseMayorC5) + vbLf + "La suma de los porcentajes del espectro de
cargas para el eje Tandem no es igual a 100. Por favor corrija los valores antes
poder continuar.", vbInformation, "Total porcentajes")
        GoTo salida1
    End If
End If
If rbtnGeneralTridem.Checked = True Then
    If sumaTridemGeneral <= 99.999 Or sumaTridemGeneral >= 100.001 Then
        MsgBox("Total porcentajes (General) = " + CStr(sumaTridemGeneral) +
vbLf + "La suma de los porcentajes del espectro de cargas para el eje Tridem no es
igual a 100. Por favor corrija los valores antes poder continuar.", vbInformation,
"Total porcentajes")
        GoTo salida1
    End If
ElseIf rbtnClaseTridem.Checked = True Then
    If sumaTridemClaseBUS <= 99.999 Or sumaTridemClaseBUS >= 100.001 Or
sumaTridemClaseC2G < 99.999 Or sumaTridemClaseC2G > 100.001 Or sumaTridemClaseC2P <
99.999 Or sumaTridemClaseC2P > 100.001 Or sumaTridemClaseC3 < 99.999 Or
sumaTridemClaseC3 > 100.001 Or sumaTridemClaseC4 < 99.999 Or sumaTridemClaseC4 >
100.001 Or sumaTridemClaseC5 < 99.999 Or sumaTridemClaseC5 > 100.001 Or
sumaTridemClaseMayorC5 < 99.999 Or sumaTridemClaseMayorC5 > 100.001 Then
        MsgBox("Total porcentajes (BUS) = " + CStr(sumaTridemClaseBUS) +
vbLf + "Total porcentajes (C2P) = " + CStr(sumaTridemClaseC2P) + vbLf + "Total
porcentajes (C2G) = " + CStr(sumaTridemClaseC2G) + vbLf + "Total porcentajes (C3) =
" + CStr(sumaTridemClaseC3) + vbLf + "Total porcentajes (C4) = " +
CStr(sumaTridemClaseC4) + vbLf + "Total porcentajes (C5) = " +
CStr(sumaTridemClaseC5) + vbLf + "Total porcentajes (>C5) = " +
CStr(sumaTridemClaseMayorC5) + vbLf + "La suma de los porcentajes del espectro de
cargas para el eje Tridem no es igual a 100. Por favor corrija los valores antes
poder continuar.", vbInformation, "Total porcentajes")
        GoTo salida1
    End If
End If
With EspectrosCarga
    .EspectroSimpleGeneral = dgvGeneralSimple.DataSource
    .EspectroTandemGeneral = dgvGeneralTandem.DataSource
    .EspectroTridemGeneral = dgvGeneralTridem.DataSource
    .EspectroSimplePorClase = dgvClaseSimple.DataSource
End With

```

```

        .EspectroTandemPorClase = dgvClaseTandem.DataSource
        .EspectroTridemPorClase = dgvClaseTridem.DataSource
    If rbtnGeneralSimple.Checked = True Then
        .TipoEspectroSimple = "General"
    Else
        .TipoEspectroSimple = "Por clase"
    End If

    If rbtnGeneralTandem.Checked = True Then
        .TipoEspectroTandem = "General"
    Else
        .TipoEspectroTandem = "Por clase"
    End If
    If rbtnGeneralTridem.Checked = True Then
        .TipoEspectroTridem = "General"
    Else
        .TipoEspectroTridem = "Por clase"
    End If
End With
If saveEspectro.ShowDialog() = DialogResult.OK Then
    strmEspectrosCarga = saveEspectro.OpenFile()
    Using strmW As StreamWriter = New StreamWriter(strmEspectrosCarga)
        srlzEspectrosCarga.Serialize(strmW, EspectrosCarga)
        strmW.Close()
    End Using
    If (strmEspectrosCarga IsNot Nothing) Then
        strmEspectrosCarga.Close()
    End If
End If
salida1:
End Sub
Private Sub btnCargarEspectros_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCargarEspectros.Click
    If openEspectro.ShowDialog() = System.Windows.Forms.DialogResult.OK Then
        Try
            strmEspectrosCarga = openEspectro.OpenFile()
            Dim strmR As New StreamReader(strmEspectrosCarga)
            EspectrosCarga = srlzEspectrosCarga.Deserialize(strmR)
            strmR.Close()
            'ASIGNAR INFORMACIÓN A CONTROLES
            With EspectrosCarga
                If .TipoEspectroSimple = "General" Then
                    rbtnGeneralSimple.Checked = True
                Else
                    rbtnClasesSimple.Checked = True
                End If
                If .TipoEspectroTandem = "General" Then
                    rbtnGeneralTandem.Checked = True
                Else
                    rbtnClaseTandem.Checked = True
                End If
                If .TipoEspectroTridem = "General" Then
                    rbtnGeneralTridem.Checked = True
                Else

```

```

        rbtnClaseTridem.Checked = True
    End If
    dgvGeneralSimple.DataSource = .EspectroSimpleGeneral
    dgvGeneralTandem.DataSource = .EspectroTandemGeneral
    dgvGeneralTridem.DataSource = .EspectroTridemGeneral
    dgvClaseSimple.DataSource = .EspectroSimplePorClase
    dgvClaseTandem.DataSource = .EspectroTandemPorClase
    dgvClaseTridem.DataSource = .EspectroTridemPorClase
End With
If (strmEspectrosCarga IsNot Nothing) Then
    strmEspectrosCarga.Close()
End If
Catch Ex As Exception
    MessageBox.Show("No se puede leer el archivo. Causa: " & Ex.Message)
Finally
    If (strmEspectrosCarga IsNot Nothing) Then
        strmEspectrosCarga.Close()
    End If
End Try
End If
End Sub
Private Sub DistribucionCargaEje_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    'FUENTES TITULOS
    Dim fuenteTitulos As New
Font(dgvGeneralSimple.DefaultCellStyle.Font.FontFamily, 7, FontStyle.Bold)
'Simple
dgvGeneralSimple.RowHeadersDefaultCellStyle.Font = fuenteTitulos
dgvGeneralSimple.ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
dgvClaseSimple.RowHeadersDefaultCellStyle.Font = fuenteTitulos
dgvClaseSimple.ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
'Tandem
dgvGeneralTandem.RowHeadersDefaultCellStyle.Font = fuenteTitulos
dgvGeneralTandem.ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
dgvClaseTandem.RowHeadersDefaultCellStyle.Font = fuenteTitulos
dgvClaseTandem.ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
'Tridem
dgvGeneralTridem.RowHeadersDefaultCellStyle.Font = fuenteTitulos
dgvGeneralTridem.ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
dgvClaseTridem.RowHeadersDefaultCellStyle.Font = fuenteTitulos
dgvClaseTridem.ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
'ASIGNAR VALORES
With glInformacionTrafico
    dgvGeneralSimple.DataSource = .EspectroCarga.EspectroSimpleGeneral
    dgvClaseSimple.DataSource = .EspectroCarga.EspectroSimplePorClase
    dgvGeneralTandem.DataSource = .EspectroCarga.EspectroTandemGeneral
    dgvClaseTandem.DataSource = .EspectroCarga.EspectroTandemPorClase
    dgvGeneralTridem.DataSource = .EspectroCarga.EspectroTridemGeneral
    dgvClaseTridem.DataSource = .EspectroCarga.EspectroTridemPorClase
End With
'Simple
With dgvGeneralSimple
    .RowHeadersWidth = 60
    For i = 0 To .RowCount - 1
        .Rows(i).HeaderCell.Value = CStr((i + 1) * 10)
    
```

```

Next
    .Columns(0).Width = 60
    .Columns(0).SortMode = DataGridViewColumnSortMode.NotSortable
    .RowHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
    .RowHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
    .ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
    .ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
    .RowHeadersVisible = True
    .EditMode = DataGridViewEditMode.EditOnEnter
    .DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
    .AllowUserToResizeColumns = False
    .AllowUserToResizeRows = False
    .AllowUserToAddRows = False
    .AllowUserToDeleteRows = False
    .AllowUserToOrderColumns = False
    .MultiSelect = False
    .TopLeftHeaderCell.Value = "Carga [KN]"
End With
With dgvClaseSimple
    .DataSource = dsSimplePorClase.Tables(0)
    .RowHeadersWidth = 60
    Dim row As DataGridViewRow
    For Each row In .Rows
        row.HeaderCell.Value = CStr((row.Index + 1) * 10)
    Next
    Dim column As DataGridViewColumn
    For Each column In .Columns
        column.Width = 60
    Next
    For i = 0 To .Columns.Count - 1
        .Columns.Item(i).SortMode = DataGridViewColumnSortMode.NotSortable
    Next i
    .RowHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
    .RowHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
    .ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
    .ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
    .RowHeadersVisible = True
    .EditMode = DataGridViewEditMode.EditOnEnter
    .DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
    .AllowUserToResizeColumns = False
    .AllowUserToResizeRows = False
    .AllowUserToAddRows = False
    .AllowUserToDeleteRows = False
    .AllowUserToOrderColumns = False
    .MultiSelect = False
    .TopLeftHeaderCell.Value = "Carga [KN]"
End With
'Tandem
With dgvGeneralTandem
    .DataSource = dsTandemGeneral.Tables(0)
    .RowHeadersWidth = 60

```

```

For i = 0 To .RowCount - 1
    .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
Next
.Columns(0).Width = 60
.Columns(0).SortMode = DataGridViewColumnSortMode.NotSortable
.RowHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
.RowHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
.ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
.ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
.RowHeadersVisible = True
.EditMode = DataGridViewEditMode.EditOnEnter
.DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
.AllowUserToResizeColumns = False
.AllowUserToResizeRows = False
.AllowUserToAddRows = False
.AllowUserToDeleteRows = False
.AllowUserToOrderColumns = False
.MultiSelect = False
.TopLeftHeaderCell.Value = "Carga [KN]"
End With
With dgvClaseTandem
.DataSource = dsTandemPorClase.Tables(0)
.RowHeadersWidth = 60
For i = 0 To .RowCount - 1
    .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
Next
Dim column As DataGridViewColumn
For Each column In .Columns
    column.Width = 60
Next
For i = 0 To .Columns.Count - 1
    .Columns.Item(i).SortMode = DataGridViewColumnSortMode.NotSortable
Next i
.RowHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
.RowHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
.ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
.ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
.RowHeadersVisible = True
.EditMode = DataGridViewEditMode.EditOnEnter
.DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
.AllowUserToResizeColumns = False
.AllowUserToResizeRows = False
.AllowUserToAddRows = False
.AllowUserToDeleteRows = False
.AllowUserToOrderColumns = False
.MultiSelect = False
.TopLeftHeaderCell.Value = "Carga [KN]"
End With
'Tridem
With dgvGeneralTridem
.DataSource = dsTridemGeneral.Tables(0)

```

```

        .RowHeadersWidth = 60
    For i = 0 To .RowCount - 1
        .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
    Next
    .Columns(0).Width = 60
    .Columns(0).SortMode = DataGridViewColumnSortMode.NotSortable
    .RowHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
    .RowHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
    .ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
    .ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
    .RowHeadersVisible = True
    .EditMode = DataGridViewEditMode.EditOnEnter
    .DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
    .AllowUserToResizeColumns = False
    .AllowUserToResizeRows = False
    .AllowUserToAddRows = False
    .AllowUserToDeleteRows = False
    .AllowUserToOrderColumns = False
    .MultiSelect = False
    .TopLeftHeaderCell.Value = "Carga [KN]"
End With
With dgvClaseTridem
    .DataSource = dsTridemPorClase.Tables(0)
    .RowHeadersWidth = 60
    For i = 0 To .RowCount - 1
        .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
    Next
    Dim column As DataGridViewColumn
    For Each column In .Columns
        column.Width = 60
    Next
    For i = 0 To .Columns.Count - 1
        .Columns.Item(i).SortMode = DataGridViewColumnSortMode.NotSortable
    Next i
    .RowHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
    .RowHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
    .ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
    .ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
    .RowHeadersVisible = True
    .EditMode = DataGridViewEditMode.EditOnEnter
    .DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
    .AllowUserToResizeColumns = False
    .AllowUserToResizeRows = False
    .AllowUserToAddRows = False
    .AllowUserToDeleteRows = False
    .AllowUserToOrderColumns = False
    .MultiSelect = False
    .TopLeftHeaderCell.Value = "Carga [KN]"
End With
'VISIBILIDAD INICIAL
'Datagridviews

```

```

    dgvGeneralSimple.Visible = True
    dgvClaseSimple.Visible = False
    dgvGeneralTandem.Visible = True
    dgvClaseTandem.Visible = False
    dgvGeneralTridem.Visible = True
    dgvClaseTridem.Visible = False
    'Groupboxes
    grpSimple.Visible = True
    grpTandem.Visible = False
    grpTridem.Visible = False
End Sub
'CAMBIO DE TIPO DE ESPECTRO (GENERAL - POR CLASE)
'Simple
Private Sub rbtnGeneralSimple_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles rbtnGeneralSimple.CheckedChanged
    If rbtnGeneralSimple.Checked = True Then
        If dgvClaseSimple.RowCount > 0 Then
            dgvClaseSimple.Visible = False
            dgvGeneralSimple.Visible = True
        End If
    End If
End Sub
Private Sub rbtnClasesSimple_CheckedChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles rbtnClasesSimple.CheckedChanged
    If rbtnClasesSimple.Checked = True Then
        dgvClaseSimple.Visible = True
        dgvGeneralSimple.Visible = False
    End If
End Sub
'Tandem
Private Sub rbtnGeneralTandem_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles rbtnGeneralTandem.CheckedChanged
    If rbtnGeneralTandem.Checked = True Then
        If dgvClaseTandem.RowCount > 0 Then
            dgvClaseTandem.Visible = False
            rbtnGeneralTandem.Visible = True
        End If
    End If
End Sub
Private Sub rbtnClaseTandem_CheckedChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles rbtnClaseTandem.CheckedChanged
    If rbtnClaseTandem.Checked = True Then
        dgvClaseTandem.Visible = True
        dgvGeneralTandem.Visible = False
    End If
End Sub
'Tridem
Private Sub rbtnGeneralTridem_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles rbtnGeneralTridem.CheckedChanged
    If rbtnGeneralTridem.Checked = True Then
        If dgvClaseTridem.RowCount > 0 Then
            dgvClaseTridem.Visible = False
            dgvGeneralTridem.Visible = True
        End If
    End If
End Sub

```

```

End Sub
Private Sub rbtnClasesTridem_CheckedChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles rbtnClaseTridem.CheckedChanged
    If rbtnClaseTridem.Checked = True Then
        dgvClaseTridem.Visible = True
        dgvGeneralTridem.Visible = False
    End If
End Sub
'CAMBIO DE GROUPBOXES SEGÚN TIPO DE EJE
Private Sub rbtnSimple_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnSimple.CheckedChanged
    If rbtnSimple.Checked = True Then
        If dgvClaseSimple.RowCount > 0 Then
            grpSimple.Visible = True
            grpTandem.Visible = False
            grpTridem.Visible = False
        End If
    End If
End Sub
Private Sub rbtnTandem_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTandem.CheckedChanged
    If rbtnTandem.Checked = True Then
        grpSimple.Visible = False
        grpTandem.Visible = True
        grpTridem.Visible = False
    End If
End Sub
Private Sub rbtnTridem_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTridem.CheckedChanged
    If rbtnTridem.Checked = True Then
        grpSimple.Visible = False
        grpTandem.Visible = False
        grpTridem.Visible = True
    End If
End Sub
'DATOS DE LOS DATAGRIDVIEW SEGÚN TIPO DE TRANSITO
'Simple
Private Sub rbtnSimple1_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnSimple1.CheckedChanged
    If rbtnSimple1.Checked = True Then
        If dgvClaseSimple.RowCount > 0 Then
            dgvClaseSimple.DataSource = dsSimplePorClase.Tables(0)
            dgvGeneralSimple.DataSource = dsSimpleGeneral.Tables(0)
            With dgvClaseSimple
                For i = 0 To .RowCount - 1
                    .Rows(i).HeaderCell.Value = CStr((i + 1) * 10)
                Next
            End With
            With dgvGeneralSimple
                For i = 0 To .RowCount - 1
                    .Rows(i).HeaderCell.Value = CStr((i + 1) * 10)
                Next
            End With
        End If
    End If
End Sub

```

```

End Sub
Private Sub rbtnSimple2_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnSimple2.CheckedChanged
    If rbtnSimple2.Checked = True Then
        dgvClaseSimple.DataSource = dsSimplePorClase.Tables(1)
        dgvGeneralSimple.DataSource = dsSimpleGeneral.Tables(1)
        With dgvClaseSimple
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr((i + 1) * 10)
            Next
        End With
        With dgvGeneralSimple
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr((i + 1) * 10)
            Next
        End With
    End If
End Sub
Private Sub rbtnSimple3_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnSimple3.CheckedChanged
    If rbtnSimple3.Checked = True Then
        dgvClaseSimple.DataSource = dsSimplePorClase.Tables(2)
        dgvGeneralSimple.DataSource = dsSimpleGeneral.Tables(2)
        With dgvClaseSimple
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr((i + 1) * 10)
            Next
        End With
        With dgvGeneralSimple
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr((i + 1) * 10)
            Next
        End With
    End If
End Sub
Private Sub rbtnSimple4_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnSimple4.CheckedChanged
    If rbtnSimple4.Checked = True Then
        dgvClaseSimple.DataSource = dsSimplePorClase.Tables(3)
        dgvGeneralSimple.DataSource = dsSimpleGeneral.Tables(3)
        With dgvClaseSimple
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr((i + 1) * 10)
            Next
        End With
        With dgvGeneralSimple
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr((i + 1) * 10)
            Next
        End With
    End If
End Sub
Private Sub rbtnSimple5_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnSimple5.CheckedChanged
    If rbtnSimple5.Checked = True Then

```

```

dgvClaseSimple.DataSource = dsSimplePorClase.Tables(4)
dgvGeneralSimple.DataSource = dsSimpleGeneral.Tables(4)
With dgvClaseSimple
    For i = 0 To .RowCount - 1
        .Rows(i).HeaderCell.Value = CStr((i + 1) * 10)
    Next
End With

With dgvGeneralSimple
    For i = 0 To .RowCount - 1
        .Rows(i).HeaderCell.Value = CStr((i + 1) * 10)
    Next
End With
End If
End Sub
Private Sub rbtnSimple6_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnSimple6.CheckedChanged
    If rbtnSimple6.Checked = True Then
        dgvClaseSimple.DataSource = dsSimplePorClase.Tables(5)
        dgvGeneralSimple.DataSource = dsSimpleGeneral.Tables(5)
        With dgvClaseSimple
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr((i + 1) * 10)
            Next
        End With
        With dgvGeneralSimple
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr((i + 1) * 10)
            Next
        End With
    End If
End Sub
'Tandem
Private Sub rbtnTandem1_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTandem1.CheckedChanged
    If rbtnTandem1.Checked = True Then
        If dgvClaseSimple.RowCount > 0 Then
            dgvClaseTandem.DataSource = dsTandemPorClase.Tables(0)
            dgvGeneralTandem.DataSource = dsTandemGeneral.Tables(0)
            With dgvClaseTandem
                For i = 0 To .RowCount - 1
                    .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
                Next
            End With
            With dgvGeneralTandem
                For i = 0 To .RowCount - 1
                    .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
                Next
            End With
        End If
    End If
End Sub
Private Sub rbtnTandem2_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTandem2.CheckedChanged
    If rbtnTandem2.Checked = True Then

```

```

dgvClaseTandem.DataSource = dsTandemPorClase.Tables(1)
dgvGeneralTandem.DataSource = dsTandemGeneral.Tables(1)
With dgvClaseTandem
    For i = 0 To .RowCount - 1
        .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
    Next
End With

With dgvGeneralTandem
    For i = 0 To .RowCount - 1
        .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
    Next
End With
End If
End Sub
Private Sub rbtnTandem3_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTandem3.CheckedChanged
    If rbtnTandem3.Checked = True Then
        dgvClaseTandem.DataSource = dsTandemPorClase.Tables(2)
        dgvGeneralTandem.DataSource = dsTandemGeneral.Tables(2)
        With dgvClaseTandem
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
            Next
        End With
        With dgvGeneralTandem
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
            Next
        End With
    End If
End Sub
Private Sub rbtnTandem4_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTandem4.CheckedChanged
    If rbtnTandem4.Checked = True Then
        dgvClaseTandem.DataSource = dsTandemPorClase.Tables(3)
        dgvGeneralTandem.DataSource = dsTandemGeneral.Tables(3)
        With dgvClaseTandem
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
            Next
        End With
        With dgvGeneralTandem
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
            Next
        End With
    End If
End Sub
Private Sub rbtnTandem5_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTandem5.CheckedChanged
    If rbtnTandem5.Checked = True Then
        dgvClaseTandem.DataSource = dsTandemPorClase.Tables(4)
        dgvGeneralTandem.DataSource = dsTandemGeneral.Tables(4)
        With dgvClaseTandem

```

```

        For i = 0 To .RowCount - 1
            .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
        Next
    End With
    With dgvGeneralTandem
        For i = 0 To .RowCount - 1
            .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
        Next
    End With
End If
End Sub
Private Sub rbtnTandem6_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTandem6.CheckedChanged
    If rbtnTandem6.Checked = True Then
        dgvClaseTandem.DataSource = dsTandemPorClase.Tables(5)
        dgvGeneralTandem.DataSource = dsTandemGeneral.Tables(5)
        With dgvClaseTandem
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
            Next
        End With
        With dgvGeneralTandem
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
            Next
        End With
    End If
End Sub
'Tridem
Private Sub rbtnTridem1_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTridem1.CheckedChanged
    If rbtnTridem1.Checked = True Then
        If dgvClaseSimple.RowCount > 0 Then
            dgvClaseTridem.DataSource = dsTridemPorClase.Tables(0)
            dgvGeneralTridem.DataSource = dsTridemGeneral.Tables(0)
            With dgvClaseTridem
                For i = 0 To .RowCount - 1
                    .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
                Next
            End With
            With dgvGeneralTridem
                For i = 0 To .RowCount - 1
                    .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
                Next
            End With
        End If
    End If
End Sub
Private Sub rbtnTridem2_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTridem2.CheckedChanged
    If rbtnTridem2.Checked = True Then
        dgvClaseTridem.DataSource = dsTridemPorClase.Tables(1)
        dgvGeneralTridem.DataSource = dsTridemGeneral.Tables(1)
        With dgvClaseTridem
            For i = 0 To .RowCount - 1

```

```

        .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
    Next
End With
With dgvGeneralTridem
    For i = 0 To .RowCount - 1
        .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
    Next
End With
End If
End Sub
Private Sub rbtnTridem3_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTridem3.CheckedChanged
    If rbtnTridem3.Checked = True Then
        dgvClaseTridem.DataSource = dsTridemPorClase.Tables(2)
        dgvGeneralTridem.DataSource = dsTridemGeneral.Tables(2)
        With dgvClaseTridem
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
            Next
        End With
        With dgvGeneralTridem
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
            Next
        End With
    End If
End Sub
Private Sub rbtnTridem4_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTridem4.CheckedChanged
    If rbtnTridem4.Checked = True Then
        dgvClaseTridem.DataSource = dsTridemPorClase.Tables(3)
        dgvGeneralTridem.DataSource = dsTridemGeneral.Tables(3)
        With dgvClaseTridem
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
            Next
        End With
        With dgvGeneralTridem
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
            Next
        End With
    End If
End Sub
Private Sub rbtnTridem5_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTridem5.CheckedChanged
    If rbtnTridem5.Checked = True Then
        dgvClaseTridem.DataSource = dsTridemPorClase.Tables(4)
        dgvGeneralTridem.DataSource = dsTridemGeneral.Tables(4)
        With dgvClaseTridem
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
            Next
        End With
        With dgvGeneralTridem

```

```

        For i = 0 To .RowCount - 1
            .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
        Next
    End With
End If
End Sub
Private Sub rbtnTridem6_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnTridem6.CheckedChanged
    If rbtnTridem6.Checked = True Then
        dgvClaseTridem.DataSource = dsTridemPorClase.Tables(5)
        dgvGeneralTridem.DataSource = dsTridemGeneral.Tables(5)
        With dgvClaseTridem
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
            Next
        End With
        With dgvGeneralTridem
            For i = 0 To .RowCount - 1
                .Rows(i).HeaderCell.Value = CStr(((2 * i) + 3) * 10)
            Next
        End With
    End If
End Sub
'BOTONES ACEPTAR/CANCELAR
Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub
Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    SumarPorcentajes()
    If rbtnGeneralSimple.Checked = True Then
        If sumaSimpleGeneral <= 99.999 Or sumaSimpleGeneral >= 100.001 Then
            MsgBox("Total porcentajes (General) = " + CStr(sumaSimpleGeneral) +
vbLf + "La suma de los porcentajes del espectro de cargas para el eje simple no es
igual a 100. Por favor corrija los valores antes poder continuar.", vbInformation,
"Total porcentajes")
            GoTo salida1
        End If
    ElseIf rbtnClasesSimple.Checked = True Then
        If sumaSimpleClaseBUS <= 99.999 Or sumaSimpleClaseBUS >= 100.001 Or
sumaSimpleClaseC2G < 99.999 Or sumaSimpleClaseC2G > 100.001 Or sumaSimpleClaseC2P <
99.999 Or sumaSimpleClaseC2P > 100.001 Or sumaSimpleClaseC3 < 99.999 Or
sumaSimpleClaseC3 > 100.001 Or sumaSimpleClaseC4 < 99.999 Or sumaSimpleClaseC4 >
100.001 Or sumaSimpleClaseC5 < 99.999 Or sumaSimpleClaseC5 > 100.001 Or
sumaSimpleClaseMayorC5 < 99.999 Or sumaSimpleClaseMayorC5 > 100.001 Then
            MsgBox("Total porcentajes (BUS) = " + CStr(sumaSimpleClaseBUS) +
vbLf + "Total porcentajes (C2P) = " + CStr(sumaSimpleClaseC2P) + vbLf + "Total
porcentajes (C2G) = " + CStr(sumaSimpleClaseC2G) + vbLf + "Total porcentajes (C3) =
" + CStr(sumaSimpleClaseC3) + vbLf + "Total porcentajes (C4) = " +
CStr(sumaSimpleClaseC4) + vbLf + "Total porcentajes (C5) = " +
CStr(sumaSimpleClaseC5) + vbLf + "Total porcentajes (>C5) = " +
CStr(sumaSimpleClaseMayorC5) + vbLf + "La suma de los porcentajes del espectro de
cargas para el eje simple no es igual a 100. Por favor corrija los valores antes
poder continuar.", vbInformation, "Total porcentajes")
        End If
    End If
End Sub

```

```

        GoTo salida1
    End If
End If
If rbtnGeneralTandem.Checked = True Then
    If sumaTandemGeneral <= 99.999 Or sumaTandemGeneral >= 100.001 Then
        MsgBox("Total porcentajes (General) = " + CStr(sumaTandemGeneral) +
vbLf + "La suma de los porcentajes del espectro de cargas para el eje Tandem no es
igual a 100. Por favor corrija los valores antes poder continuar.", vbInformation,
"Total porcentajes")
        GoTo salida1
    End If
    ElseIf rbtnClaseTandem.Checked = True Then
        If sumaTandemClaseBUS <= 99.999 Or sumaTandemClaseBUS >= 100.001 Or
sumaTandemClaseC2G < 99.999 Or sumaTandemClaseC2G > 100.001 Or sumaTandemClaseC2P <
99.999 Or sumaTandemClaseC2P > 100.001 Or sumaTandemClaseC3 < 99.999 Or
sumaTandemClaseC3 > 100.001 Or sumaTandemClaseC4 < 99.999 Or sumaTandemClaseC4 >
100.001 Or sumaTandemClaseC5 < 99.999 Or sumaTandemClaseC5 > 100.001 Or
sumaTandemClaseMayorC5 < 99.999 Or sumaTandemClaseMayorC5 > 100.001 Then
            MsgBox("Total porcentajes (BUS) = " + CStr(sumaTandemClaseBUS) +
vbLf + "Total porcentajes (C2P) = " + CStr(sumaTandemClaseC2P) + vbLf + "Total
porcentajes (C2G) = " + CStr(sumaTandemClaseC2G) + vbLf + "Total porcentajes (C3) =
" + CStr(sumaTandemClaseC3) + vbLf + "Total porcentajes (C4) = " +
CStr(sumaTandemClaseC4) + vbLf + "Total porcentajes (C5) = " +
CStr(sumaTandemClaseC5) + vbLf + "Total porcentajes (>C5) = " +
CStr(sumaTandemClaseMayorC5) + vbLf + "La suma de los porcentajes del espectro de
cargas para el eje Tandem no es igual a 100. Por favor corrija los valores antes
poder continuar.", vbInformation, "Total porcentajes")
            GoTo salida1
        End If
    End If
    If rbtnGeneralTridem.Checked = True Then
        If sumaTridemGeneral <= 99.999 Or sumaTridemGeneral >= 100.001 Then
            MsgBox("Total porcentajes (General) = " + CStr(sumaTridemGeneral) +
vbLf + "La suma de los porcentajes del espectro de cargas para el eje Tridem no es
igual a 100. Por favor corrija los valores antes poder continuar.", vbInformation,
"Total porcentajes")
            GoTo salida1
        End If
        ElseIf rbtnClaseTridem.Checked = True Then
            If sumaTridemClaseBUS <= 99.999 Or sumaTridemClaseBUS >= 100.001 Or
sumaTridemClaseC2G < 99.999 Or sumaTridemClaseC2G > 100.001 Or sumaTridemClaseC2P <
99.999 Or sumaTridemClaseC2P > 100.001 Or sumaTridemClaseC3 < 99.999 Or
sumaTridemClaseC3 > 100.001 Or sumaTridemClaseC4 < 99.999 Or sumaTridemClaseC4 >
100.001 Or sumaTridemClaseC5 < 99.999 Or sumaTridemClaseC5 > 100.001 Or
sumaTridemClaseMayorC5 < 99.999 Or sumaTridemClaseMayorC5 > 100.001 Then
                MsgBox("Total porcentajes (BUS) = " + CStr(sumaTridemClaseBUS) +
vbLf + "Total porcentajes (C2P) = " + CStr(sumaTridemClaseC2P) + vbLf + "Total
porcentajes (C2G) = " + CStr(sumaTridemClaseC2G) + vbLf + "Total porcentajes (C3) =
" + CStr(sumaTridemClaseC3) + vbLf + "Total porcentajes (C4) = " +
CStr(sumaTridemClaseC4) + vbLf + "Total porcentajes (C5) = " +
CStr(sumaTridemClaseC5) + vbLf + "Total porcentajes (>C5) = " +
CStr(sumaTridemClaseMayorC5) + vbLf + "La suma de los porcentajes del espectro de
cargas para el eje Tridem no es igual a 100. Por favor corrija los valores antes
poder continuar.", vbInformation, "Total porcentajes")
                GoTo salida1
            End If
        End If
    End If
End If

```

```

    End If
End If
With EspectrosCarga
    .EspectroSimpleGeneral = dgvGeneralSimple.DataSource
    .EspectroTandemGeneral = dgvGeneralTandem.DataSource
    .EspectroTridemGeneral = dgvGeneralTridem.DataSource
    .EspectroSimplePorClase = dgvClaseSimple.DataSource
    .EspectroTandemPorClase = dgvClaseTandem.DataSource
    .EspectroTridemPorClase = dgvClaseTridem.DataSource
    If rbtnGeneralSimple.Checked = True Then
        .TipoEspectroSimple = "General"
    Else
        .TipoEspectroSimple = "Por clase"
    End If
    If rbtnGeneralTandem.Checked = True Then
        .TipoEspectroTandem = "General"
    Else
        .TipoEspectroTandem = "Por clase"
    End If
    If rbtnGeneralTridem.Checked = True Then
        .TipoEspectroTridem = "General"
    Else
        .TipoEspectroTridem = "Por clase"
    End If
End With
glInformacionTrafico.EspectroCarga = EspectrosCarga
Me.Close()
salida1:
    End Sub
End Class

```

Factores de crecimiento

```

Public Class FacCrec
    'INICIO
    Private Sub FacCrec_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        With glInformacionTrafico
            If .FactCrecimiento.TipoCrecimiento = "Ninguno" Then
                rbtnNinguno.Checked = True
            ElseIf .FactCrecimiento.TipoCrecimiento = "Lineal" Then
                rbtnLineal.Checked = True
            ElseIf .FactCrecimiento.TipoCrecimiento = "Compuesto" Then
                rbtnCompuesto.Checked = True
            End If

            If .FactCrecimiento.UnicoValor = False Then
                checkPorClase.Checked = True
            Else
                checkPorClase.Checked = False
            End If

            txtFactorGeneral.Text = CStr(.FactCrecimiento.FactorGeneral)
            txtBus.Text = CStr(.FactCrecimiento.FactorPorClase(0))
            txtC2P.Text = CStr(.FactCrecimiento.FactorPorClase(1))
            txtC2G.Text = CStr(.FactCrecimiento.FactorPorClase(2))
            txtC3.Text = CStr(.FactCrecimiento.FactorPorClase(3))
        End With
    End Sub
End Class

```

```

        txtC4.Text = CStr(.FactCrecimiento.FactorPorClase(4))
        txtC5.Text = CStr(.FactCrecimiento.FactorPorClase(5))
        txtMayorC5.Text = CStr(.FactCrecimiento.FactorPorClase(6))
    End With
End Sub

'Tipo Crecimiento
Private Sub rbtnNinguno_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnNinguno.CheckedChanged
    grpGeneral.Enabled = False
    grpPorClase.Enabled = False
    checkPorClase.Enabled = False
End Sub
Private Sub rbtnLineal_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnLineal.CheckedChanged
    checkPorClase.Enabled = True
    grpPorClase.Enabled = True
    grpGeneral.Enabled = True
End Sub
Private Sub rbtnCompuesto_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles rbtnCompuesto.CheckedChanged
    checkPorClase.Enabled = True
    grpPorClase.Enabled = True
    grpGeneral.Enabled = True
End Sub
'MOSTRAR/OCULTAR FACTORES DE CRECIMIENTO POR CLASE
Private Sub checkPorClase_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles checkPorClase.CheckedChanged
    If checkPorClase.Checked = True Then
        grpGeneral.Visible = False
        grpPorClase.Visible = True
    Else
        grpGeneral.Visible = True
        grpPorClase.Visible = False
    End If
End Sub
'BOTONES ACEPTAR/CANCELAR
Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub
Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    With glInformacionTrafico
        .FactCrecimiento = glFacCrecimiento
        If rbtnNinguno.Checked = True Then
            .FactCrecimiento.TipoCrecimiento = "Ninguno"
        ElseIf rbtnLineal.Checked = True Then
            .FactCrecimiento.TipoCrecimiento = "Lineal"
        ElseIf rbtnCompuesto.Checked = True Then
            .FactCrecimiento.TipoCrecimiento = "Compuesto"
        End If
        If checkPorClase.Checked = True Then
            .FactCrecimiento.UnicoValor = False
        End If
    End With
End Sub

```

```

        Else
            .FactCrecimiento.UnicoValor = True
        End If
        .FactCrecimiento.FactorGeneral = CDb1(txtFactorGeneral.Text)
        .FactCrecimiento.FactorPorClase = {CDb1(txtBus.Text), CDb1(txtC2P.Text),
CDb1(txtC2G.Text), CDb1(txtC3.Text), CDb1(txtC4.Text), CDb1(txtC5.Text),
CDb1(txtMayorC5.Text)}
        End With
        Me.Close()
    End Sub
End Class

```

Anexo C.2.2 Clases

Tráfico

```

Public Class ClaseTrafico
    Private miTPDAT As Double
    Private miFactorDir As Double
    Private miFactorCarril As Double
    Private miNumCarril As Integer
    Private miVelOper As Double
    Private miDistrClase As ClaseDistrClase
    Private miDistrHoraria As ClaseDistrHoraria
    Private miFactorMeses As ClaseFactMensuales
    Private miEspectroCarga As ClaseEspectroCarga
    Private miMediaHuella As Double
    Private miDesvHuella As Double
    Private miAnchoCarril As Double
    Private miBatalla As ClaseBatalla
    Private miConfiguracionEje As ClaseConfiguracionEje
    Private miNumEjesClase As DataTable
    Private miFactCrecimiento As ClaseFactCrecimiento
    Public Property TPDAT() As Double
        Get
            TPDAT = miTPDAT
        End Get
        Set(ByVal value As Double)
            miTPDAT = value
        End Set
    End Property
    Public Property FactorDir As Double
        Get
            FactorDir = miFactorDir
        End Get
        Set(ByVal value As Double)
            miFactorDir = value
        End Set
    End Property
    Public Property FactorCarril As Double
        Get
            FactorCarril = miFactorCarril
        End Get
        Set(ByVal value As Double)
            miFactorCarril = value
        End Set
    End Property

```

```

    End Set
End Property
Public Property NumCarril As Integer
    Get
        NumCarril = miNumCarril
    End Get

    Set(ByVal value As Integer)
        miNumCarril = value
    End Set
End Property
Public Property VelOper As Double
    Get
        VelOper = miVelOper
    End Get
    Set(ByVal value As Double)
        miVelOper = value
    End Set
End Property
Public Property DistrClase() As ClaseDistrClase
    Get
        DistrClase = miDistrClase
    End Get
    Set(ByVal value As ClaseDistrClase)
        miDistrClase = value
    End Set
End Property
Public Property DistrHoraria() As ClaseDistrHoraria
    Get
        DistrHoraria = miDistrHoraria
    End Get
    Set(ByVal value As ClaseDistrHoraria)
        miDistrHoraria = value
    End Set
End Property
Public Property FactorMeses As ClaseFactMensuales
    Get
        FactorMeses = miFactorMeses
    End Get
    Set(ByVal value As ClaseFactMensuales)
        miFactorMeses = value
    End Set
End Property
Public Property EspectroCarga As ClaseEspectroCarga
    Get
        EspectroCarga = miEspectroCarga
    End Get
    Set(ByVal value As ClaseEspectroCarga)
        miEspectroCarga = value
    End Set
End Property
Public Property MediaHuella As Double
    Get
        MediaHuella = miMediaHuella
    End Get

```

```

        Set(ByVal value As Double)
            miMediaHuella = value
        End Set
    End Property
    Public Property DesvHuella As Double

        Get
            DesvHuella = miDesvHuella
        End Get
        Set(ByVal value As Double)
            miDesvHuella = value
        End Set
    End Property
    Public Property AnchoCarril As Double
        Get
            AnchoCarril = miAnchoCarril
        End Get
        Set(ByVal value As Double)
            miAnchoCarril = value
        End Set
    End Property
    Public Property Batalla As ClaseBatalla
        Get
            Batalla = miBatalla
        End Get
        Set(ByVal value As ClaseBatalla)
            miBatalla = value
        End Set
    End Property
    Public Property ConfiguracionEje As ClaseConfiguracionEje
        Get
            ConfiguracionEje = miConfiguracionEje
        End Get
        Set(ByVal value As ClaseConfiguracionEje)
            miConfiguracionEje = value
        End Set
    End Property
    Public Property NumEjesClase As DataTable
        Get
            NumEjesClase = miNumEjesClase
        End Get
        Set(ByVal value As DataTable)
            miNumEjesClase = value
        End Set
    End Property
    Public Property FactCrecimiento As ClaseFactCrecimiento
        Get
            FactCrecimiento = miFactCrecimiento
        End Get
        Set(ByVal value As ClaseFactCrecimiento)
            miFactCrecimiento = value
        End Set
    End Property
End Class

```

Distribución por clase

```
Public Class ClaseDistrClase
    Private miPorcentajes As Double()
    Public Property Porcentajes() As Double()

        Get
            Porcentajes = miPorcentajes
        End Get
        Set(ByVal value As Double())
            miPorcentajes = value
        End Set
    End Property
End Class
```

Distribución horaria

```
Public Class ClaseDistrHoraria
    Private miPorcentajes As Double()
    Public Property Porcentajes() As Double()
        Get
            Porcentajes = miPorcentajes
        End Get
        Set(ByVal value As Double())
            miPorcentajes = value
        End Set
    End Property
End Class
```

Factores de ajuste mensuales

```
Public Class ClaseFactMensuales
    Private miFactores As DataTable
    Public Property Factores As DataTable
        Get
            Factores = miFactores
        End Get
        Set(ByVal value As DataTable)
            miFactores = value
        End Set
    End Property
End Class
```

Espectro de cargas

```
Public Class ClaseEspectroCarga
    Private miTipoEspectroSimple As String
    Private miTipoEspectroTandem As String
    Private miTipoEspectroTridem As String
    Private miEspectroSimpleGeneral As DataTable
    Private miEspectroTandemGeneral As DataTable
    Private miEspectroTridemGeneral As DataTable
    Private miEspectroSimplePorClase As DataTable
    Private miEspectroTandemPorClase As DataTable
```

```

Private miEspectroTridemPorClase As DataTable
Public Property TipoEspectroSimple As String
    Get
        TipoEspectroSimple = miTipoEspectroSimple
    End Get

    Set(ByVal Value As String)
        miTipoEspectroSimple = Value
    End Set
End Property
Public Property TipoEspectroTandem As String
    Get
        TipoEspectroTandem = miTipoEspectroTandem
    End Get
    Set(ByVal Value As String)
        miTipoEspectroTandem = Value
    End Set
End Property
Public Property TipoEspectroTridem As String
    Get
        TipoEspectroTridem = miTipoEspectroTridem
    End Get
    Set(ByVal Value As String)
        miTipoEspectroTridem = Value
    End Set
End Property
Public Property EspectroSimplePorClase As DataTable
    Get
        EspectroSimplePorClase = miEspectroSimplePorClase
    End Get
    Set(ByVal value As DataTable)
        miEspectroSimplePorClase = value
    End Set
End Property
Public Property EspectroTandemPorClase As DataTable
    Get
        EspectroTandemPorClase = miEspectroTandemPorClase
    End Get
    Set(ByVal value As DataTable)
        miEspectroTandemPorClase = value
    End Set
End Property
Public Property EspectroTridemPorClase As DataTable
    Get
        EspectroTridemPorClase = miEspectroTridemPorClase
    End Get
    Set(ByVal value As DataTable)
        miEspectroTridemPorClase = value
    End Set
End Property
Public Property EspectroSimpleGeneral As DataTable
    Get
        EspectroSimpleGeneral = miEspectroSimpleGeneral

```

```

    End Get
    Set(ByVal value As DataTable)
        miEspectroSimpleGeneral = value
    End Set
End Property
Public Property EspectroTandemGeneral As DataTable

    Get
        EspectroTandemGeneral = miEspectroTandemGeneral
    End Get
    Set(ByVal value As DataTable)
        miEspectroTandemGeneral = value
    End Set
End Property
Public Property EspectroTridemGeneral As DataTable
    Get
        EspectroTridemGeneral = miEspectroTridemGeneral
    End Get
    Set(ByVal value As DataTable)
        miEspectroTridemGeneral = value
    End Set
End Property
End Class

```

Batalla

```

Public Class ClaseBatalla
    Private miMediaEspaciamento As Double()
    Private miPorcentajeCamiones As Double()
    Public Property MediaEspaciamento() As Double()
        Get
            MediaEspaciamento = miMediaEspaciamento
        End Get
        Set(ByVal value As Double())
            miMediaEspaciamento = value
        End Set
    End Property
    Public Property PorcentajeCamiones() As Double()
        Get
            PorcentajeCamiones = miPorcentajeCamiones
        End Get
        Set(ByVal value As Double())
            miPorcentajeCamiones = value
        End Set
    End Property
End Class

```

Configuración de ejes

```

Public Class ClaseConfiguracionEje
    Private miAnchoEje As Double
    Private miDistGemelado As Double
    Private miPresionInflado As Double
    Private miEspaciamentoTandem As Double
    Private miEspaciamentoTridem As Double

```

```

Public Property AnchoEje As Double
    Get
        AnchoEje = miAnchoEje
    End Get
    Set(ByVal value As Double)
        miAnchoEje = value
    End Set
End Property
Public Property DistGemelado As Double
    Get
        DistGemelado = miDistGemelado
    End Get
    Set(ByVal value As Double)
        miDistGemelado = value
    End Set
End Property
Public Property PresionInflado As Double
    Get
        PresionInflado = miPresionInflado
    End Get
    Set(ByVal value As Double)
        miPresionInflado = value
    End Set
End Property
Public Property EspaciamientoTandem As Double
    Get
        EspaciamientoTandem = miEspaciamientoTandem
    End Get
    Set(ByVal value As Double)
        miEspaciamientoTandem = value
    End Set
End Property
Public Property EspaciamientoTridem As Double
    Get
        EspaciamientoTridem = miEspaciamientoTridem
    End Get
    Set(ByVal value As Double)
        miEspaciamientoTridem = value
    End Set
End Property
End Class

```

Factores de crecimiento

```

Public Class ClaseFactCrecimiento
    Private miTipoCrecimiento As String
    Private miUnicoValor As Boolean
    Private miFactorPorClase As Double()
    Private miFactorGeneral As Double
    Public Property TipoCrecimiento As String
        Get
            TipoCrecimiento = miTipoCrecimiento
        End Get
        Set(ByVal value As String)
            miTipoCrecimiento = value
        End Set
    End Property
End Class

```

```

End Property
Public Property UnicoValor As Boolean
    Get
        UnicoValor = miUnicoValor
    End Get
    Set(ByVal value As Boolean)
        miUnicoValor = value
    End Set
End Property
Public Property FactorPorClase As Double()
    Get
        FactorPorClase = miFactorPorClase
    End Get
    Set(ByVal value As Double())
        miFactorPorClase = value
    End Set
End Property
Public Property FactorGeneral As Double
    Get
        FactorGeneral = miFactorGeneral
    End Get
    Set(ByVal value As Double)
        miFactorGeneral = value
    End Set
End Property
End Class

```

Anexo C.3 Información estructural del pavimento

Anexo C.3.1 Formularios

Información de estructura

```

Public Class Estructura
    Public InformacionEstructura As New ClaseEstructura
    Private Sub Estructura_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        With dgvCapas
            .Columns.Add("Numero", "No.")
            Dim cmbTipo As New DataGridViewComboBoxColumn()
            cmbTipo.HeaderText = "Tipo"
            cmbTipo.Name = "Tipo"
            cmbTipo.MaxDropDownItems = 4
            cmbTipo.Items.Add("Capa asfáltica")
            cmbTipo.Items.Add("Losa de concreto")
            cmbTipo.Items.Add("Base")
            cmbTipo.Items.Add("Sub-base")
            cmbTipo.Items.Add("Sub-rasante")
            .Columns.Add(cmbTipo)
            .Columns.Add("Espesor", "Espesor [cm]")
            If glInformacionGeneral.TipoPavimento = "Flexible" Then
                .Rows.Add(1, "Capa asfáltica", 10.0)
                .Rows.Add(2, "Base", 20.0)
                .Rows.Add(3, "Sub-base", 30.0)
                .Rows.Add(4, "Sub-rasante", "Semi-infinita")
            End If
        End With
    End Sub
End Class

```

```

ElseIf glInformacionGeneral.TipoPavimento = "Rígido" Then
    .Rows.Add(1, "Los de concreto", 20.0)
    .Rows.Add(2, "Base", 20.0)
    .Rows.Add(3, "Sub-rasante", "Semi-infinita")
Else
    .Rows.Add(1, "Capa asfáltica", 10.0)
    .Rows.Add(2, "Base", 20.0)
    .Rows.Add(3, "Sub-base", 30.0)
    .Rows.Add(4, "Sub-rasante", "Semi-infinita")
End If
Dim row As DataGridViewRow
For Each row In dgvCapas.Rows
    row.Height = 20
Next
Dim fuenteTitulos As New Font(.DefaultCellStyle.Font.FontFamily, 7,
FontStyle.Bold)
.ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
.ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
.ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
.Columns(0).DefaultCellStyle = .ColumnHeadersDefaultCellStyle
.DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
.RowHeadersVisible = False
.Columns("Numero").ReadOnly = True
.Columns("Numero").Width = 60
.Columns("Tipo").Width = 120
.Columns("Espesor").Width = 70
.AllowUserToResizeColumns = False
.AllowUserToResizeRows = False
.AllowUserToAddRows = False
.AllowUserToDeleteRows = False
.AllowUserToOrderColumns = False
End With
End Sub
'Agregar/quitar capas
Private Sub btnAgregar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAgregar.Click
    dgvCapas.Rows.Insert(dgvCapas.CurrentCell.RowIndex, 1)
    Dim row As DataGridViewRow
    For Each row In dgvCapas.Rows
        row.Height = 20
        row.Cells(0).Value = CStr(row.Index + 1)
    Next
End Sub
Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
    dgvCapas.Rows.RemoveAt(dgvCapas.CurrentCell.RowIndex)
    Dim row As DataGridViewRow
    For Each row In dgvCapas.Rows
        row.Cells(0).Value = CStr(row.Index + 1)
    Next
End Sub
Private tipoActual As String
Private Sub btnEditar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEditar.Click

```

```

tipoActual = dgvCapas.CurrentRow.Cells(1).Value
Select Case tipoActual
    Case "Capa asfáltica"
        CapaAsfaltica.Show()
    Case "Losa de concreto"
        CapaRigida.Show()
    Case "Base"
        CapaBase.Show()
    Case "Sub-base"
        CapaSubBase.Show()
    Case "Sub-rasante"
        CapaSubRasante.Show()
    Case ""
        MsgBox("Seleccione el tipo de capa para poder ingresar las
propiedades", vbInformation, "Tipo de capa no válido")
End Select
End Sub
Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub
Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    Me.Close()
End Sub
End Class

```

Capa asfáltica

```

Public Class CapaAsfaltica
    'Ocultar/Mostrar groupboxes para obtener información de la MEZCLA
    Private Sub rbtnEnsayoMezcla_CheckedChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles rbtnEnsayoMezcla.CheckedChanged
        If rbtnEnsayoMezcla.Checked = True Then
            grpGradacion.Visible = False
            grpModuloDinamico.Visible = True
            btnGuardarEnsayoMezcla.Enabled = True
            btnCargarEnsayoMezcla.Enabled = True
        End If
    End Sub
    Private Sub rbtnEstandarMezcla_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles rbtnEstandarMezcla.CheckedChanged
        If rbtnEstandarMezcla.Checked = True Then
            grpModuloDinamico.Visible = False
            grpGradacion.Visible = True
            btnGuardarEnsayoMezcla.Enabled = False
            btnCargarEnsayoMezcla.Enabled = False
        End If
    End Sub
    'Ocultar/Mostrar opciones secundarias LIGANTE
    Private Sub rbtnEnsayosLigante_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles rbtnEnsayosLigante.CheckedChanged
        If rbtnEnsayosLigante.Checked = True Then
            grpEstandarLigante.Visible = False
            grpEnsayosLigante.Visible = True
            rbtnEnsayosConvencionales.Checked = False
        End If
    End Sub

```

```

        btnEstandarPenetracion.Checked = False
        btnEstandarSuperpave.Checked = False
        btnEnsayoSuperpave.Checked = True
        btnCargarEnsayoSuperpave.Enabled = False
        btnGuardarEnsayoSuperpave.Enabled = False
    End If
End Sub
Private Sub rbtnEstandarLigante_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles rbtnEstandarLigante.CheckedChanged
    If rbtnEstandarLigante.Checked = True Then
        grpEnsayosLigante.Visible = False
        grpEstandarLigante.Visible = True
        rbtnEnsayosConvencionales.Checked = False
        rbtnEnsayoSuperpave.Checked = False
        rbtnEstandarPenetracion.Checked = False
        rbtnEstandarSuperpave.Checked = True
        btnCargarEnsayoSuperpave.Enabled = True
        btnGuardarEnsayoSuperpave.Enabled = True
    End If
End Sub
Private Sub rbtnEnsayoSuperpave_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles rbtnEnsayoSuperpave.CheckedChanged
    If rbtnEnsayoSuperpave.Checked = True Then
        grpEnsayoSuperpave.Visible = True
        grpEnsayosConvencionales.Visible = False
        grpEstandarSuperpave.Visible = False
        grpEstandarPenetracion.Visible = False
    End If
End Sub
Private Sub rbtnEnsayosConvencionales_CheckedChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
rbtnEnsayosConvencionales.CheckedChanged
    If rbtnEnsayosConvencionales.Checked = True Then
        grpEnsayoSuperpave.Visible = False
        grpEnsayosConvencionales.Visible = True
        grpEstandarSuperpave.Visible = False
        grpEstandarPenetracion.Visible = False
    End If
End Sub
Private Sub rbtnEstandarSuperpave_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles rbtnEstandarSuperpave.CheckedChanged
    If rbtnEstandarSuperpave.Checked = True Then
        grpEnsayoSuperpave.Visible = False
        grpEnsayosConvencionales.Visible = False
        grpEstandarSuperpave.Visible = True
        grpEstandarPenetracion.Visible = False
    End If
End Sub
Private Sub rbtnEstandarPenetracion_CheckedChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
rbtnEstandarPenetracion.CheckedChanged
    If rbtnEstandarPenetracion.Checked = True Then
        grpEnsayoSuperpave.Visible = False
        grpEnsayosConvencionales.Visible = False
        grpEstandarSuperpave.Visible = False
    End If
End Sub

```

```

        grpEstandarPenetracion.Visible = True
    End If
End Sub
'Poblar los datagridview, entre otras cosas
Private Sub CapaAsfaltica_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    'Labels superpave estandar
    lblAltaTemp.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblAltaTemp46.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblAltaTemp52.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblAltaTemp58.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblAltaTemp64.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblAltaTemp70.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblAltaTemp76.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblAltaTemp82.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblBajaTemp.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblBajaTemp10.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblBajaTemp16.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblBajaTemp22.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblBajaTemp28.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblBajaTemp34.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblBajaTemp40.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    lblBajaTemp46.BackColor =
dgvFrecModuloDinamico.ColumnHeadersDefaultCellStyle.BackColor
    'Datagridview módulo dinámico
    With dgvTempModuloDinamico
        .RowHeadersVisible = False
        .Columns.Add("Temp", "Temp")
        .Columns("Temp").Width = 68
        .Rows.Add(5)
        .Rows(0).Cells(0).Value = "[°C]"
        .Rows(0).DefaultCellStyle = .ColumnHeadersDefaultCellStyle
        .Rows(0).ReadOnly = True
        .Rows(1).Cells(0).Value = 20
        .Rows(2).Cells(0).Value = 30
        .Rows(3).Cells(0).Value = 40
        .Rows(4).Cells(0).Value = 50
        .MultiSelect = False
        lblTituloTemp.BackColor = .ColumnHeadersDefaultCellStyle.BackColor
    End With
    Dim rows1 As DataGridViewRow

```

```

        For Each rows1 In .Rows
            rows1.Height = 20
        Next
        For i = 0 To .Columns.Count - 1
            .Columns.Item(i).SortMode = DataGridViewColumnSortMode.NotSortable
        Next i
        Dim fuenteTitulos As New Font(.DefaultCellStyle.Font.FontFamily, 7,
FontStyle.Bold)
        .ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
        .ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
        .ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
        .EditMode = DataGridViewEditMode.EditOnEnter
        .DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
        .AllowUserToResizeColumns = False
        .AllowUserToResizeRows = False
        .AllowUserToAddRows = False
        .AllowUserToDeleteRows = False
        .AllowUserToOrderColumns = False
    End With
    With dgvFrecModuloDinamico
        .RowHeadersVisible = False
        .Columns.Add("Frec1", "")
        .Columns("Frec1").Width = 80
        .Columns.Add("Frec2", "Frec")
        .Columns("Frec2").Width = 80
        .Columns.Add("Frec3", "")
        .Columns("Frec3").Width = 80
        lblTituloFrecuencia.BackColor = .ColumnHeadersDefaultCellStyle.BackColor
        .Rows.Add(5)
        .Rows(0).DefaultCellStyle = .ColumnHeadersDefaultCellStyle
        .Rows(0).Cells(0).Value = 0.1
        .Rows(0).Cells(1).Value = 1
        .Rows(0).Cells(2).Value = 10
        .MultiSelect = False
    Dim rows2 As DataGridViewRow
    For Each rows2 In .Rows
        rows2.Height = 20
    Next
    For i = 0 To .Columns.Count - 1
        .Columns.Item(i).SortMode = DataGridViewColumnSortMode.NotSortable
    Next i
    Dim fuenteTitulos As New Font(.DefaultCellStyle.Font.FontFamily, 7,
FontStyle.Bold)
        .ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
        .ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
        .ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
        .EditMode = DataGridViewEditMode.EditOnEnter
        .DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
        .AllowUserToResizeColumns = False
        .AllowUserToResizeRows = False
        .AllowUserToAddRows = False
        .AllowUserToDeleteRows = False
        .AllowUserToOrderColumns = False

```

```

End With
'Datagridview ensayo superpave
With dgvSuperpave
    .RowHeadersVisible = False
    .Columns.Add("Temp", "Temp [°C]")
    .Columns("Temp").Width = 80
    .Columns.Add("Modulo", "G* [MPa]")
    .Columns("Modulo").Width = 80
    .Columns.Add("Delta", "Delta [°]")
    .Columns("Delta").Width = 80
    .Rows.Add(3)
    .MultiSelect = False
Dim rows3 As DataGridViewRow
For Each rows3 In .Rows
    rows3.Height = 20
Next
For i = 0 To .Columns.Count - 1
    .Columns.Item(i).SortMode = DataGridViewColumnSortMode.NotSortable
Next i
Dim fuenteTitulos As New Font(.DefaultCellStyle.Font.FontFamily, 7,
FontStyle.Bold)
    .ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
    .ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
    .ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
    .EditMode = DataGridViewEditMode.EditOnEnter
    .DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
    .AllowUserToResizeColumns = False
    .AllowUserToResizeRows = False
    .AllowUserToAddRows = False
    .AllowUserToDeleteRows = False
    .AllowUserToOrderColumns = False
End With
'Datagridview ensayos penetración
With dgvEnsayosPenetracion
    .RowHeadersVisible = False
    .Columns.Add("Temp", "Temp [°C]")
    .Columns("Temp").Width = 74
    .Columns.Add("Penetracion", "Penetración [mm/10]")
    .Columns("Penetracion").Width = 80
    .Rows.Add(3)
    .MultiSelect = False
Dim rows4 As DataGridViewRow
For Each rows4 In .Rows
    rows4.Height = 20
Next
For i = 0 To .Columns.Count - 1
    .Columns.Item(i).SortMode = DataGridViewColumnSortMode.NotSortable
Next i
Dim fuenteTitulos As New Font(.DefaultCellStyle.Font.FontFamily, 7,
FontStyle.Bold)
    .ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
    .ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
    .ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken

```

```

        .EditMode = DataGridViewEditMode.EditOnEnter
        .DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
        .AllowUserToResizeColumns = False
        .AllowUserToResizeRows = False
        .AllowUserToAddRows = False
        .AllowUserToDeleteRows = False
        .AllowUserToOrderColumns = False
    End With
End Sub
'Limpiar selección al salir datagridview
Private Sub dgvTempModuloDinamico_Leave(ByVal sender As Object, ByVal e As
System.EventArgs) Handles dgvTempModuloDinamico.Leave
    dgvTempModuloDinamico.ClearSelection()
End Sub
Private Sub dgvFrecModuloDinamico_Leave(ByVal sender As Object, ByVal e As
System.EventArgs) Handles dgvFrecModuloDinamico.Leave
    dgvFrecModuloDinamico.ClearSelection()
End Sub
'Agregar/quitar temperaturas módulo dinámico
Private Sub btnAgregarTemp_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAgregarTemp.Click
    dgvTempModuloDinamico.Rows.Add()
    dgvTempModuloDinamico.Rows(dgvTempModuloDinamico.RowCount - 1).Height = 20
    dgvFrecModuloDinamico.Rows.Add()
    dgvFrecModuloDinamico.Rows(dgvFrecModuloDinamico.RowCount - 1).Height = 20
End Sub
Private Sub btnQuitarTemp_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnQuitarTemp.Click
    dgvTempModuloDinamico.Rows.RemoveAt(dgvTempModuloDinamico.RowCount - 1)
    dgvFrecModuloDinamico.Rows.RemoveAt(dgvTempModuloDinamico.RowCount - 1)
End Sub
'Agregar/quitar frecuencias módulo dinámico
Private Sub btnAgregarFrec_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAgregarFrec.Click
    dgvFrecModuloDinamico.Columns.Add("Frec" &
dgvFrecModuloDinamico.ColumnCount.ToString, "")
    dgvFrecModuloDinamico.Columns(dgvFrecModuloDinamico.ColumnCount - 1).Width =
80
    dgvFrecModuloDinamico.Columns.Item(dgvFrecModuloDinamico.ColumnCount -
1).SortMode = DataGridViewColumnSortMode.NotSortable
End Sub
Private Sub btnQuitarFrec_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnQuitarFrec.Click
    dgvFrecModuloDinamico.Columns.Remove(dgvFrecModuloDinamico.Columns(dgvFrecModuloDina
mico.ColumnCount - 1))
End Sub
'Agregar/quitar temperaturas superpave
Private Sub btnAgregarTempSuperpave_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnAgregarTempSuperpave.Click
    dgvSuperpave.Rows.Add()
    dgvSuperpave.Rows(dgvSuperpave.RowCount - 1).Height = 20
End Sub
Private Sub btnQuitarTempSuperpave_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnQuitarTempSuperpave.Click

```

```

        dgvSuperpave.Rows.RemoveAt(dgvSuperpave.RowCount - 1)
    End Sub
    'Agregar/quitar ensayo penetración
    Private Sub btnAgregarPenetracion_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnAgregarPenetracion.Click
        dgvEnsayosPenetracion.Rows.Add()
        dgvEnsayosPenetracion.Rows(dgvEnsayosPenetracion.RowCount - 1).Height = 20
    End Sub
    Private Sub btnQuitarPenetracion_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnQuitarPenetracion.Click
        dgvEnsayosPenetracion.Rows.RemoveAt(dgvEnsayosPenetracion.RowCount - 1)
    End Sub
    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub
    Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
        Me.Close()
    End Sub
End Class

```

Capa de pavimento rígido

```

Public Class CapaRigida
    'JPCP o CRPC
    Private Sub rbtnJPCP_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnJPCP.CheckedChanged
        txtDias50RetraccionUltima.Enabled = True
    End Sub
    Private Sub rbtnCRPC_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnCRPC.CheckedChanged
        txtDias50RetraccionUltima.Enabled = False
    End Sub
    'Ocultar/mostrar groupboxes de Resistencia según jerarquia
    Private Sub rbtnResistencia3_CheckedChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles rbtnResistencia3.CheckedChanged
        grpResistenciaNivel1.Visible = False
        grpResistenciaNivel2.Visible = False
        grpResistenciaNivel3.Visible = True
        btnCargarEnsayosRigido.Enabled = False
        btnGuardarEnsayosRigido.Enabled = False
    End Sub
    Private Sub rbtnResistencia2_CheckedChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles rbtnResistencia2.CheckedChanged
        grpResistenciaNivel1.Visible = False
        grpResistenciaNivel2.Visible = True
        grpResistenciaNivel3.Visible = False
        btnCargarEnsayosRigido.Enabled = True
        btnGuardarEnsayosRigido.Enabled = True
    End Sub
    Private Sub rbtnResistencia1_CheckedChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles rbtnResistencia1.CheckedChanged
        grpResistenciaNivel1.Visible = True
        grpResistenciaNivel2.Visible = False
        grpResistenciaNivel3.Visible = False
    End Sub

```

```

        btnCargarEnsayosRigido.Enabled = True
        btnGuardarEnsayosRigido.Enabled = True
    End Sub
    'CAMBIOS EN EL FORM DEPENDIENDO DE NIVEL JERÁRQUICO
    'Información Nivel 3
    Private Sub rbtnOpcionMr_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles rbtnOpcionMr.CheckedChanged
        txtMr28dias.Enabled = True
        txtfc28dias.Enabled = False
    End Sub
    Private Sub rbtnOpcionfc_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles rbtnOpcionfc.CheckedChanged
        txtMr28dias.Enabled = False
        txtfc28dias.Enabled = True
    End Sub
    Private Sub checkE_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles checkE.CheckedChanged
        CambioEnCheckboxE()
    End Sub
    Private Sub CambioEnCheckboxE()
        If checkE.Checked = True Then
            txtE28dias.Enabled = True
        Else
            txtE28dias.Enabled = False
        End If
    End Sub
    'Información Nivel 2
    Private Sub check90diasNivel2_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles check90diasNivel2.CheckedChanged
        CambionEnCheckbox90DiasNivel2()
    End Sub
    Private Sub CambionEnCheckbox90DiasNivel2()
        If check90diasNivel2.Checked = True Then
            txtfc90.Enabled = True
        Else
            txtfc90.Enabled = False
        End If
    End Sub
    'Información Nivel 1
    Private Sub check90diasNivel1_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles check90diasNivel1.CheckedChanged
        CambionEnCheckbox90DiasNivel1()
    End Sub
    Private Sub CambionEnCheckbox90DiasNivel1()
        If check90diasNivel1.Checked = True Then
            txtE90.Enabled = True
            txtMr90.Enabled = True
        Else
            txtE90.Enabled = False
            txtMr90.Enabled = False
        End If
    End Sub
    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click, btnGuardarEnsayosRigido.Click
        Me.Close()
    End Sub

```

```

End Sub
End Class

```

Capas granulares

```

Public Class CapaBase
    'Poblar datagridview granulometria
    Private Sub CapaBase_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        With dgvGranulometria
            .Columns.Add("LimInf", "Límite Inferior")
            .Columns.Add("LimSup", "Límite Superior")
            .RowCount = 18
            .TopLeftHeaderCell.Value = "Tamiz"
            .Rows(0).HeaderCell.Value = "#200"
            .Rows(1).HeaderCell.Value = "#100"
            .Rows(2).HeaderCell.Value = "#80"
            .Rows(3).HeaderCell.Value = "#60"
            .Rows(4).HeaderCell.Value = "#50"
            .Rows(5).HeaderCell.Value = "#40"
            .Rows(6).HeaderCell.Value = "#30"
            .Rows(7).HeaderCell.Value = "#20"
            .Rows(8).HeaderCell.Value = "#16"
            .Rows(9).HeaderCell.Value = "#10"
            .Rows(10).HeaderCell.Value = "#8"
            .Rows(11).HeaderCell.Value = "#4"
            .Rows(12).HeaderCell.Value = "#3/8'"
            .Rows(13).HeaderCell.Value = "#1/2'"
            .Rows(14).HeaderCell.Value = "#3/4'"
            .Rows(15).HeaderCell.Value = "#1'"
            .Rows(16).HeaderCell.Value = "#1 1/2'"
            .Rows(17).HeaderCell.Value = "#2'"
            .Columns(0).Width = 60
            .Columns(1).Width = 60
            .RowHeadersWidth = 62
            For i = 0 To .Columns.Count - 1
                .Columns.Item(i).SortMode = DataGridViewColumnSortMode.Programmatic
            Next i
            Dim fuenteTitulos As New Font(.DefaultCellStyle.Font.FontFamily, 6.5,
FontStyle.Bold)
            .ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
            .ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
            .RowHeadersDefaultCellStyle.Font = fuenteTitulos
            .RowHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleLeft
            Dim fuente As New Font(.DefaultCellStyle.Font.FontFamily, 6.5)
            .RowsDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
            .RowsDefaultCellStyle.Font = fuente
            Dim row As DataGridViewRow
            For Each row In dgvGranulometria.Rows
                row.Height = 17
            Next

```

```

        .ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Raised
        .SelectionMode = DataGridViewSelectionMode.CellSelect
    End With
End Sub

'Metodo clasificación suelos
Private Sub rbtnAASHTO_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnAASHTO.CheckedChanged
    If rbtnAASHTO.Checked = True Then
        cmbMaterial.Items.Clear()
        cmbMaterial.Items.AddRange({"A 1-a", "A 1-b", "A 2-4", "A 2-5", "A 2-6",
"A 2-7", "A 3", "A 4", "A 5", "A 6", "A 7-5", "A 7-6"})
        cmbMaterial.Text = "A 1-a"
    End If
End Sub
Private Sub rbtnUCS_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnUCS.CheckedChanged
    If rbtnUCS.Checked = True Then
        cmbMaterial.Items.Clear()
        cmbMaterial.Items.AddRange({"CH", "MH", "CL", "ML", "SW", "SP", "SW-SC",
"SW-SM", "SP-SC", "SP-SM", "SC", "SM", "GW", "GP", "GW-GC", "GW-GM", "GP-GC", "GP-
GM", "GC", "GM"})
        cmbMaterial.Text = "CH"
    End If
End Sub
'Miveles de Mr
Private Sub rbtnNivel1_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnNivel1.CheckedChanged
    If rbtnNivel1.Checked = True Then
        grpMrNivel1.Visible = True
        grpMrNivel2.Visible = False
        grpMrNivel3.Visible = False
    End If
End Sub
Private Sub rbtnNivel2_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnNivel2.CheckedChanged
    If rbtnNivel2.Checked = True Then
        grpMrNivel1.Visible = False
        grpMrNivel2.Visible = True
        grpMrNivel3.Visible = True
        txtMr.Enabled = False
    End If
End Sub
Private Sub rbtnNivel3_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnNivel3.CheckedChanged
    If rbtnNivel3.Checked = True Then
        grpMrNivel1.Visible = False
        grpMrNivel2.Visible = False
        grpMrNivel3.Visible = True
        txtMr.Enabled = True
    End If
End Sub
'Calcular Mr con granulometría
Private Sub checkMrGranulometría_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles checkMrGranulometría.CheckedChanged

```

```

If checkMrGranulometría.Checked = True Then
    grpMrNivel2.Enabled = False
    grpMrNivel3.Enabled = False
    grpMrNiveles.Enabled = False
    txtMr.Enabled = False
Else
    grpMrNivel2.Enabled = True
    grpMrNivel3.Enabled = True
    grpMrNiveles.Enabled = True
    txtMr.Enabled = True
End If
End Sub
'Datos de ensayo triaxial
Private Sub checkEnsayoTriaxial_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles checkEnsayoTriaxial.CheckedChanged
    If checkEnsayoTriaxial.Checked = True Then
        txtk1.Enabled = False
        txtk2.Enabled = False
    Else
        txtk1.Enabled = True
        txtk2.Enabled = True
    End If
End Sub
'Tipo de información de nivel 2
Private Sub rbtnCBR_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnCBR.CheckedChanged
    If rbtnCBR.Checked = True Then
        txtCBR.Enabled = True
        txtValorR.Enabled = False
        txtAi.Enabled = False
        txtPenetrometro.Enabled = False
    End If
End Sub
Private Sub rbtnValorR_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnValorR.CheckedChanged
    If rbtnValorR.Checked = True Then
        txtCBR.Enabled = False
        txtValorR.Enabled = True
        txtAi.Enabled = False
        txtPenetrometro.Enabled = False
    End If
End Sub
Private Sub rbtnAi_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtnAi.CheckedChanged
    If rbtnAi.Checked = True Then
        txtCBR.Enabled = False
        txtValorR.Enabled = False
        txtAi.Enabled = True
        txtPenetrometro.Enabled = False
    End If
End Sub
Private Sub rbtnPenetrometro_CheckedChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles rbtnPenetrometro.CheckedChanged
    If rbtnPenetrometro.Checked = True Then
        txtCBR.Enabled = False

```

```

        txtValorR.Enabled = False
        txtAi.Enabled = False
        txtPenetrometro.Enabled = True
    End If
End Sub
Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub
Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    Me.Close()
End Sub
Private Sub btnEnsayoTriaxial_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEnsayoTriaxial.Click
    EnsayoTriaxialBase.Show()
End Sub
End Class

```

Ensayo triaxial para capas granulares

```

Public Class EnsayoTriaxialBase
    Private Sub dgvEnsayoTriaxial_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Dim tablaEnsTriaxial As New DataTable
        With tablaEnsTriaxial
            .Columns.Add("PRESIÓN DE CONFINAMIENTO", GetType(String))
            .Columns.Add("ESFUERZO DESVIADOR", GetType(String))
            .Columns.Add("DEFORMACIÓN RECUPERABLE", GetType(String))
            .Rows.Add(20, 1, 0.264)
            .Rows.Add(20, 2, 0.496)
            .Rows.Add(20, 5, 1.184)
            .Rows.Add(20, 10, 2.284)
            .Rows.Add(20, 15, 3.428)
            .Rows.Add(20, 20, 4.42)
            .Rows.Add(15, 1, 0.26)
            .Rows.Add(15, 2, 0.512)
            .Rows.Add(15, 5, 1.3)
            .Rows.Add(15, 10, 2.5)
            .Rows.Add(15, 15, 3.636)
            .Rows.Add(15, 20, 4.572)
            .Rows.Add(10, 1, 0.324)
            .Rows.Add(10, 2, 0.672)
            .Rows.Add(10, 5, 1.74)
            .Rows.Add(10, 10, 3.636)
            .Rows.Add(10, 15, 3.872)
            .Rows.Add(5, 1, 0.508)
            .Rows.Add(5, 2, 0.988)
            .Rows.Add(5, 5, 2.224)
            .Rows.Add(5, 10, 3.884)
            .Rows.Add(5, 15, 5.768)
            .Rows.Add(1, 1, 0.636)
            .Rows.Add(1, 2, 0.88)
            .Rows.Add(1, 5, 2.704)
            .Rows.Add(1, 7.5, 3.26)
            .Rows.Add(1, 10, 4.444)
        End With
    End Sub
End Class

```

```

End With
Dim dsEnsTriaxial As New DataSet
dsEnsTriaxial.Tables.Add(tablaEnsTriaxial)
With dgvEnsayoTriaxial
    dgvEnsayoTriaxial.DataSource = dsEnsTriaxial.Tables(0)
    .Columns("PRESIÓN DE CONFINAMIENTO").Width = 110
    .Columns("ESFUERZO DESVIADOR").Width = 100
    .Columns("DEFORMACIÓN RECUPERABLE").Width = 100
    For i = 0 To .Columns.Count - 1
        .Columns.Item(i).SortMode = DataGridViewColumnSortMode.NotSortable
    Next i
    Dim fuenteTitulos As New Font(.DefaultCellStyle.Font.FontFamily, 6.1,
FontStyle.Bold)
    .ColumnHeadersDefaultCellStyle.Font = fuenteTitulos
    .ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
    .ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.Sunken
    .RowHeadersVisible = False
    Dim row As DataGridViewRow
    For Each row In .Rows
        row.Height = 20
    Next
    .EditMode = DataGridViewEditMode.EditOnEnter
    .DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter
    .AllowUserToResizeColumns = False
    .AllowUserToResizeRows = False
    .AllowUserToAddRows = False
    .AllowUserToDeleteRows = False
    .AllowUserToOrderColumns = False
    .MultiSelect = True
End With
End Sub
Private Sub btnCalcular_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCalcular.Click
    Dim DatosEnsayo As New DataTable
    DatosEnsayo = dgvEnsayoTriaxial.DataSource
    Dim MatrizEnsayo = DataTable2Matriz(DatosEnsayo)
    Dim MatrizResultados(MatrizEnsayo.GetUpperBound(0), 5) As Double
    Dim LongInicial = txtLongInicial.Text
    For i = 0 To MatrizEnsayo.GetUpperBound(0)
        MatrizResultados(i, 0) = MatrizEnsayo(i, 2) / LongInicial
        MatrizResultados(i, 1) = MatrizEnsayo(i, 1) / MatrizResultados(i, 0)
        MatrizResultados(i, 2) = MatrizEnsayo(i, 1) + 3 * MatrizEnsayo(i, 0)
        MatrizResultados(i, 3) = Math.Log10(MatrizResultados(i, 1))
        MatrizResultados(i, 4) = Math.Log10(MatrizResultados(i, 2))
    Next
    Dim Intercepto As Double
    Dim Pendiente As Double
    Dim Sumxy As Double = 0
    Dim Sumx As Double = 0
    Dim Sumy As Double = 0
    Dim Sumx2 As Double = 0
    Dim Promx As Double = 0
    Dim Promy As Double = 0

```

```

'Calculo Intercepto

Dim n = MatrizEnsayo.GetUpperBound(0) + 1
For i = 0 To n - 1
    Sumxy += MatrizResultados(i, 3) * MatrizResultados(i, 4)
    Sumx += MatrizResultados(i, 4)
    Sumy += MatrizResultados(i, 3)
    Sumx2 += Math.Pow(MatrizResultados(i, 4), 2)
Next
Pendiente = (n * Sumxy - Sumx * Sumy) / (n * Sumx2 - Math.Pow(Sumx, 2))
Promx = Sumx / n
Promy = Sumy / n
Intercepto = (Math.Pow(10, Promy - Pendiente * Promx)) * 1000
txtk1.Text = Pendiente
txtk2.Text = Intercepto
End Sub
Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    CapaBase.txtk1.Text = txtk1.Text
    CapaBase.txtk2.Text = txtk2.Text
    Me.Close()
End Sub
Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub
End Class

```

Anexo C.3.2 Clases

Estructura

```

Public Class ClaseEstructura
    Private miAbsorcionOndaCorta As Double
    Private miNumCapas As Integer
    Private miCapa As ClaseCapa
    Public Property AbsorcionOndaCorta As Double
        Get
            AbsorcionOndaCorta = miAbsorcionOndaCorta
        End Get
        Set(ByVal value As Double)
            miAbsorcionOndaCorta = value
        End Set
    End Property
    Public Property NumCapas As Integer
        Get
            NumCapas = miNumCapas
        End Get
        Set(ByVal value As Integer)
            miNumCapas = value
        End Set
    End Property
    Public Property Capa As ClaseCapa
        Get
            Capa = miCapa
        End Get

```

```

        Set(ByVal value As ClaseCapa)
            miCapa = value
        End Set
    End Property
End Class

```

Capa estructural

```

Public Class ClaseCapa
    Private miNumCapa As Integer
    Private miTipoCapa As String
    Private miEspesorCapa As Double
    Private miAsfalto As ClaseAsfalto
    Private miRigido As ClaseRigido
    Private miBase As ClaseGranular
    Private miSubBase As ClaseGranular
    Private miSubRasante As ClaseGranular
    Public Property NumCapa As Integer
        Get
            NumCapa = miNumCapa
        End Get
        Set(ByVal value As Integer)
            miNumCapa = value
        End Set
    End Property
    Public Property TipoCapa As String
        Get
            TipoCapa = miTipoCapa
        End Get
        Set(ByVal value As String)
            miTipoCapa = value
        End Set
    End Property
    Public Property EspesorCapa As Double
        Get
            EspesorCapa = miEspesorCapa
        End Get
        Set(ByVal value As Double)
            miEspesorCapa = value
        End Set
    End Property
    Public Property Asfalto As ClaseAsfalto
        Get
            Asfalto = miAsfalto
        End Get
        Set(ByVal value As ClaseAsfalto)
            miAsfalto = value
        End Set
    End Property
    Public Property Rigido As ClaseRigido
        Get
            Rigido = miRigido
        End Get
        Set(ByVal value As ClaseRigido)
            miRigido = value
        End Set
    End Property

```

```

End Property
Public Property Base As ClaseGranular
    Get
        Base = miBase
    End Get
    Set(ByVal value As ClaseGranular)
        miBase = value
    End Set
End Property
Public Property SubBase As ClaseGranular
    Get
        SubBase = miSubBase
    End Get
    Set(ByVal value As ClaseGranular)
        miSubBase = value
    End Set
End Property
Public Property SubRasante As ClaseGranular
    Get
        SubRasante = miSubRasante
    End Get
    Set(ByVal value As ClaseGranular)
        miSubRasante = value
    End Set
End Property
End Class

```

Capa asfáltica

```

Public Class ClaseAsfalto
    Private miMezcla As ClaseMezcla
    Private miLigante As ClaseLigante
    Private miGeneral As ClaseAsfaltoGeneral
    Public Property Mezcla As ClaseMezcla
        Get
            Mezcla = miMezcla
        End Get
        Set(ByVal value As ClaseMezcla)
            miMezcla = value
        End Set
    End Property
    Public Property Ligante As ClaseLigante
        Get
            Ligante = miLigante
        End Get
        Set(ByVal value As ClaseLigante)
            miLigante = value
        End Set
    End Property
    Public Property General As ClaseAsfaltoGeneral
        Get
            General = miGeneral
        End Get
        Set(ByVal value As ClaseAsfaltoGeneral)
            miGeneral = value
        End Set
    End Property

```

```
End Property
End Class
```

Información general de capa asfáltica

```
Public Class ClaseAsfaltoGeneral
    Private miTempReferencia As Double
    Private miContEfectivoLigante As Double
    Private miPorcentajeVacios As Double
    Private miDensidad As Double
    Private miPoisson As Double
    Private miConductividadTermica As Double
    Private miCapacidadCalorica As Double
    Public Property TempReferencia As Double
        Get
            TempReferencia = miTempReferencia
        End Get
        Set(ByVal value As Double)
            miTempReferencia = value
        End Set
    End Property
    Public Property ContEfectivoLigante As Double
        Get
            ContEfectivoLigante = miContEfectivoLigante
        End Get
        Set(ByVal value As Double)
            miContEfectivoLigante = value
        End Set
    End Property
    Public Property PorcentajeVacios As Double
        Get
            PorcentajeVacios = miPorcentajeVacios
        End Get
        Set(ByVal value As Double)
            miPorcentajeVacios = value
        End Set
    End Property
    Public Property Densidad As Double
        Get
            Densidad = miDensidad
        End Get
        Set(ByVal value As Double)
            miDensidad = value
        End Set
    End Property
    Public Property Poisson As Double
        Get
            Poisson = miPoisson
        End Get
        Set(ByVal value As Double)
            miPoisson = value
        End Set
    End Property
    Public Property ConductividadTermica As Double
        Get
```

```

        ConductividadTermica = miConductividadTermica
    End Get

    Set(ByVal value As Double)
        miConductividadTermica = value
    End Set
End Property
Public Property CapacidadCalorica As Double
    Get
        CapacidadCalorica = miCapacidadCalorica
    End Get
    Set(ByVal value As Double)
        miCapacidadCalorica = value
    End Set
End Property
End Class

```

Información de la mezcla asfáltica

```

Public Class ClaseMezcla
    Private miTipo As String
    Private miEnsayosMezcla As ClaseEnsayosMezcla
    Private miAlfa As Double
    Private miBeta As Double
    Private miDelta As Double
    Private miGama As Double
    Private miCe As Double
    Public Property Tipo As String
        Get
            Tipo = miTipo
        End Get
        Set(ByVal value As String)
            miTipo = value
        End Set
    End Property
    Public Property EnsayosMezcla As ClaseEnsayosMezcla
        Get
            EnsayosMezcla = miEnsayosMezcla
        End Get
        Set(ByVal value As ClaseEnsayosMezcla)
            miEnsayosMezcla = value
        End Set
    End Property
    Public Property Alfa As Double
        Get
            Alfa = miAlfa
        End Get
        Set(ByVal value As Double)
            miAlfa = value
        End Set
    End Property
    Public Property Beta As Double
        Get
            Beta = miBeta
        End Get

```

```

        Set(ByVal value As Double)
            miBeta = value
        End Set
    End Property
    Public Property Delta As Double
        Get
            Delta = miDelta
        End Get
        Set(ByVal value As Double)
            miDelta = value
        End Set
    End Property
    Public Property Gama As Double
        Get
            Gama = miGama
        End Get
        Set(ByVal value As Double)
            miGama = value
        End Set
    End Property
    Public Property Ce As Double
        Get
            Ce = miCe
        End Get
        Set(ByVal value As Double)
            miCe = value
        End Set
    End Property
End Class

```

Ensayo de módulo dinámico de la mezcla

```

Public Class ClaseEnsayosMezcla
    Private miEnsayoModuloDinamico As DataTable
    Public Property EnsayoModuloDinamico As DataTable
        Get
            EnsayoModuloDinamico = miEnsayoModuloDinamico
        End Get
        Set(ByVal value As DataTable)
            miEnsayoModuloDinamico = value
        End Set
    End Property
End Class

```

Información estándar de la mezcla

```

Public Class ClaseEstandarMezcla
    Private miRetenido3cuartos As Double
    Private miRetenido3octavos As Double
    Private miRetenidoN4 As Double
    Private miPasa200 As Double
    Public Property Retenido3cuartos As Double
        Get
            Retenido3cuartos = miRetenido3cuartos
        End Get
        Set(ByVal value As Double)

```

```

        miRetenido3cuartos = value
    End Set
End Property

Public Property Retenido3octavos As Double
    Get
        Retenido3octavos = miRetenido3octavos
    End Get
    Set(ByVal value As Double)
        miRetenido3octavos = value
    End Set
End Property
Public Property RetenidoN4 As Double
    Get
        RetenidoN4 = miRetenidoN4
    End Get
    Set(ByVal value As Double)
        miRetenidoN4 = value
    End Set
End Property
Public Property Pasa200 As Double
    Get
        Pasa200 = miPasa200
    End Get
    Set(ByVal value As Double)
        miPasa200 = value
    End Set
End Property
End Class

```

Información del ligante asfáltico

```

Public Class ClaseLigante
    Private miTipoInput As String
    Private miA As Double
    Private miVTS As Double
    Private miEnsayosligante As ClaseEnsayosLigante
    Private miEstandarLigante As ClaseEstandarLigante
    Public Property TipoInput As String
        Get
            TipoInput = miTipoInput
        End Get
        Set(ByVal value As String)
            miTipoInput = value
        End Set
    End Property
    Public Property A As Double
        Get
            A = miA
        End Get
        Set(ByVal value As Double)
            miA = value
        End Set
    End Property
    Public Property VTS As Double
        Get

```

```

        VTS = miVTS
    End Get

    Set(ByVal value As Double)
        miVTS = value
    End Set
End Property
Public Property EnsayosLigante As ClaseEnsayosLigante
    Get
        EnsayosLigante = miEnsayosligante
    End Get
    Set(ByVal value As ClaseEnsayosLigante)
        miEnsayosligante = value
    End Set
End Property
Public Property EstandarLigante As ClaseEstandarLigante
    Get
        EstandarLigante = miEstandarLigante
    End Get
    Set(ByVal value As ClaseEstandarLigante)
        miEstandarLigante = value
    End Set
End Property
End Class

```

Clases de ensayos del ligante

```

Public Class ClaseEnsayosLigante
    Private miTipoEnsayo As String
    Private miEnsayoSuperpave As ClaseEnsayoSuperpaveLigante
    Private miEnsayosConvencionalesLigante As ClaseEnsayosConvencionalesLigante
    Public Property TipoEnsayo As String
        Get
            TipoEnsayo = miTipoEnsayo
        End Get
        Set(ByVal value As String)
            miTipoEnsayo = value
        End Set
    End Property
    Public Property EnsayoSuperpave As ClaseEnsayoSuperpaveLigante
        Get
            EnsayoSuperpave = miEnsayoSuperpave
        End Get
        Set(ByVal value As ClaseEnsayoSuperpaveLigante)
            miEnsayoSuperpave = value
        End Set
    End Property
    Public Property EnsayosConvencionalesLigante As
ClaseEnsayosConvencionalesLigante
        Get
            EnsayosConvencionalesLigante = miEnsayosConvencionalesLigante
        End Get
        Set(ByVal value As ClaseEnsayosConvencionalesLigante)
            miEnsayosConvencionalesLigante = value
        End Set
    End Property

```

```
End Property
End Class
```

Ensayos convencionales del ligante

```
Public Class ClaseEnsayosConvencionalesLigante
Private miPuntoAblandamiento As Double()
Private miViscosidadAbsoluta As Double()
Private miViscosidadCinematica As Double()
Private miGravedadEspecifica As Double()
Private miPenetracion As DataTable
Public Property PuntoAblandamiento() As Double()
Get
PuntoAblandamiento = miPuntoAblandamiento
End Get
Set(ByVal value As Double())
miPuntoAblandamiento = value
End Set
End Property
Public Property ViscosidadAbsoluta() As Double()
Get
ViscosidadAbsoluta = miViscosidadAbsoluta
End Get
Set(ByVal value As Double())
miViscosidadAbsoluta = value
End Set
End Property
Public Property ViscosidadCinematica() As Double()
Get
ViscosidadCinematica = miViscosidadCinematica
End Get
Set(ByVal value As Double())
miViscosidadCinematica = value
End Set
End Property
Public Property GravedadEspecifica() As Double()
Get
GravedadEspecifica = miGravedadEspecifica
End Get
Set(ByVal value As Double())
miGravedadEspecifica = value
End Set
End Property
Public Property Penetracion As DataTable
Get
Penetracion = miPenetracion
End Get
Set(ByVal value As DataTable)
miPenetracion = value
End Set
End Property
End Class
```

Ensayo superpave del ligante

```
Public Class ClaseEnsayoSuperpaveligante
```

```

Private miEnsayoSuperpave As DataTable
Public Property EnsayoSuperpave As DataTable

    Get
        EnsayoSuperpave = miEnsayoSuperpave
    End Get
    Set(ByVal value As DataTable)
        miEnsayoSuperpave = value
    End Set
End Property
End Class

```

Información estándar del ligante

```

Public Class ClaseEstandarLigante
Private miTipoPropiedad As String
Private miSuperpave As ClaseSuperpave
Private miPenetracion As ClasePenetracion
Public Property TipoPropiedad As String
    Get
        TipoPropiedad = miTipoPropiedad
    End Get
    Set(ByVal value As String)
        miTipoPropiedad = value
    End Set
End Property
Public Property Superpave As ClaseSuperpave
    Get
        Superpave = miSuperpave
    End Get
    Set(ByVal value As ClaseSuperpave)
        miSuperpave = value
    End Set
End Property
Public Property Penetracion As ClasePenetracion
    Get
        Penetracion = miPenetracion
    End Get
    Set(ByVal value As ClasePenetracion)
        miPenetracion = value
    End Set
End Property
End Class

```

Información del grado de penetración

```

Public Class ClasePenetracion
Private miGradoPenetracion As String
Public Property GradoPenetracion As String
    Get
        GradoPenetracion = miGradoPenetracion
    End Get
    Set(ByVal value As String)
        miGradoPenetracion = value
    End Set
End Property
End Class

```

Información del grado superpave

```
Public Class ClaseSuperpave
    Private miAltaTemp As Integer
    Private miBajaTemp As Integer
    Public Property AltaTemp As Integer
        Get
            AltaTemp = miAltaTemp
        End Get
        Set(ByVal value As Integer)
            miAltaTemp = value
        End Set
    End Property
    Public Property BajaTemp As Integer
        Get
            BajaTemp = miBajaTemp
        End Get
        Set(ByVal value As Integer)
            miBajaTemp = value
        End Set
    End Property
End Class
```

Capa de pavimento rígido

```
Public Class ClaseRigido
    Private miPesoUnitario As Double
    Private miPoisson As Double
    Private miCoefExpTermica As Double
    Private miConductividadTermica As Double
    Private miCapacidadCalorica As Double
    Private miTipoCemento As String
    Private miContenidoCemento As Double
    Private miRelacionAC As Double
    Private miTipoAgregado As String
    Private miTempCeroEsfuerzo As Double
    Private miResistenciaUltimaRetraccion40hr As Double
    Private miDeformacionReversible As Double
    Private miTiempo50ResistenciaRetraccion As Double
    Private miMetodoCurado As String
    Private miNivelEsfuerzo As Integer
    Private miEnsayosRigido As ClaseEnsayosRigido
    Public Property PesoUnitario As Double
        Get
            PesoUnitario = miPesoUnitario
        End Get
        Set(ByVal value As Double)
            miPesoUnitario = value
        End Set
    End Property
    Public Property Poisson As Double
        Get
            Poisson = miPoisson
        End Get
```

```

        Set(ByVal value As Double)
            miPoisson = value
        End Set
    End Property
    Public Property CoefExpTermica As Double
        Get
            CoefExpTermica = miCoefExpTermica
        End Get
        Set(ByVal value As Double)
            miCoefExpTermica = value
        End Set
    End Property
    Public Property ConductividadTermica As Double
        Get
            ConductividadTermica = miConductividadTermica
        End Get
        Set(ByVal value As Double)
            miConductividadTermica = value
        End Set
    End Property
    Public Property CapacidadCalorica As Double
        Get
            CapacidadCalorica = miCapacidadCalorica
        End Get
        Set(ByVal value As Double)
            miCapacidadCalorica = value
        End Set
    End Property
    Public Property TipoCemento As String
        Get
            TipoCemento = miTipoCemento
        End Get
        Set(ByVal value As String)
            miTipoCemento = value
        End Set
    End Property
    Public Property ContenidoCemento As Double
        Get
            ContenidoCemento = miContenidoCemento
        End Get
        Set(ByVal value As Double)
            miContenidoCemento = value
        End Set
    End Property
    Public Property RelacionAC As Double
        Get
            RelacionAC = miRelacionAC
        End Get
        Set(ByVal value As Double)
            miRelacionAC = value
        End Set
    End Property
    Public Property TipoAgregado As String

```

```

    Get
        TipoAgregado = miTipoAgregado
    End Get
    Set(ByVal value As String)
        miTipoAgregado = value
    End Set
End Property
Public Property TempCeroEsfuerzo As Double
    Get
        TempCeroEsfuerzo = miTempCeroEsfuerzo
    End Get
    Set(ByVal value As Double)
        miTempCeroEsfuerzo = value
    End Set
End Property
Public Property ResistenciaUltimaRetraccion40hr As Double
    Get
        ResistenciaUltimaRetraccion40hr = miResistenciaUltimaRetraccion40hr
    End Get
    Set(ByVal value As Double)
        miResistenciaUltimaRetraccion40hr = value
    End Set
End Property
Public Property DeformacionReversible As Double
    Get
        DeformacionReversible = miDeformacionReversible
    End Get
    Set(ByVal value As Double)
        miDeformacionReversible = value
    End Set
End Property
Public Property Tiempo50ResistenciaRetraccion As Double
    Get
        Tiempo50ResistenciaRetraccion = miTiempo50ResistenciaRetraccion
    End Get
    Set(ByVal value As Double)
        miTiempo50ResistenciaRetraccion = value
    End Set
End Property
Public Property MetodoCurado As String
    Get
        MetodoCurado = miMetodoCurado
    End Get
    Set(ByVal value As String)
        miMetodoCurado = value
    End Set
End Property
Public Property NivelEsfuerzo As String
    Get
        NivelEsfuerzo = miNivelEsfuerzo
    End Get
    Set(ByVal value As String)
        miNivelEsfuerzo = value
    End Set
End Property

```

```

Public Property EnsayosRigido As ClaseEnsayosRigido

    Get
        EnsayosRigido = miEnsayosRigido
    End Get
    Set(ByVal value As ClaseEnsayosRigido)
        miEnsayosRigido = value
    End Set
End Property
End Class

```

Clases de ensayo de pavimento rígido

```

Public Class ClaseEnsayosRigido
    Private miEnsayosRotura As ClaseEnsayosRotura
    Private miEnsayosCompresion As ClaseEnsayosCompresion
    Private miMr28dias As Double
    Private miFc28dias As Double
    Private miE28dias As Double
    Public Property EnsayosRotura As ClaseEnsayosRotura
        Get
            EnsayosRotura = miEnsayosRotura
        End Get
        Set(ByVal value As ClaseEnsayosRotura)
            miEnsayosRotura = value
        End Set
    End Property
    Public Property EnsayosCompresion As ClaseEnsayosCompresion
        Get
            EnsayosCompresion = miEnsayosCompresion
        End Get
        Set(ByVal value As ClaseEnsayosCompresion)
            miEnsayosCompresion = value
        End Set
    End Property
    Public Property Mr28dias As Double
        Get
            Mr28dias = miMr28dias
        End Get
        Set(ByVal value As Double)
            miMr28dias = value
        End Set
    End Property
    Public Property Fc28dias As Double
        Get
            Fc28dias = miFc28dias
        End Get
        Set(ByVal value As Double)
            miFc28dias = value
        End Set
    End Property
    Public Property E28dias As Double
        Get
            E28dias = miE28dias
        End Get

```

```

        Set(ByVal value As Double)
            miE28dias = value
        End Set
    End Property
End Class

```

Ensayos de elasticidad y rotura

```

Public Class ClaseEnsayosRotura
    Private miElasticidad As Double()
    Private miRotura As Double()
    Public Property Elasticidad() As Double()
        Get
            Elasticidad = miElasticidad
        End Get
        Set(ByVal value As Double())
            miElasticidad = value
        End Set
    End Property
    Public Property Rotura() As Double()
        Get
            Rotura = miRotura
        End Get
        Set(ByVal value As Double())

        End Set
    End Property
End Class

```

Ensayo de compresión

```

Public Class ClaseEnsayosCompresion
    Private miCompresion As Double()
    Public Property Compresion() As Double()
        Get
            Compresion = miCompresion
        End Get
        Set(ByVal value As Double())
            miCompresion = value
        End Set
    End Property
End Class

```

Información de capas granulares

```

Public Class ClaseGranular
    Private miTipoClasificacion As String
    Private miTipoMaterial As String
    Private miPoisson As Double
    Private miKCero As Double
    Private miNivelMr As Integer
    Private miEnsayosk1k2 As ClaseEnsayosk1k2
    Private miTipoDatoEntrada As Integer
    Private miValor As Double

```

```

Private miCalculoMrIcm As Boolean
Private miMr As Double
Private miIcm As ClaseIcm
Public Property TipoClasificacion As String
    Get
        TipoClasificacion = miTipoClasificacion
    End Get
    Set(ByVal value As String)
        miTipoClasificacion = value
    End Set
End Property
Public Property TipoMaterial As String
    Get
        TipoMaterial = miTipoMaterial
    End Get
    Set(ByVal value As String)
        miTipoMaterial = value
    End Set
End Property
Public Property Poisson As Double
    Get
        Poisson = miPoisson
    End Get
    Set(ByVal value As Double)
        miPoisson = value
    End Set
End Property
Public Property KCero As Double
    Get
        KCero = miKCero
    End Get
    Set(ByVal value As Double)
        miKCero = value
    End Set
End Property
Public Property NivelMr As Integer
    Get
        NivelMr = miNivelMr
    End Get
    Set(ByVal value As Integer)
        miNivelMr = value
    End Set
End Property
Public Property Ensayosk1k2 As ClaseEnsayosk1k2
    Get
        Ensayosk1k2 = miEnsayosk1k2
    End Get
    Set(ByVal value As ClaseEnsayosk1k2)
        miEnsayosk1k2 = value
    End Set
End Property
Public Property TipoDatoEntrada As Integer
    Get
        TipoDatoEntrada = miTipoDatoEntrada
    End Get

```

```

        Set(ByVal value As Integer)
            miTipoDatoEntrada = value
        End Set
    End Property
    Public Property Valor As Double
        Get
            Valor = miValor
        End Get
        Set(ByVal value As Double)
            miValor = value
        End Set
    End Property
    Public Property CalculoMrIcm As Boolean
        Get
            CalculoMrIcm = miCalculoMrIcm
        End Get
        Set(ByVal value As Boolean)
            miCalculoMrIcm = value
        End Set
    End Property
    Public Property Mr As Double
        Get
            Mr = miMr
        End Get
        Set(ByVal value As Double)
            miMr = value
        End Set
    End Property
    Public Property Icm As ClaseIcm
        Get
            Icm = miIcm
        End Get
        Set(ByVal value As ClaseIcm)
            miIcm = value
        End Set
    End Property
End Class

```

Clase ensayo triaxial

```

Public Class ClaseEnsayosk1k2
    Private miK1 As Double
    Private miK2 As Double
    Public Property K1 As Double
        Get
            K1 = miK1
        End Get
        Set(ByVal value As Double)
            miK1 = value
        End Set
    End Property
    Public Property K2 As Double
        Get
            K2 = miK2

```

```

        End Get

        Set(ByVal value As Double)
            miK2 = value
        End Set
    End Property
End Class

```

Información de granulometría y suelo

```

Public Class ClaseIcm
    Private miGranulometria As DataTable
    Private miIp As Double
    Private miLl As Double
    Private miMaxDensidadSeca As Double
    Private miGravedadEspecifica As Double
    Private miPermeabilidad As Double
    Private miHumedadOptima As Double
    Private miOptimoGradoSaturacion As Double
    Private miCurvaAguaSuelo As ClaseCurvaAguaSuelo
    Public Property Granulometria As DataTable
        Get
            Granulometria = miGranulometria
        End Get
        Set(ByVal value As DataTable)
            miGranulometria = value
        End Set
    End Property
    Public Property Ip As Double
        Get
            Ip = miIp
        End Get
        Set(ByVal value As Double)
            miIp = value
        End Set
    End Property
    Public Property Ll As Double
        Get
            Ll = miLl
        End Get
        Set(ByVal value As Double)
            miLl = value
        End Set
    End Property
    Public Property MaxDensidadSeca As Double
        Get
            MaxDensidadSeca = miMaxDensidadSeca
        End Get
        Set(ByVal value As Double)
            miMaxDensidadSeca = value
        End Set
    End Property
    Public Property GravedadEspecifica As Double
        Get
            GravedadEspecifica = miGravedadEspecifica
        End Get

```

```

        Set(ByVal value As Double)
            miGravedadEspecifica = value
        End Set
    End Property
    Public Property Permeabilidad As Double
        Get
            Permeabilidad = miPermeabilidad
        End Get
        Set(ByVal value As Double)
            miPermeabilidad = value
        End Set
    End Property
    Public Property HumedadOptima As Double
        Get
            HumedadOptima = miHumedadOptima
        End Get
        Set(ByVal value As Double)
            miHumedadOptima = value
        End Set
    End Property
    Public Property OptimoGradoSaturacion As Double
        Get
            OptimoGradoSaturacion = miOptimoGradoSaturacion
        End Get
        Set(ByVal value As Double)
            miOptimoGradoSaturacion = value
        End Set
    End Property
    Public Property CurvaAguaSuelo As ClaseCurvaAguaSuelo
        Get
            CurvaAguaSuelo = miCurvaAguaSuelo
        End Get
        Set(ByVal value As ClaseCurvaAguaSuelo)
            miCurvaAguaSuelo = value
        End Set
    End Property
End Class

```

Curva característica del suelo

```

Public Class ClaseCurvaAguaSuelo
    Private miAf As Double
    Private miBf As Double
    Private miCf As Double
    Private miHr As Double
    Public Property Af As Double
        Get
            Af = miAf
        End Get
        Set(ByVal value As Double)
            miAf = value
        End Set
    End Property

```

```
Public Property Bf As Double
    Get
        Bf = miBf
    End Get
    Set(ByVal value As Double)
        miBf = value
    End Set
End Property
Public Property Cf As Double
    Get
        Cf = miCf
    End Get
    Set(ByVal value As Double)
        miCf = value
    End Set
End Property
Public Property Hr As Double
    Get
        Hr = miHr
    End Get
    Set(ByVal value As Double)
        miHr = value
    End Set
End Property
End Class
```