UNIVERSIDAD MICHOACANA DE SAN NICOLÁS DE HIDALGO

FACULTAD DE INGENERIA ELÉCTRICA

“SEMÁFOROS INTELIGENTES UTILIZANDO PLC SIEMENS 1500 Y HMI”

TESIS

PARA OBTENER EL TÍTULO DE:

INGENIERO EN ELECTRÓNICA

PRESENTA:

Rafael Dassaev Rodríguez Contreras

ASESOR:

M.C. Octavio Barriga Torres

CO-ASESOR:

Dr. José Aarón Padilla García

Octubre del 2019, Morelia Michoacán

AGRADECIMIENTOS

A MIS PADRES

Quiero agradecer a mis padres que me ayudaron y me apoyaron en las buenas y en las malas, por enseñarme lo importante de la vida, crearon un hogar, una familia, me dieron una educación, me enseñaron a trabajar y no darme por vencido, a seguir adelante y perseguir mis metas, por su apoyo incondicional.

A mis hermanos

Quiero agradecer a mis hermanos por ser mi familia y alentarme en todo este tiempo, darme ánimos, ayudarme cuando lo necesite y cuidar de mí, por apoyarme en mis decisiones.

A mi familia y amigos

Gracias por ser una parte importante de mi vida, gracias por sus consejos, por darme de comer, por apoyarme cuando lo necesitaba.

A mi novia

Por ser una persona tan maravillosa e importante en mi vida, por estar cuando lo necesite, por creer en mí y darme ánimos, por sacarme una sonrisa en el momento que necesitaba.

Resumen

En el presente proyecto se implementó la automatización de semáforos inteligentes con un PLC como herramienta principal, siendo necesario la implementación de varias secuencias para determinar los diferentes casos, teniendo dos modos de manejo, el modo automático que es el que detecta la fecha y hora para hacer diferentes cambios de trabajo de horas y días, el modo dos que es el modo manual, cuando está funcionando el modo manual, el modo automático no funciona y viceversa, el modo manual no detecta fecha y hora ya que este solo puede ser activado por algún caso especial, dependiendo de la duración del caso especial este funcionará y al término del modo manual se regresará al modo automático.

PALABRAS CLAVES: Implementación, automatización, inteligente, mejoramiento y razonamiento.

Abstract

In this project, intelligent traffic light automation was implemented with a PLC as the main tool, being necessary to implement several sequences to determine the different cases, having two management modes, the mode is the automatic mode that is the one that detects the date and time to make the different work changes of hours and days, the mode two which is the manual mode, when the manual mode is working the automatic mode does not work and vice versa, the manual mode doesn´t detect date and time since this can only be activated by some special case, depending on the duration of the special case it will work and at the end of the manual mode it will return to the automatic mode.

Índice general

Capítulo 1 Controladores lógicos programables 1
1.1. Introducción 1
1.2. Objetivo general 1
1.2.1. Objetivo especifico 1
1.3. Conceptos generales 2
1.4. Funciones básicas de un PLC 3
1.5. Estructura del PLC 4
1.5.1. Funciones generales de las secciones internas del PLC 4
1.6. Lenguajes de programación 7
1.7. Ventajas y desventajas de los PLC 9
Capítulo 2 Semáforos 11
2.1. Introducción 11
2.2. Clasificación de los semáforos 12
2.2.1. Ventajas y desventajas de los semáforos 12
2.3. Aplicación de semáforos inteligentes 13
2.3.1. Análisis de tráfico vehicular 13
2.4. Diferentes tecnologías en los semáforos inteligentes 14
2.4.1. Semáforo inteligente con RFID 14
2.4.2. Semáforo inteligente usando redes de sensores inalámbricos 15
2.4.3. Semaforización inteligente mediante el procesamiento de imágenes 16
2.4.4. Semaforización inteligente implementando sistemas embebidos 19
2.5. Los semáforos en la actualidad 20
Capítulo 3 Metodología 22
3.1. Planteamiento del problema 22
3.2. Justificación 22
3.3. Investigación de campo 23
3.3.1. Investigación de la situación actual del tráfico vehicular en la ciudad de Morelia 23
3.4. Implementación 30
3.5. Descripción del sistema 31
Capítulo 4 Descripción del software 34
4.1. Vista del portal 34
4.2. PLCSIM V13 35

4.2.1. Vista principal del PLCSIM V13 36
4.3. Configuración de la comunicación HMI 37
Capítulo 5 Implementación semáforo inteligente 39
5.1. Estructura semáforos inteligentes 39
5.2. Bloque de datos (DB) 39
5.3. IMPLEMENTACION CON DIAGRAMAS DE ESCALERAS 41
5.3.1 Detección de hora y fecha 42
5.3.2. Detección de fecha y hora de secuencias 43
5.3.3. Paso para los peatones 46
5.3.4. Secuencia de hora normal y hora pico 47
5.3.5. Secuencia de día especial 54
5.3.6. Selección modo manual 57
5.4. Simulación HMI 62
5.4.1. Pantalla inicio 62
5.4.2. Hora normal 66
5.4.3. Hora pico 69
5.4.4. Día especiales 72
5.4.5. Caso especial 74
Capítulo 6 Conclusiones y trabajos futuros 77
6.1. Conclusiones 77
6.2. Trabajos futuros 78
Apéndice 79
A.1. Bloque de datos (DB) 79
A.2. Tipos de datos fecha y hora 79
A.2.1. hora 80
A.2.2. Fecha 80
A.2.3. Tod 80
A.2.4. Datos DTL 80
A.3. Contactores 81
A.4. Bobina de relé 81
A.5. Instrucciones activar/desactivar salida 82
A.6. Flipflop de RS/SR 82
A.6.1. RS 82

A.6.2. SR 83
A.7. Temporizador 83
A.8. Comparación 84
A.9. Instrucciones avanzadas 85
A.9.1. Ajustar y leer el reloj del sistema 85
A.10. Visualizador HMI 87
A.11. Herramientas de visualización en la HMI 88
A.11.1. Circulo 88
A.11.2. Botón 88
A.11.3. Campo de texto 88
A.11.4. Visor de gráficos 89
A.11.5. Campo E/S 89
A.11.6. Rectángulo 89
Bibliografía 90

Índice de figuras

Figura 1. 1: Esquema general de sistemas automatizados. 2
Figura 1. 2: Componente general de un PLC (Programmig Logic Control). 4
Figura 1. 3: Comunicación de una PC a un PLC. 6
Figura 1. 4: Arquitectura interna del PLC. 6
Figura 1. 5: Símbolo para programar en lenguaje de diagramas de función/bloques. 9

Figura 2. 1: Un semáforo y sus significados. 11
Figura 2. 2: Clasificación de los semáforos. 12
Figura 2. 3: Esquema de semáforo inteligente utilizando RFID. 14
Figura 2. 4: Esquema del sistema de semaforización usando sensores inalámbricos. 15
Figura 2. 5: Esquema de semáforo basado en procesamiento digital de señales. 17
Figura 2. 6: Concentración vehicular. 18
Figura 2. 7: Cámara KLIP Xtreme. 18
Figura 2. 8: Placa Raspberry Pi. 19
Figura 2. 9: Interfaz gráfica del entorno. 20
Figura 2. 10: Semáforo con temporizador para automóviles y peatones. 21

Figura 3. 1: Tráfico vehicular en la calle El Nigromante. 25
Figura 3. 2: Tráfico vehicular en la calle Abasolo. 25
Figura 3. 3: Tráfico vehicular en la calle Morelos norte. 26
Figura 3. 4: Tráfico vehicular en la calle Virrey de Mendoza. 26
Figura 3. 5: Tráfico vehicular en la calle vasco de Quiroga. 27
Figura 3. 6: Tráfico vehicular en la calle Belisario Domínguez. 27
Figura 3. 7: Tráfico vehicular en la calle Dr. Miguel Silva. 28
Figura 3. 8: Tráfico vehicular en la calle Revolución. 28
Figura 3. 9:Tráfico vehicular en la calle M. Villalongin. 29
Figura 3. 10: Tráfico vehicular en la Av. Acueducto. 29
Figura 3. 11: Ubicación del sistema de automatización semáforo inteligente. 31
Figura 3. 12: Diagrama de bloques del sistema horas normales. 31
Figura 3. 13: Diagrama de bloques del sistema horas pico. 32
Figura 3. 14: Diagrama de bloques del sistema día especial. 33
Figura 3. 15: Diagrama de bloques del sistema caso especial. 33

Figura 4. 1: Vista TIA Portal. 34
Figura 4. 2: Vista del proyecto. 35
Figura 4. 3: Vista general PLCSIM V13. 36
Figura 4. 4: Vista del proyecto PLCSIM V13. 36
Figura 4. 5: Panel de control. 37
Figura 4. 6: Comunicación PLC-HMI "PLCSIM S7-1200/S7-1500.TCPIP.1". 38

Figura 5. 1: Semáforo. 39
Figura 5. 2: Memoria de bloque de datos (DB) global e instancia. 40
Figura 5. 3: Vista del DB global Reloj_RTC. 41

Figura 5. 4: Vista del DB global temporizador. 41
Figura 5. 5: Implementación de modo automático y detección de fecha y hora. 42
Figura 5. 6: Implementación de día domingo. 43
Figura 5. 7: Implementación de detección de días lunes a viernes. 44
Figura 5. 8: Implementación de detección de días lunes a viernes. 45
Figura 5. 9: Implementación de detección de día sábado. 45
Figura 5. 10: Implementación de detección de día sábado. 46
Figura 5. 11: Implementación de secuencia para el paso de los peatones. 46
Figura 5. 12: Implementación de secuencia para el paso de los peatones. 47
Figura 5. 13: Visualización para el paso de los peatones. 47
Figura 5. 14: Implementación de activadores secuencia hora pico. 48
Figura 5. 15: Implementación de secuencia hora pico, semáforos principales. 48
Figura 5. 16: Implementación de secuencia hora pico, semáforos principales. 49
Figura 5. 17: Implementación de secuencia hora pico, semáforos principales. 50
Figura 5. 18: Implementación de secuencia hora pico, semáforos secundarios. 50
Figura 5. 19: Implementación de secuencia hora pico, semáforos secundarios. 51
Figura 5. 20: Implementación de secuencia hora pico, semáforos secundarios. 52
Figura 5. 21: Implementación para el paro de secuencia hora pico. 52
Figura 5. 22: Visualización para la hora pico. 53
Figura 5. 23: Visualización para la hora pico. 54
Figura 5. 24: Implementación de activadores de día especial. 54
Figura 5. 25: Implementación de secuencia día especial, semáforos secundarios. 55
Figura 5. 26: Implementación de secuencia día especial, semáforos principal. 55
Figura 5. 27: Implementación de secuencia día especial, semáforos principales. 56
Figura 5. 28: Visualización para días especiales. 57
Figura 5. 29: Implementación de modo manual y caso especial. 57
Figura 5. 30: Implementación de activadores de caso especial. 58
Figura 5. 31: Implementación de secuencia caso especial, semáforos grupo 1. 58
Figura 5. 32: Implementación de secuencia caso especial, semáforos grupo 1. 59
Figura 5. 33: Implementación de secuencia caso especial, semáforos grupo 2. 60
Figura 5. 34: Implementación de secuencia caso especial, semáforos grupo 2. 61
Figura 5. 35: Visualización caso especial. 61
Figura 5. 36: Tabla de variables en PLCIM. 62
Figura 5. 37: Pantalla inicio. 63
Figura 5. 38: Modo automático funcionado en hora normal. 63
Figura 5. 39: Modo automático funcionando en día especial. 64
Figura 5. 40: Muestra del funcionamiento de los semáforos del grupo 1. 65
Figura 5. 41: Muestra del funcionamiento de los semáforos del grupo 3. 65
Figura 5. 42. Funcionamiento del cruce de peatones. 66
Figura 5. 43: Semáforos secundarios en verde. 67
Figura 5. 44: Desincronización de los semáforos del grupo 3. 67
Figura 5. 45: Desincronización de los semáforos del grupo 2. 68
Figura 5. 46: Desincronización de los semáforos del grupo 1. 68
Figura 5. 47: Semáforos principales en verde. 69
Figura 5. 48: Semáforos secundarios en verde. 70
Figura 5. 49: Desincronización de semáforos del grupo 3. 70
Figura 5. 50: Desincronización de semáforos del grupo 2. 71

Figura 5. 51: Desincronización de los semáforos del grupo 1. 71
Figura 5. 52: Semáforos principales en verde. 72
Figura 5. 53: Semáforos principales sincronizados en verde. 73
Figura 5. 54: Semáforos secundarios sincronizados en verde. 73
Figura 5. 55: Modo manual funcionando. 74
Figura 5. 56: Muestra de cómo se agrega el tiempo en caso especial. 75
Figura 5. 57: Tiempos puestos en caso especial. 75
Figura 5. 58: Funcionamiento de caso especial. 76
Figura 5. 59: Funcionamiento de caso especial. 76

Figura A. 1: Símbolo de contadores, a) normalmente cerrado, b) normalmente abierto. .. 81
Figura A. 2: Símbolo de Bobinas, a) bobina de salida, b) bobina de salida negada. 82
Figura A. 3: Símbolo de bobina con salida set. 82
Figura A. 4: Símbolo de bobina con salida reset. 82
Figura A. 5. Símbolos de flipflop RS/SR. 83
Figura A. 6: Símbolos de los temporizadores. 84
Figura A. 7: Símbolo de comparación. 84
Figura A. 8: Símbolo RD_SYS_T. 85
Figura A. 9: Símbolo de RD_LOC_T. 85
Figura A. 10: Símbolo T_CONV. 86
Figura A. 11: Pantalla HMI. 87
Figura A. 12: Visualizador de circulo. 88
Figura A. 13: Visualizador de botón. 88
Figura A. 14: Visualizador de campo de texto. 88
Figura A. 15: Visualizador de visor de gráficos. 89
Figura A. 16: Visualizador de campo E/S. 89
Figura A. 17: Visualizador de rectángulo. 89

Índice de tablas

Tabla 1.1: Equivalencia de instrucciones-contactos. 8

Tabla A. 1: Tipos de datos fecha y hora. 79
Tabla A. 2: Tamaño y rango para un DTL. 80
Tabla A. 3: Elementos de la estructura DTL. 81
Tabla A. 4: Tamaño y rango del tipo de datos TIME. 84
Tabla A. 5: Tipos de datos para los parámetros. 85
Tabla A. 6: Tipos de datos para los parámetros. 86
Tabla A. 7: Tipos de datos para los parámetros T_CONV. 87

Capítulo 1 Controladores lógicos programables

1.1. Introducción

En la naturaleza los sistemas de control son comunes, funcionan de forma exacta y precisa, manteniendo un equilibrio en el ciclo de la evolución y desarrollo de los seres vivos [1].

Los controladores han ido asumiendo e incorporando cada vez más responsabilidades en su función, desde la propia automatización o la regulación de magnitudes físicas hasta la seguridad de las personas y del medio ambiente. Su sencilla aplicabilidad, su versatilidad y su amplia divulgación le han dado el carácter de universalidad, lo cual explica su constante crecimiento y su gran futuro. Con el paso del tiempo y gracias al desarrollo de la microelectrónica, los PLC se han convertido en sistemas electrónicos complejos cuya organización y arquitectura están especialmente orientados a facilitar el diseño e implementación de sistemas electrónicos de control de cualquier tipo de proceso industrial [2].

En el campo de la industria se han desarrollado los controladores lógicos programables (PLC, de las siglas en inglés Programming logic control) que son dispositivos electrónicos a los que se les pueden programar lógicas para los sistemas de producción, aumentando la calidad del producto y mejorando los servicios que proporcionan los equipos o procesos que se controlan tales como (bandas transportadoras, motores, control en líneas de montaje, etc.), estos dispositivos han adquirido mucha popularidad gracias a sus características y gran funcionamiento por lo que son utilizados en casi todos los sectores de la industria, ya que hacen que el proceso sea más fácil y más eficiente gracias al proceso de automatizado [2].

1.2. Objetivo general

Establecer una solución a través de la automatización de semáforos inteligentes que permita realizar con eficiencia la disminución de la aglomeración vehicular.

1.2.1. Objetivo especifico

a) Analizar los controladores utilizados y PLC en semáforos inteligentes.
b) Desarrollar una aplicación del PLC en un sistema de automatización de semáforos inteligentes.

c) Reducir el tráfico vehicular en la zona.
d) Diseñar los diferentes modos y secuencias de trabajo.

1.3. Conceptos generales

Se planteó el desarrollo de equipos capaces de procesar y memorizar variables físicas, que constituyen sistemas de tratamiento de la información. La necesidad de estos sistemas de tratamiento se remonta a los primeros estados del desarrollo de la ciencia y la tecnología, con el descubrimiento de la electricidad y el gran avance tecnológico de la electrónica lo que permitió el desarrollo de los sistemas que pueden memorizar y procesar información mediante señales eléctricas con un consumo muy pequeño de energía lo cual ha permitido reducir tamaños, costo, eficacia y beneficios [2].

Los sistemas automatizados están formados por tres partes [1]:

1. La parte operativa de los procesos que se desean controlar.
2. La parte de control y/o controladores que se requieren para gobernar el proceso operativo deseado.
3. La parte de supervisión y explotación de sistemas que se utilizan como interfaz entre el proceso controlado y el operador, el cual se le llama HMI (Interfaz Hombre Maquina).

Para poder medir las diferentes variables físicas que no son eléctricas como la temperatura, la presión, desplazamiento de un objeto, la posición, velocidad, aceleración, etc. Es necesario un transductor (interface) o bien una etapa de acondicionamiento, el cual se encarga de transformar una variable física en una señal eléctrica. En la Figura 1.1, se muestra una etapa de control de procesos automatizados.

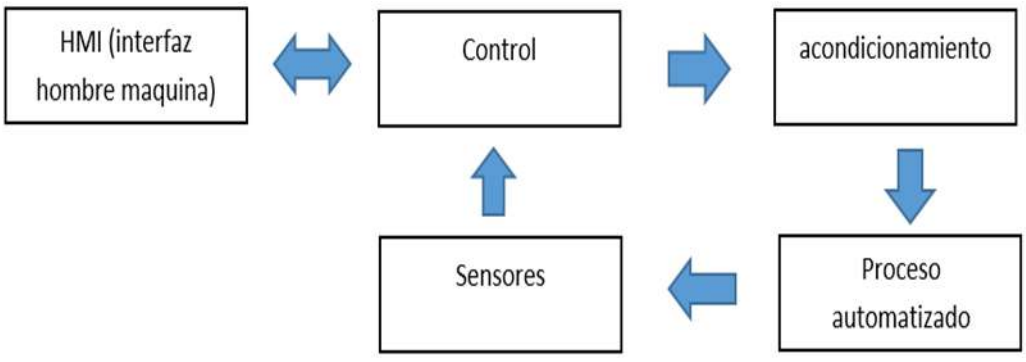


Figura 1. 1: Esquema general de sistemas automatizados.

La parte de control está compuesta por los sensores o elementos primarios de medición (las cuales son capaces de detectar las magnitudes físicas a medir del

sistema), los actuadores o los elementos de acción final (los cuales se encargan de hacer las acciones sobre los componentes del sistema) que están conectados al autómata programable (PLC) [1].

La información proporcionada o indicada por los sensores se convierte en señales eléctricas, se adaptan o acondicionan a los alcances de trabajo del PLC, se transmite a este por medio de interfaces de conexión. El PLC, con esta información y en base al programa de control, indicará las acciones correspondientes a realizar a los equipos del sistema [1].

1.4. Funciones básicas de un PLC

Dentro de las capacidades básicas del PLC, las cuales hacen que sea posible la comunicación y el control de un sistema, se pueden encontrar las siguientes funciones [3]:

- **La *detección*:** esta función es la encargada de la lectura de la señal de los captadores distribuidos por el sistema.
- ***El mando*:** elabora y envía las acciones al sistema mediante los accionadores y preaccionadores.
- ***El diálogo máquina:*** mantiene un diálogo con los operarios de producción, obedeciendo sus consignas e informando del estado del proceso.
- ***Redes de comunicación:*** permite establecer comunicación con otras partes del control. Las redes industriales permiten la comunicación y el intercambio de datos entre autómatas en tiempo real. En solo un par de milisegundos puedes enviarse telegramas e intercambiar tablas de memoria compartida.
- ***Sistemas de supervisión:*** también los autómatas permiten comunicarse con computadoras provistas de programas de supervisión industrial. Esta comunicación se realiza por una simple conexión por el puerto serie de la computadora.
- ***Control de procesos continuos:*** además de dedicarse al control de sistemas de eventos discretos, los PLC llevan incorporadas funciones que permiten el control de procesos continuos. Disponen de módulos de entrada y salida para señales analógicas y la posibilidad de ejecutar algoritmos PID que ya está programados en el autómata.
- ***Entradas-Salidas distribuidas:*** los módulos de entrada-salida pueden estar distribuidos por la instalación, se comunican con la unidad central del autómata mediante un cable de red.

- ***Buses de campo:*** mediante un solo cable de comunicación se pueden conectar al bus de captadores y accionadores, remplazando al cableado.

1.5. Estructura del PLC

En la Figura 1.2 se observa de manera general los componentes básicos del PLC.

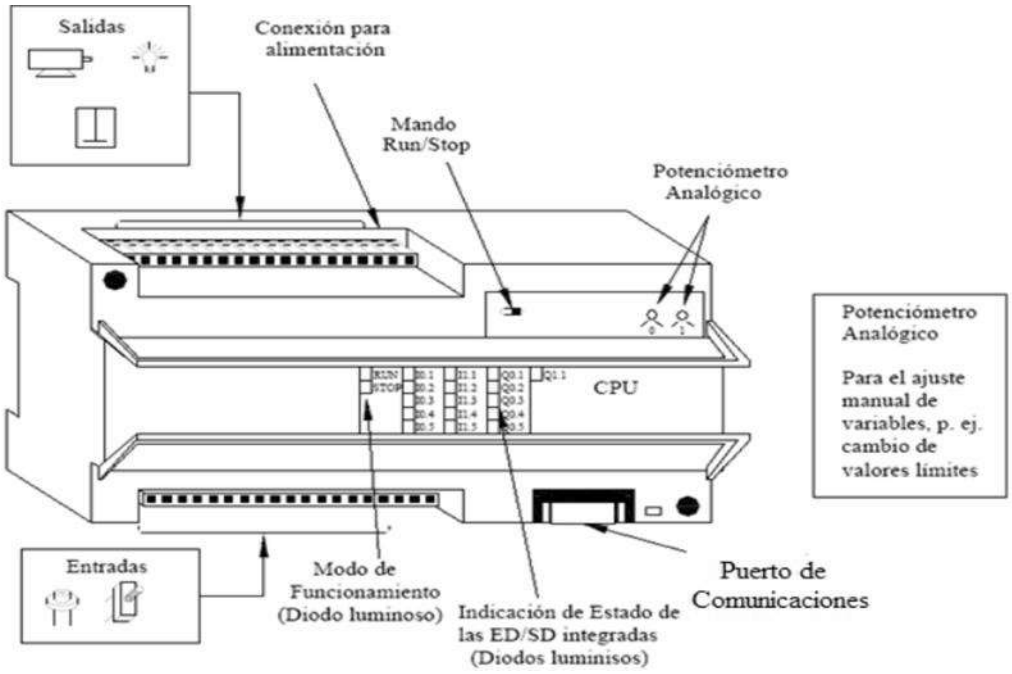


Figura 1. 2: Componente general de un PLC (Programmig Logic Control).

1.5.1. Funciones generales de las secciones internas del PLC

Entre los componentes básicos de un PLC se encuentra:

- **Unidad central de procesos (CPU):** se encarga de procesar el programa del usuario que fue introducido, para ello se disponen de diversas zonas de memoria, instrucciones de programa y registros [3].
- **Sección de entradas:** esta sección es una interfaz que adapta y codifica, las señales procedentes de los dispositivos de entrada, los cuales se denomina captadores o elementos primarios, los captadores pueden ser detectores de

valores, pulsadores, sensores y otros dispositivos de entrada, a este punto de entrada se le asigna una dirección de memoria que refleja el estado abierto/ cerrado (on/off). Estas entradas pueden ser del tipo analógica o digital [1].

- **Sección de salidas:** es una interfaz que decodifica las señales generadas por la CPU, las amplifica y las envía a los actuadores, los actuadores pueden ser lámparas, relés, electroválvulas o cualquier elemento de acción final. Estas salidas pueden ser del tipo analógica o digital [1].
- **Sección de alimentación:** esta sección se encarga de suministrar la alimentación eléctrica a los circuitos del controlador, generalmente 110 V c.a. o cd. a 24 V a la de funcionamiento de los circuitos eléctricos internos del controlador [3].
- ***Unidad de programación:*** es un dispositivo externo mediante el cual se introducen al controlador las instrucciones del programa creado por el usuario. Este dispositivo se le denomina consola de programación y se utiliza para escribir y transferir programas al PLC, así como monitorear las operaciones y modificar datos. La consola se puede conectar directamente al PLC en caso de que solo sea un PLC, también se puede conectar a un adaptador de enlace para poder acceder y monitorear diferentes PLC´s [1].
- ***Periféricos:*** son dispositivos independientes que facilitan el uso del controlador o amplían su campo de aplicación. Entre los periféricos conocidos se encuentran los llamados módulos de expansión [1].
- ***Los módulos de expansión:*** pueden ofrecer un mayor número de puntos de entradas y salidas, diferentes interfaces de comunicación, temporizadores y contadores que pueden ser útiles dependiendo de la aplicación en la cual se utiliza el PLC [1].
- ***Módulo de comunicación:*** son unidades que permiten la interconexión de la CPU con la consola de programación y con los periféricos. La configuración que presenta el conector que enlaza el PLC con los diferentes dispositivos se denomina interfaz y entre las más utilizadas se encuentran el RS-232 (es una interfaz que designa una norma para el intercambio de datos binarios serie entre un DTE “Data Terminal Equipment” por su siglas en ingles equipo terminal de datos y un DCE “Data Communication Equipment” por sus siglas en ingles equipo comunicación de datos) y RS-422/485 (es una interfaz de comunicación de transmisión diferencial entre una computadora y varias UTR Unidades Terminales Remotas). Como se muestra en la siguiente Figura 1.3, la conexión entre un ordenador (PC) y un PLC [1].
- **Memoria:** esta sección contiene todos los datos e instrucciones que se necesita para ejecutar la tarea de control. La memoria interna es la encargada de almacenar los datos intermedios de cálculo y las variables internas que no aparecen directamente sobre las salidas, como un reflejo o imagen de los últimos estados leídos sobre las señales de entrada o enviados a las salidas [1].

- **La memoria de programación:** contiene la secuencia de operaciones que deben realizarse a partir de las señales de entrada para generar las señales de salida, así como los parámetros de configuración del controlador [3].

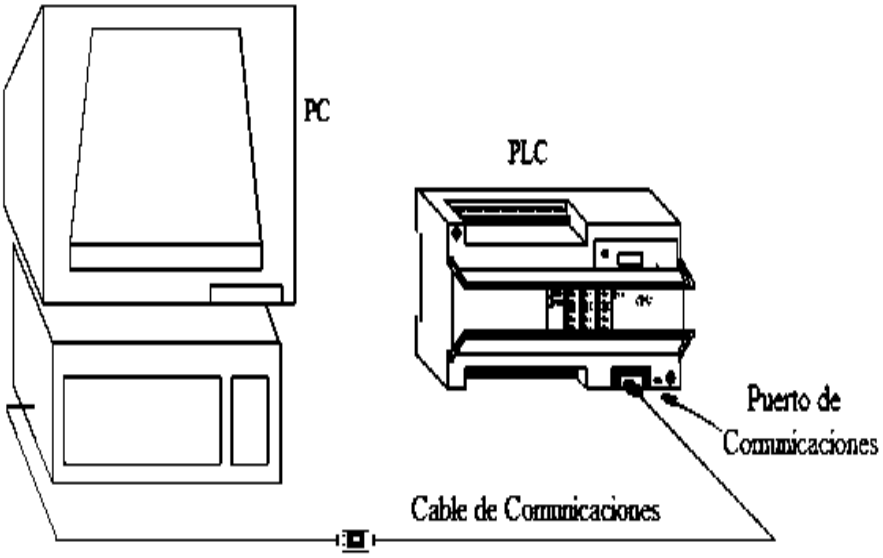


Figura 1. 3: Comunicación de una PC a un PLC.

La arquitectura interna de un PLC se puede observar en la Figura 1.4, en la cual se ven las secciones mencionadas.

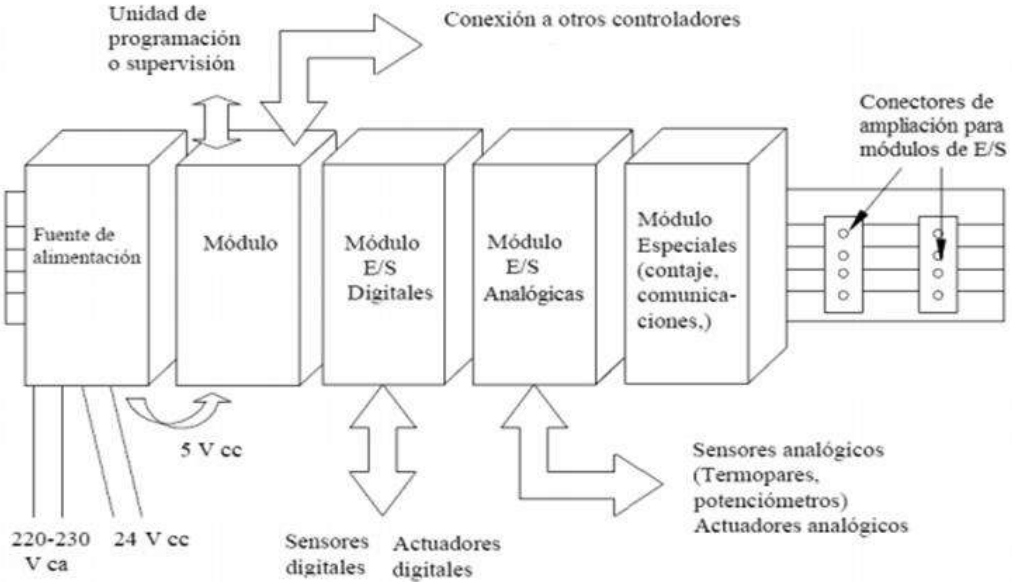


Figura 1. 4: Arquitectura interna del PLC.

1.6. Lenguajes de programación

Un programa es el conjunto de instrucciones, símbolos y órdenes reconocidas por el controlador, mediante una unidad de programación, al que le permite ejecutar la secuencia de control deseada o programada. El lenguaje depende del controlador empleado y del fabricante, el cual decide qué tipo de unidad de programación y el intérprete que utiliza su máquina [3].

Por lo cual, los lenguajes de programación para los controladores intentan ser lo más parecidos posibles. Así los lenguajes pueden ser [1]:

1. Algebraicos
    - Lenguaje booleano y de instrucciones.
2. Gráficos
    - Diagrama de contactos.
    - Diagrama de funciones/ bloques.

Lenguaje booleano y lista de instrucciones [1]

El lenguaje booleano está constituido por un conjunto de instrucciones que son transcripción literal de las funciones del algebra de Boole (es una estructura algebraica que esquematiza las operaciones lógicas OR, AND, NOT). Estas instrucciones se complementan con otras instrucciones de inicio de sentencia y asignación de resultados (LOD, OUT).

Al lenguaje resultante, que no puede llamarse ya booleano después de ser ampliado con estas extensiones, se le denomina lista de instrucciones.

Lenguaje de Diagrama de contactos [1]

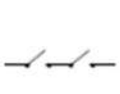
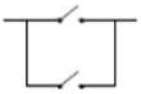
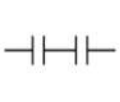
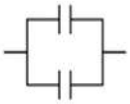
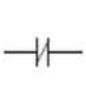
El leguaje de contactos expresa las relaciones entre señales binarias como una sucesión de contactos en serie y en paralelo, según las equivalencias que se muestra en la Tabla 1.1.

Es un lenguaje gráfico, derivado del lenguaje de relevadores, que mediante símbolos representan contactos, solenoides, etc.

Normalmente incluye restricciones en cuanto al número de contactos se puede colocar en cada rama, cualquier función lógica del algebra de Boole se puede

transcribir al diagrama de contactos y viceversa. El diagrama de contactos se creó básicamente para poder hacer la transcripción de los esquemas eléctricos de relés de uso común en el área de automatización previo a la primera aparición de los sistemas programables, por esta razón los diagramas de contactos incluyen temporizadores y contadores.

Tabla 1.1: Equivalencia de instrucciones-contactos.

FUNCIÓN LÓGICA / LENGUAJES	OPERACIÓN PRODUCTO LÓGICO	OPERACIÓN SUMA LÓGICA	OPERACIÓN NEGACIÓN	ASIGNACIÓN DE VALOR
INSTRUCCIONES BOOLEANAS	AND	OR	NOT	OUT
ESQUEMAS DE RELÉS (DIN 40713-16)				
DIAGRAMAS DE CONTACTOS (NEMA/DIN 19239)				

Lenguaje de diagramas de funciones/ bloques [1]

El diagrama de funciones es un lenguaje gráfico que permite programar elementos que aparece como bloques para ser cableados entre sí de forma analógica al esquema de un circuito. Es adecuado para muchas funciones o aplicaciones que involucren el flujo de información o de datos entre componentes de control.

En la Figura 1.5 se muestra el símbolo con el cual están representado la mayoría de los bloques, ya que en algunos bloques puede tener más entradas o más salidas.

La programación por diagrama de funciones/bloques incluye como bloques funciones típicas en automatización como temporizadores, contadores, incluso algunos bloques combinados, como multiplexores, sumadores multiplicadores, etc. Como también el usuario puede crear diagrama de funciones/ bloques y ser llamados cuando sean necesarios.

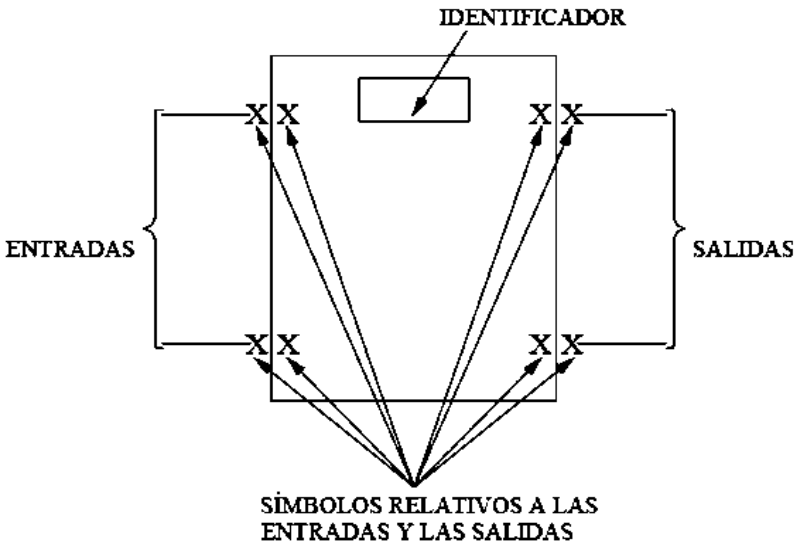


Figura 1. 5: Símbolo para programar en lenguaje de diagramas de función/bloques.

1.7. Ventajas y desventajas de los PLC

Los dispositivos PLC ofrecen muchas ventajas y desventajas, pero debido a que no todos los PLC son iguales, ya que existe gran variedad de PLC, estos están innovando constantemente por lo cual cada modelo es diferente, se hará una referencia a las ventajas y desventajas generales.

Ventajas:

- Escaso espacio para la implementación física, el controlador es muy compacto por lo cual no ocupa mucho espacio.
- Menor costo de obra de instalación.
- Se puede manejar varios dispositivos con el mismo controlador.
- Se puede utilizar para varios proyectos.
- Poco costo de mantenimiento. Además de aumentar la fiabilidad del sistema, al eliminar contactos móviles, los controladores en sí mismos pueden indicar y detectar averías.
- Se puede modificar o incrementar el proyecto sin necesidad de cambiarlo, sin la necesidad de cambiar el cableado.
- Se puede conectar con gran variedad de componentes a la salida o la entrada reduciendo el costo de los materiales.
- Son confiables, por su gran fiabilidad y su duración de vida útil.

Desventajas:

- Costo inicial es uno de los inconvenientes presentes a la hora de realizar un proyecto debido al costo del PLC.
- Falta de conocimientos o de personal que pueda utilizarlo, la falta de un programador que este revisando y mejorando el funcionamiento del PLC, lo que obliga a adiestrar a uno o más técnicos en tal sentido, al no tener personal con conocimiento sobre PLC se ven obligados a contratar personas especializadas en este rublo para darle mantenimiento lo cual es muy costoso.
- Si se necesita cambiar un dispositivo que este fallando o querer expandir más el proceso el costo de los dispositivos es un inconveniente.

Capítulo 2 Semáforos

2.1. Introducción

El flujo vehicular es uno de los factores socioeconómicos que afecta de forma directa a las principales ciudades, para poder obtener o diseñar un control de los sistemas de semaforización, los cuales necesitan ser diseñados para la regulación del tráfico vehicular como el tráfico de los peatones, mediante una secuencia de los tiempos de encendido en las luces de los mismos, que permiten o detienen el avance de los vehículos, así como de los mismos peatones [4].

Un control ineficiente en el tráfico vehicular genera impactos negativos en varios aspectos; por una parte, a nivel ambiental, siendo este la mayor fuente de contaminación acústica en las ciudades, generando ruido, que es un contaminante común, por otro lado, a nivel de seguridad vial, debido a que los accidentes de tráfico constituyen hoy en día una de las principales causas de muerte e invalidez en el mundo [4].

El semáforo utilizado hoy en día es muy importante ya que controla el flujo vehicular y de los peatones, ya que el más utilizado es el que solamente tiene una secuencia de tiempos definidos para el flujo de los vehículos y de los peatones, también funciona como regulador de velocidad ya que entre las distancias de un semáforo a otro no pueden ir a exceso de velocidad, es utilizado también para controlar la circulación en los diferentes carriles de las calles y de los sentidos entre las calles. A continuación, en la Figura 2.1, se muestra una imagen de los semáforos y su significado [5].

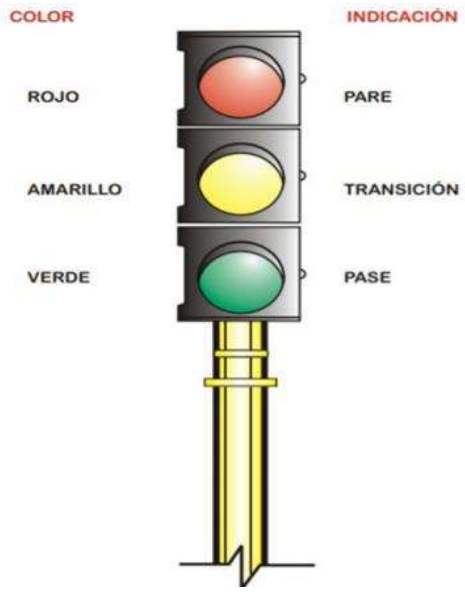


Figura 2. 1: Un semáforo y sus significados.

2.2. Clasificación de los semáforos

La siguiente clasificación que refiere, de acuerdo con el mecanismo de operación y de sus unidades de control, los semáforos se clasifican como se ilustra en la Figura 2.2 [5].

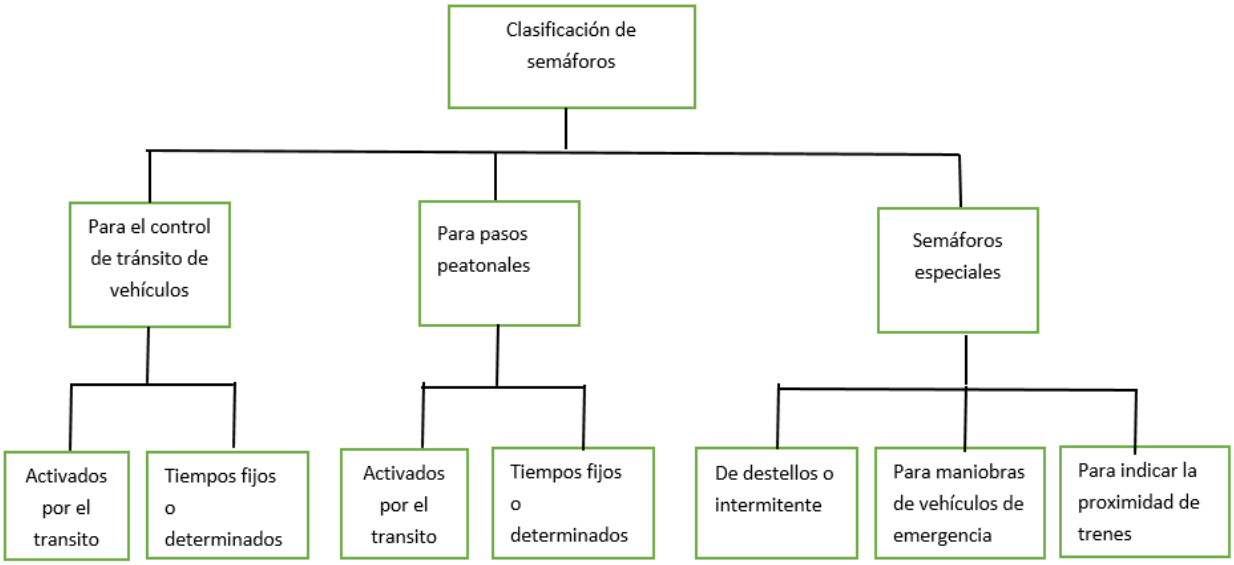


Figura 2. 2: Clasificación de los semáforos.

2.2.1. Ventajas y desventajas de los semáforos

Algunas de las ventajas del uso del semáforo son:

- Proporciona ordenamiento de los movimientos de transito ya que el semáforo asigna el derecho de paso.
- Agrupa a los peatones en aglomeraciones, permite a los peatones cierta facilidad de cruce en las intersecciones.
- Reduce algunos tipos de accidentes.
- Velocidad media/baja de los vehículos.

Algunas de las desventajas son:

- Suele tener fallas seguidas.
- El tiempo de pase de los semáforos es muy corto.
- Suele incrementar el flujo vehicular.

- Aumentan los accidentes menores entre automovilistas.
- Produce estrés y desesperación.
- Produce irrespeto entre los automovilistas.
- Tiempos excesivos de viaje.

2.3. Aplicación de semáforos inteligentes

La implementación de semáforos inteligentes es muy diversa, algunas se enlistan enseguida:

- Controlar la longitud de la fila de los vehículos.
- Semáforos con un mando a distancia que permite que se encienda solo cuando lo necesita una persona con alguna discapacidad.
- Semáforos que reaccionen a la aglomeración de peatones.
- Forzar semáforos en casos especiales (paso de ambulancias, policías, etc.).
- Forzar al semáforo a tomar una decisión cuando exista aglomeración vehicular para que pueda darle fluidez al flujo vehicular.

2.3.1. Análisis de tráfico vehicular

En la actualidad el creciente flujo vehicular a nivel general, hace que las autoridades responsables de la movilidad estén tratando de innovar alternativas para dar solución, a este problema para ello se ha implementado semáforos inteligentes este tipo de semáforos son capaces de tomar decisiones dependiendo de una serie de parámetros de entrada tales como:

- Flujo de vehículos.
- Velocidad media.
- Identificación de la calle.

Algunos de los problemas a solucionar con la implementación de estos semáforos inteligentes son:

- Congestión vehicular.
- Tiempos excesivos de viaje.
- Esperas innecesarias.
- Mayor gasto de combustible.
- Mayor contaminación en el ambiente.

2.4. Diferentes tecnologías en los semáforos inteligentes

En esta sección se mencionarán y analizarán algunos de los tipos de tecnologías que se han implementado en los semáforos inteligentes para mejorar el tráfico vehicular.

2.4.1. Semáforo inteligente con RFID

Su principal función es tener un control inteligente del tráfico, haciendo variar los tiempos de espera en los puntos de intersección de acuerdo a la congestión vehicular. Esta tecnología costa de los siguientes elementos [6]:

- ***Tarjeta RFID:*** consiste fundamentalmente en colocar una tarjeta en cada vehículo que se desee contabilizar.
- ***Punto de acceso***: con una antena se capta el código único de cada tarjeta, entonces se envía una señal de radiofrecuencia de baja potencia y se registra la presencia de un vehículo.
- ***Base de datos:*** la información recibida del punto de acceso se almacena y procesa en un servidor.
- ***Ubicación de servidor de redes:*** donde se almacena y procesa la información
- ***Red de área amplia o WAN (Wide Área Network por sus siglas en inglés):*** es una red de computadoras que une varias redes locales.

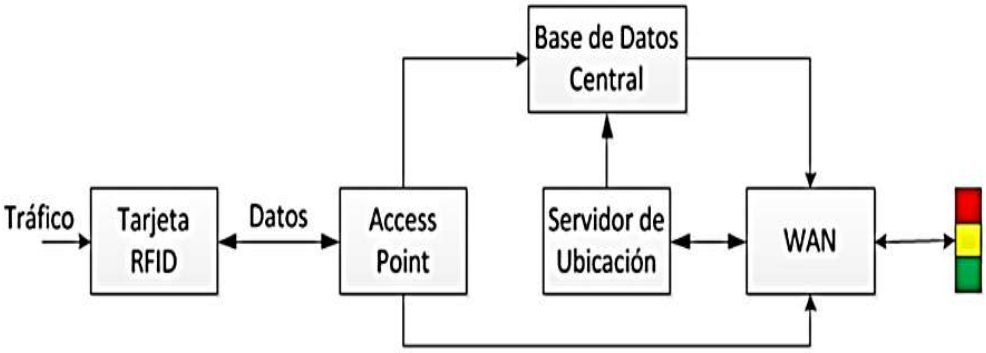


Figura 2. 3: Esquema de semáforo inteligente utilizando RFID.

Este proceso de decisión para el control de tráfico inteligente depende de la información en tiempo real proporcionada por el sistema RFID. Se captan los datos y son almacenados en una base de datos centralizada. Estos sensores captan la ubicación y el tiempo real de cada vehículo y se guardan como una etiqueta que puede ser referenciada a través de un identificador [6].

2.4.2. Semáforo inteligente usando redes de sensores inalámbricos

Capaz de controlar el tráfico y manipular en forma dinámica los tiempos de espera. Maneja básicamente la misma idea que los semáforos que utilizan RFID.

Este sistema cuenta con 2 etapas principales [5]:

- ***La red de sensores inalámbrica (Wireless Sensor Network o WSN):*** cada sensor se encarga de generar los datos de tráfico como el número de vehículos, velocidad de cada vehículo y longitud de separación entre ellos.
- ***La estación base (Base Station o BS):*** los datos adquiridos se envían en tiempo real a la base.

Se utiliza TDMA (Time Division Multiple Access por sus siglas en inglés, acceso múltiple por división de tiempo) para la comunicación, aprovechando en forma eficiente la energía, ya que permite a los nodos de la red entrar en estados inactivos hasta que sus espacios de tiempo sean asignados [5].

En la Figura 2.4, se muestra el esquema típico del sistema de semaforización inteligente usando sensores inalámbricos [5].

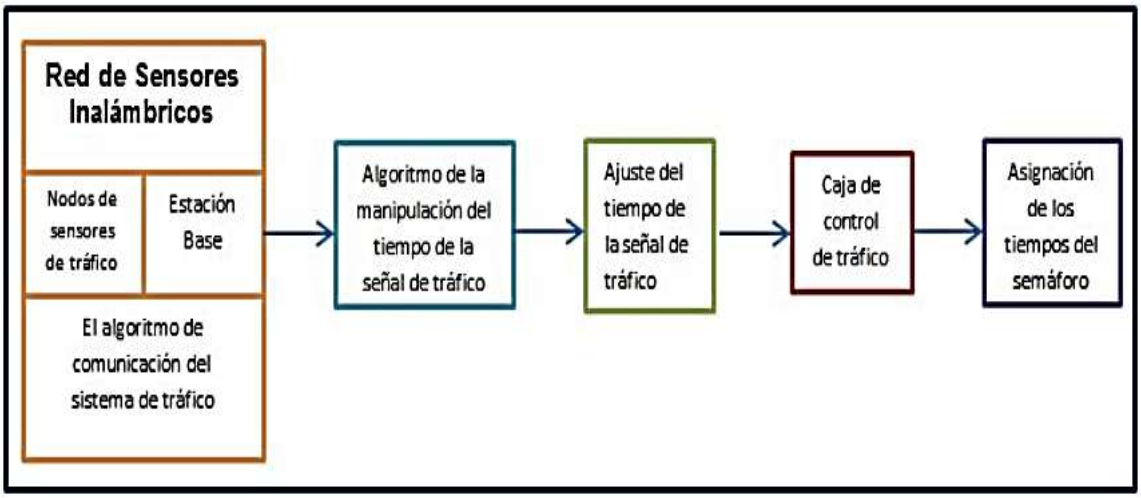


Figura 2. 4: Esquema del sistema de semaforización usando sensores inalámbricos.

Este sistema cuenta con las siguientes etapas [7]:

- ***El nodo de sensores de tráfico:*** se encarga de detectar los vehículos, contarlos y transmitir la información continuamente hacia la estación base.
- ***El sistema de detección de vehículos***: requiere de cuatro componentes: el sensor el cual se encarga de detectar las señales generadas por los vehículos, un procesador para los datos detectados, una unidad de comunicación para transferir los datos a una estación base y una fuente de energía.
- ***El algoritmo de comunicación:*** es el encargado de intercambiar los datos obtenidos por los sensores y la estación base, así como con la caja de control de tráfico.
- ***Los datos obtenidos:*** por la estación base pasa por el algoritmo de la manipulación del tiempo de la señal de tráfico aquí se fijan las duraciones de tiempo de señales de manera dinámica de acuerdo a los parámetros recibidos.
- ***Asignación de tiempos del semáforo:*** es un algoritmo de tiempo de control adaptativo desarrollado para calcular la duración de la luz roja-verde.

2.4.3. Semaforización inteligente mediante el procesamiento de imágenes

La semaforización inteligente mediante procesamiento de imágenes; consiste básicamente en la adquisición de información respecto al lugar de interés a través de un sensor óptico, mediante la imagen captada se puede obtener diferente información como la velocidad, distancia, posición de los vehículos, etc [7].

Mediante el proceso de imágenes también es posible medir la densidad de tráfico y de acuerdo con los datos obtenidos los tiempos de semaforización, el proceso es el siguiente:

Para procesar las imágenes se debe cumplir con un flujo de trabajo los cuales son [7]:

- **Acceder a los datos de las imágenes**: adquirir en vivo o ya capturadas del objeto de interés. Esto se realiza a través de una cámara ya sea digital o analógica.

- **Exploración y descubrimiento:** establecer las características de la imagen como numero de pixeles. Crear y modificar algoritmos. Esto se consigue mediante software (por ejemplo, Matlab, Labiew, Opencv).
- **Aplicación**: obtener resultados y aprovecharlos en una aplicación.

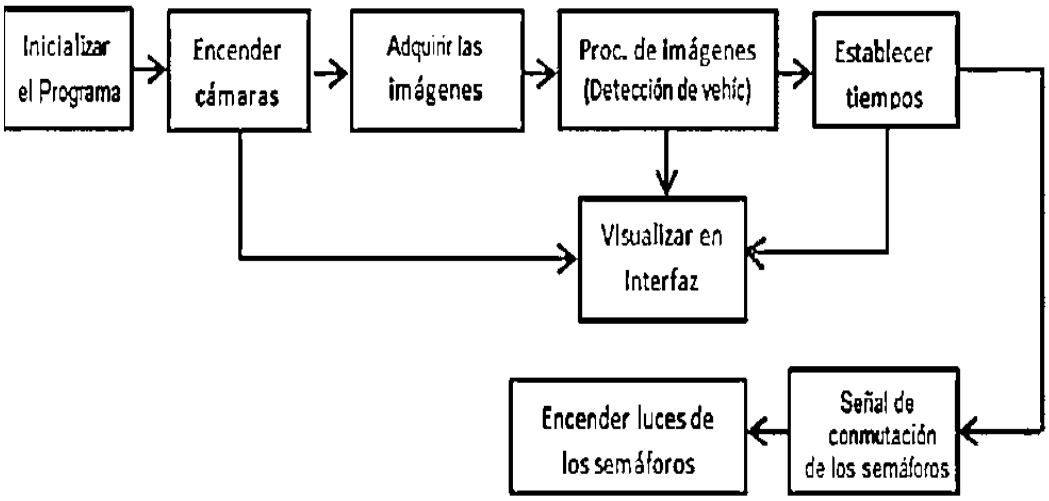


Figura 2. 5: Esquema de semáforo basado en procesamiento digital de señales.

Para procesar las imágenes se debe cumplir con un flujo de trabajo los cuales son [7]:

- ***Acceder a los datos de las imágenes***: adquirir en vivo o ya capturadas del objeto de interés. Esto se realiza a través de una cámara ya sea digital o analógica.
- ***Exploración y descubrimiento***: establecer las características de la imagen como numero de pixeles. Crear y modificar algoritmos. Esto se consigue mediante software (por ejemplo, Matlab, Labiew, Opencv).
- ***Aplicación:*** obtener resultados y aprovecharlos en una aplicación.

En la ciudad de Ambato Ecuador se hizo un estudio de tráfico vehicular de las calles más transitadas en esta ciudad con los datos proporcionados de la unidad municipal de tránsito (UMT), donde el estudio arrojó que a la semana pasan un porcentaje de automóviles como muestra en la Figura 2.6 [7].

Con los resultados obtenidos se decidió hacer el proyecto de "sistema de semaforización inteligente para el control de flujo vehicular mediante el procesamiento digital de imágenes", utilizando como sensor cámara KLIP Xtreme en la Figura 2.7 para obtención de imágenes, una vez obtenidas las imágenes pasan

por un proceso de recolección de imágenes e RGB, el cual mediante un algoritmo y un proceso lo pasa a binario donde un proceso nuevo se lleva acabo para determinación de cual carril hay que darle prioridad [7].

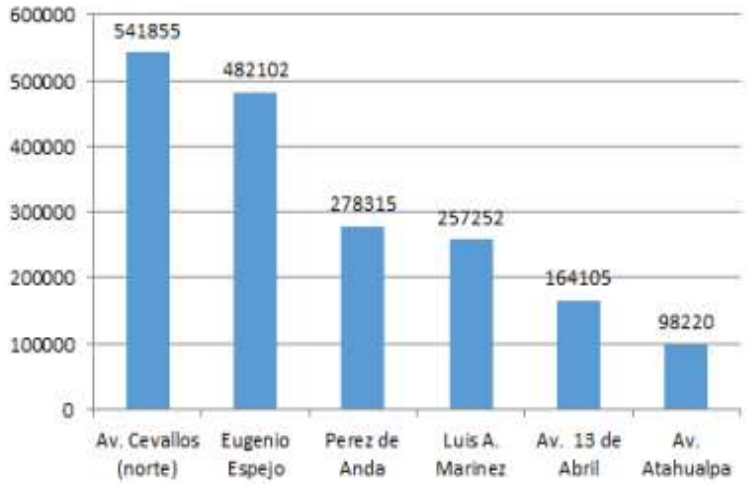


Figura 2. 6: Concentración vehicular.

Figura 2. 7: Cámara KLIP Xtreme.

Características principales de esta cámara son [7]:

- Resolución: 1600 x 1200 pixeles.
- Captura de imágenes: hasta 30 fps (cuadros por segundo).
- Distancia focal: 30 mm.
- Formato de imagen RGB24 y YUY2.
- Interfaz: USB 2.0.

- Longitud de cable: 1.5 m.
- Requisitos de sistema: familia Windows o Mac OS.

2.4.4. Semaforización inteligente implementando sistemas embebidos

Para optimizar el flujo vehicular en una intersección vial se diseñó un algoritmo de control adaptativo y una base de datos que apoya la depuración del rendimiento del controlador, ambos alojados en el sistema embebido Raspberry Pi (ordenador de placa simple utilizado como tarjeta de adquisición de datos, como ejemplo de aplicación en el control de semáforos inteligentes se muestra un ejemplo de aplicación, servidor de impresión, consola de videojuegos retro, controladores para robótica, etc.). Para ello se emuló la intersección de calles de la ciudad de Cúcuta, Colombia, cuyos parámetros serían los sensores magneto resistivos, el encendido de luces y el flujo vehicular llegando a reducir el tiempo de espera de 28 segundos [4].

Figura 2. 8: Placa Raspberry Pi.

Aumentando el flujo vehicular hasta en un 5.5% de esto se puede inferir que es posible integrar el control adaptativo y los sistemas embebidos como herramienta de software y hardware para mejorar el funcionamiento en los sistemas de regulación vial [4].

El entorno simulado se llevó acabo en san José de Cúcuta, en la intersección semafórica calle 10 avenida 0, con la interfaz hecha y la presencia de vehículos generó señales simultáneas, emuladas con una herramienta computacional que garantizó la concurrencia. La manipulación de los tiempos de encendido de las luces de los semáforos por parte del algoritmo de control adaptativo, mejoró la eficiencia del flujo vehicular en la intersección vial [4].

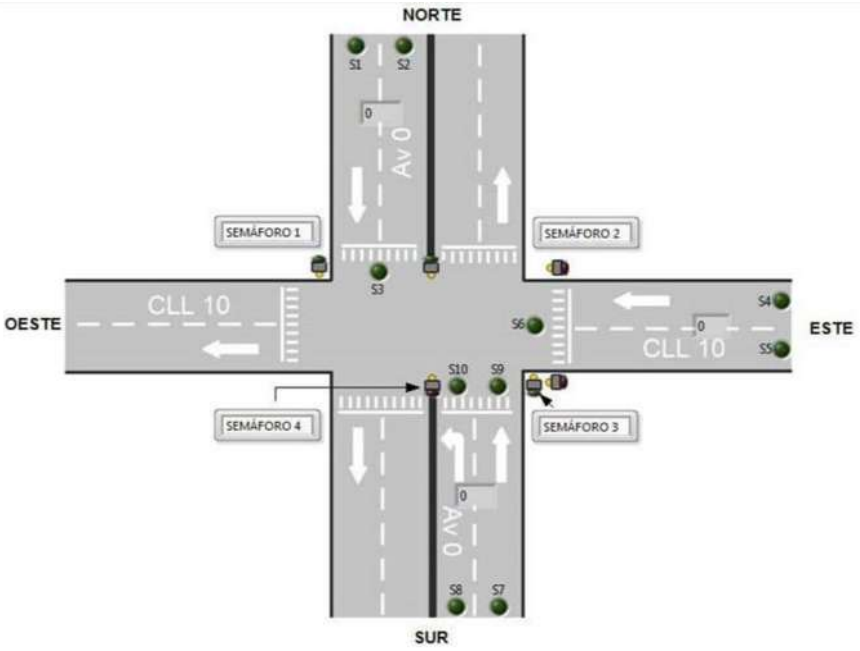


Figura 2. 9: Interfaz gráfica del entorno.

2.5. Los semáforos en la actualidad

En la actualidad los semáforos más utilizados en la mayoría de los países son los que cuentan con temporizadores como se muestra en la Figura 2.10, con la finalidad de indicar tanto a conductores como a los transeúntes el momento que cambiará el semáforo o el tiempo que dura el semáforo en cambiar [6].

Figura 2. 10: Semáforo con temporizador para automóviles y peatones.

De tal manera el conductor sabe cuánto tiempo tendrá para pasar el semáforo y a los transeúntes cuando podrán pasar. También son utilizados otros semáforos sincronizados los cuales su función principal es estar sincronizados con otros semáforos para así poder ofrecer un flujo fluido a los conductores, el problema de estos semáforos es que cuando cambian de verde a rojo o viceversa los conductores no alcanzan a pasar entre los semáforos ya que cuando avanza en uno el trafico acumulado del otro semáforo lo hace quedarse en el semáforo o en el siguiente, lo cual perjudica la movilidad, ya que esto hace que en horas pico se tenga una gran cantidad de automóviles en el tráfico lo cual lleva a más tiempo de recorrido, que se estresen, gran cantidad de ruido y de contaminación [6].

Capítulo 3 Metodología

3.1. Planteamiento del problema

La congestión vehicular en México, así como para el resto del mundo se ha convertido en un gran problema tanto para conductores como para peatones, esto causa conflictos en la población, como ciudad de México, Buenos Aires, Sao Paulo, Quito, Bogotá, Santiago o Lima son ciudades que protagonizan largas filas de vehículos en las vías principales, generando impaciencia en los conductores, imprudencia al volante, ruido de las bocinas, invasiones de carril, estrés, gases de vehículos, violencia verbal e inclusive violencia física [7].

Un problema de hoy en día es el aumento excesivo del parque automotor alrededor del mundo, cada año se construyen cantidades masivas de automóviles por año, lo cual la gente compra cada par de años un carro nuevo, el aumento excesivo del parque automotor no es una excepción en México, el número de automóviles matriculados ha crecido a más de 45 millones 476 mil 133 vehículos según estudios hechos por el INEGI (Instituto Nacional de Estadística y Geografía). En la entidad de Morelia el número de automóviles matriculados del año 1996 al 2019 ha crecido un 301 por ciento en los últimos 23 años, paso de tener 132 automotores a más de 530 mil automóviles, a los cuales hay que sumarles de 200 mil a 300 mil que circulan por la zona metropolitana de Morelia, que comprenden también los municipios como Tarímbaro y Charo entre otros [8].

Las largas filas de automóviles en la ciudad de Morelia, que pueden ser generadas por manifestaciones en las vías principales, por algún accidente vehicular (un automóvil descompuesto, choque entre vehículos, automóviles de carga descargando material, etc.) manifiestan que es necesario el desarrollo de actividades para mejorar la movilidad vehicular. Muchas veces se observa a los agentes de tránsito colocados en las vías de tránsito para poder permitir que el trafico fluya más rápido sin tomar en cuenta el semáforo y lo que indique, al hacer esta maniobra el agente de tránsito corre peligro de ser atropellado, lo que implica que esa maniobra no es óptima ya que no cumple con el funcionamiento del semáforo ni la del agente [7].

3.2. Justificación

El incremento del parque automotor en los últimos años, obliga a tomar medidas de descongestionamiento vehicular, por lo cual diferentes instituciones

gubernamentales han planteado diferentes medidas como; sistema placa, sistema integral del transporte urbano, la motivación al uso de transporte no motorizado [7].

En la actualidad en la ciudad de Morelia existe sistema de semaforización vehicular síncrona, es decir que los semáforos están sincronizados y por lo tal hace que en el trascurso del recorrido en una avenida principal el vehículo solo pueda pasar un semáforo haciendo un poco obsoleta este sistema de semaforización que se emplea actualmente. El sistema de semaforización inteligente permite controlar la duración de las fases de los semáforos, el tiempo máximo del semáforo en estado verde es de 50 segundos en la vía principal y de 40 segundos en la vía secundaria, una vez mencionado, además de que contaría con un sistema vehicular adaptativo para las fases de horas pico y para los días que se cierra la avenida principal para darle más fluidez a la avenida secundaria.

Los beneficios de este sistema de semaforización inteligente será que los conductores transitarán de forma más fluida, consumir menos combustible y minimizar niveles de estrés, reducir la violencia verbal. Los beneficiados de este sistema también serán los peatones que transiten por las vías puesto que tendrán un control sobre el tráfico vehicular.

3.3. Investigación de campo

En el presente proyecto se ha hecho una investigación de campo debido a que se cuenta con datos reales sobre el flujo vehicular en la ciudad de Morelia, lo cual permitió conocer de mejor manera la situación actual del tráfico vehicular.

Para el estudio se tomaron los datos del volumen del tráfico vehicular para cada uno de los semáforos que conforman la intersección de la avenida Madero desde la altura de las Tarascas hasta el mercado de dulces.

3.3.1. Investigación de la situación actual del tráfico vehicular en la ciudad de Morelia

En la ciudad de Morelia existen diferentes puntos donde se concentra la mayor cantidad de tráfico vehicular, solo se centró en una pequeña parte, la cual es la avenida Madero de la altura de las Tarascas hasta el mercado de dulces y sus calles aledañas que cruzan con la avenida Madero, teniendo en cuanta las diferentes situaciones del tráfico en las horas comunes y las horas pico. Se tomó en cuenta las horas pico que es cuando más tráfico hay en la ciudad, las cuales son de 7:00 a las 9:30 estas horas se vuelven pico debido a que muchas personas cruzan esta avenida para poder llegar a las escuelas y trabajos.

La siguiente hora pico de 12:00 a 16:00 cuando la mayoría de los estudiantes de primaria salen de clases y los estudiantes del turno vespertino entran a clases, muchas personas que salen a comer o solo pasan al centro de la ciudad a buscar un lugar donde consumir sus alimentos.

La siguiente hora pico es de 18:30 a 20:30 cuando las personas salen de los trabajos y los estudiantes de los turnos vespertinos terminan actividades en sus escuelas y muchas personas que llegan a pasear al centro tienen que cruzar por esta avenida.

Se tomó en cuenta por ser una situación especial que pasa en la ciudad, los días domingos desde las 7:00 a las 14:00 suele estar cerrada la avenida Madero desde la altura de las Tarascas hasta la altura del mercado de dulces, ese día cierran la avenida para que muchas personas puedan pasear en familia, puedan salir en bicicletas, patines o patinetas, también suelen hacer carreras especiales por diferentes empresas o por causas específicas.

Al realizar el estudio de la situación actual del tráfico se revisó que las calles con mayor concentración vehicular se encuentran:

- Calle El Nigromante esquina con avenida Francisco I. Madero.
- Calle Abasolo esquina con avenida Francisco I. Madero.
- Calle Morelos norte esquina con Francisco I. Madero.
- Calle Virrey de Mendoza esquina con avenida Francisco I. Madero.
- Calle Vasco de Quiroga esquina con avenida Francisco I. Madero.
- Calle Belisario Domínguez esquina con avenida Francisco I. Madero.
- Calle Dr. Miguel Silva esquina con avenida Francisco I. Madero.
- Calle Revolución esquina con avenida Francisco I. Madero.
- Calle M. Villalongin esquina con avenida Francisco I. Madero.
- Calle avenida Acueducto con avenida Francisco I. Madero.

En las Figuras 3.1 a 3.11 se muestra el número de vehículos que circulan en cada calle o avenida verificada, los resultados que se obtuvieron de 4 días (los días que se realizaron los estudios fueron lunes, miércoles, jueves y viernes) de estudio en los diferentes horarios se hizo un promedio de automóviles que pasan en cada calle o avenida.

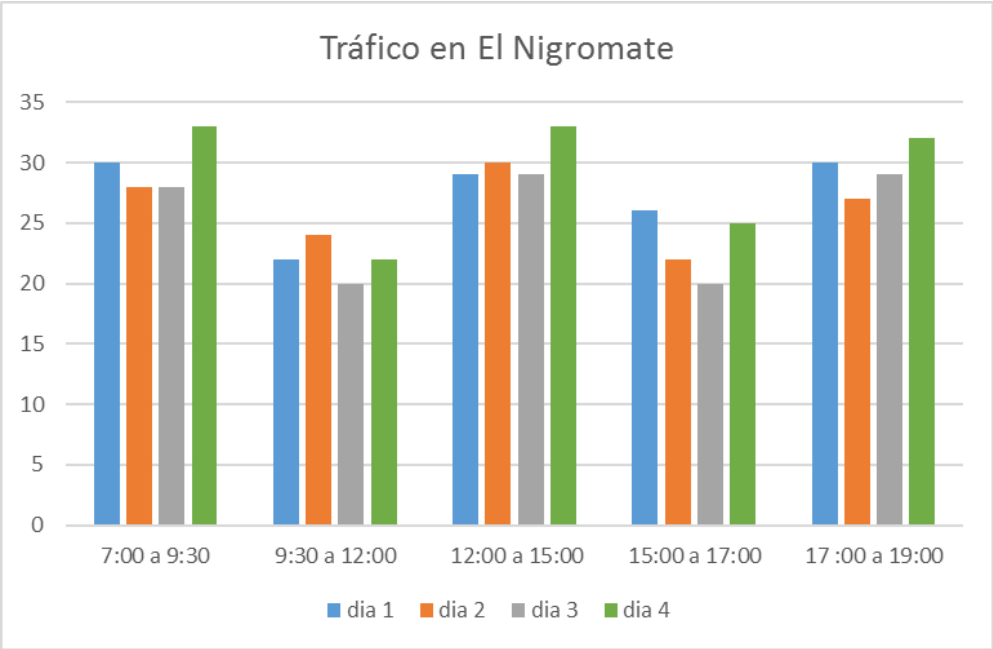


Figura 3. 1: Tráfico vehicular en la calle El Nigromante.

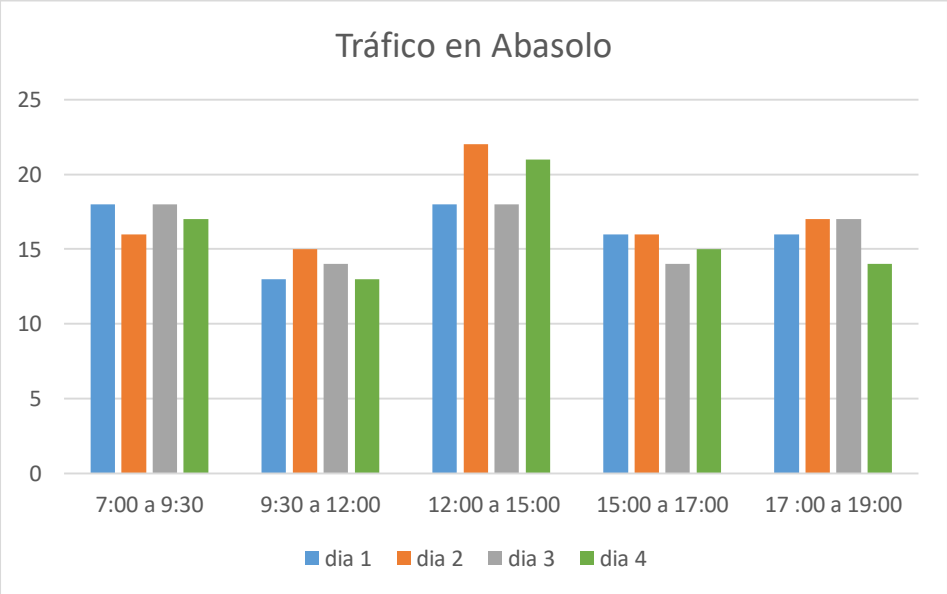


Figura 3. 2: Tráfico vehicular en la calle Abasolo.

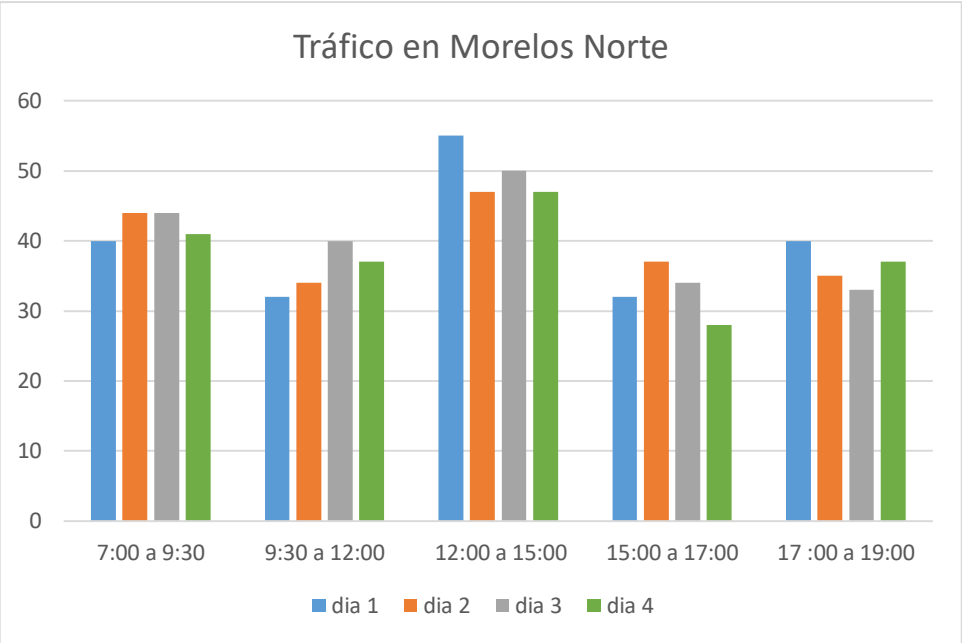


Figura 3. 3: Tráfico vehicular en la calle Morelos norte.

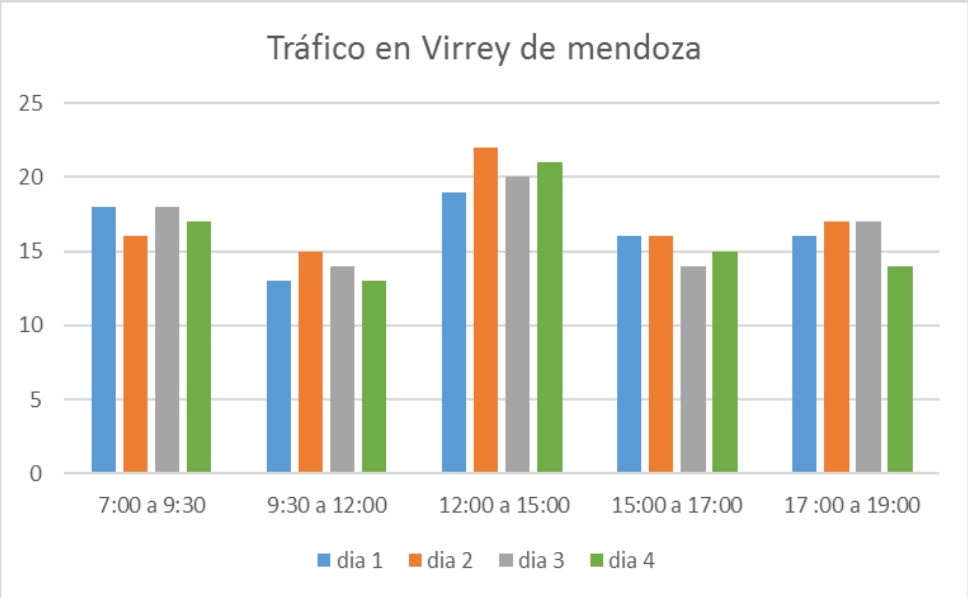


Figura 3. 4: Tráfico vehicular en la calle Virrey de Mendoza.

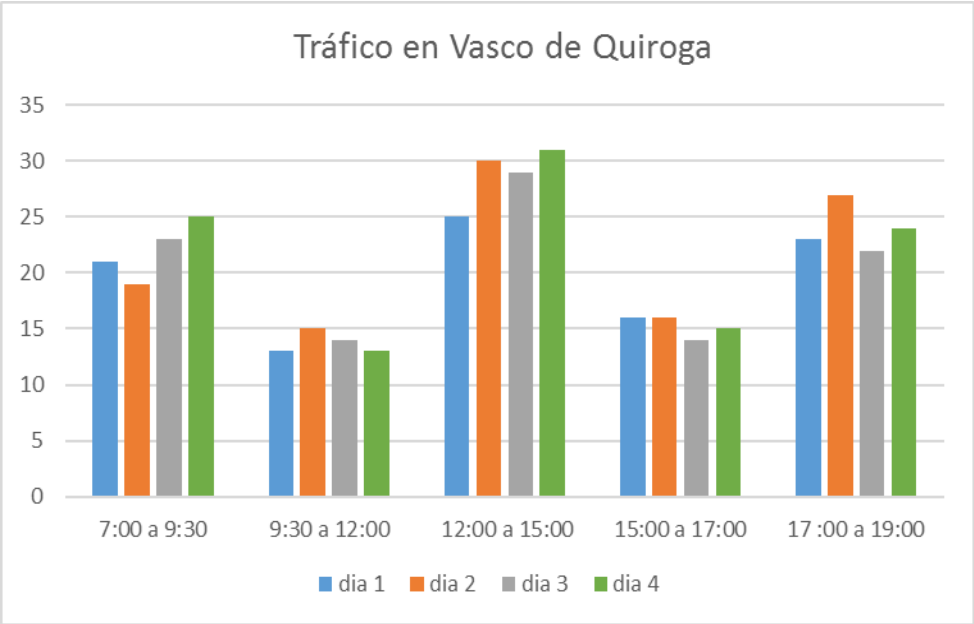


Figura 3. 5: Tráfico vehicular en la calle vasco de Quiroga.

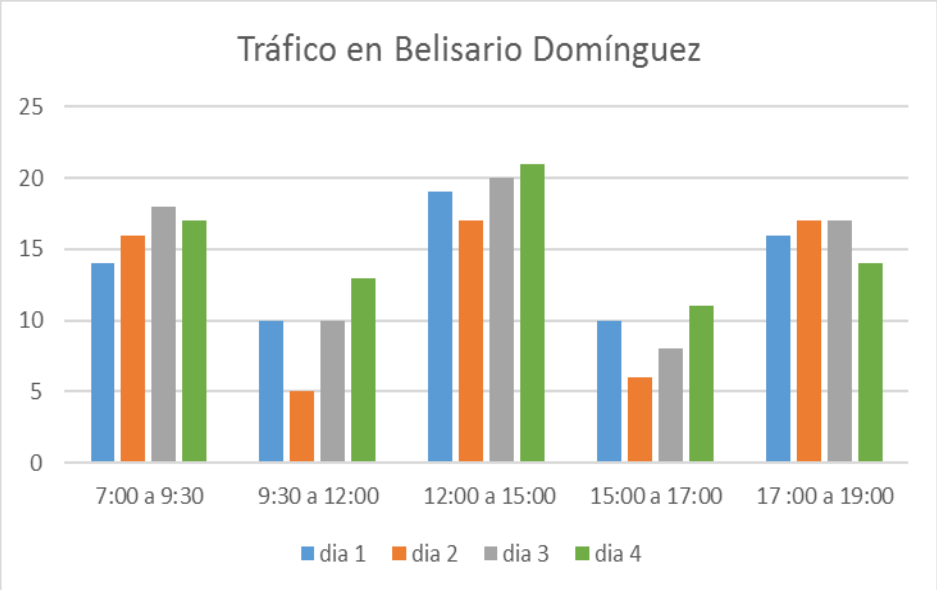


Figura 3. 6: Tráfico vehicular en la calle Belisario Domínguez.

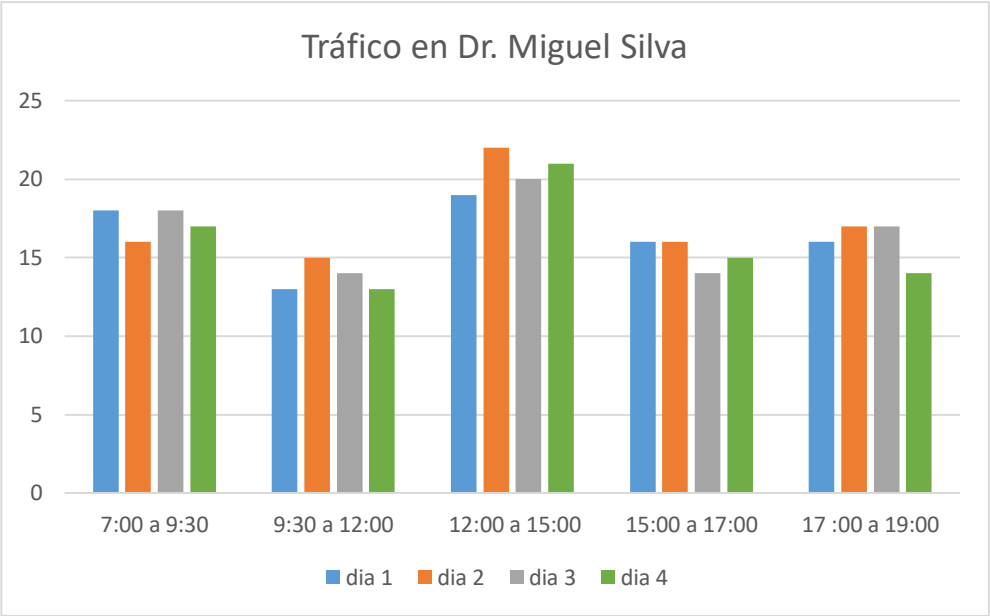


Figura 3. 7: Tráfico vehicular en la calle Dr. Miguel Silva.

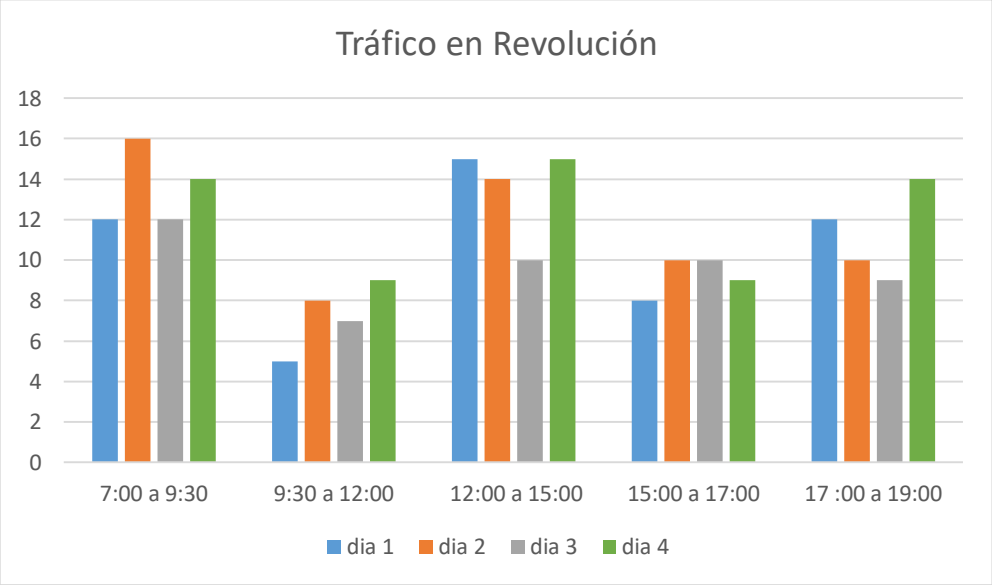


Figura 3. 8: Tráfico vehicular en la calle Revolución.

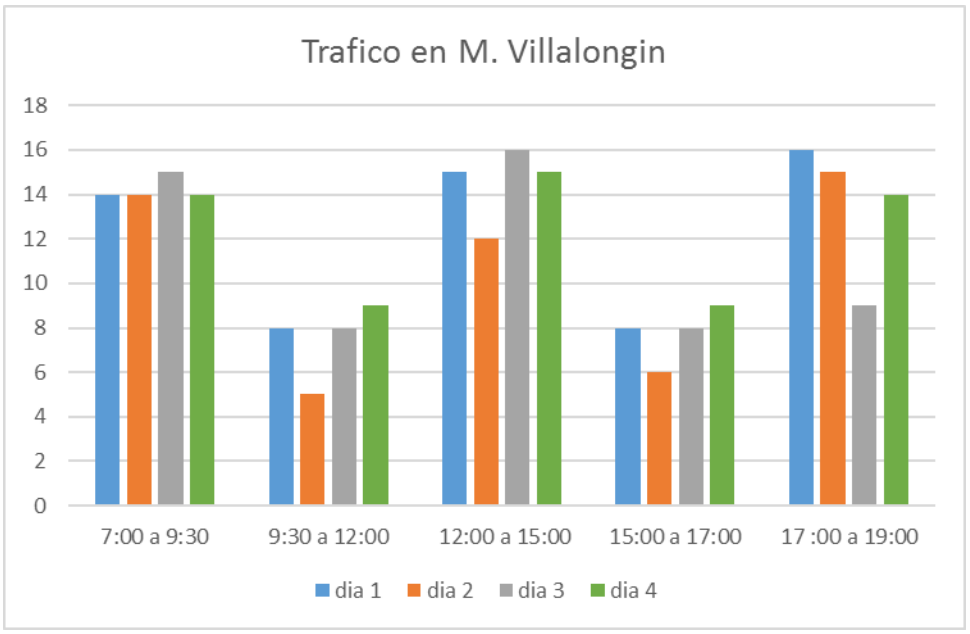


Figura 3. 9:Tráfico vehicular en la calle M. Villalongin.

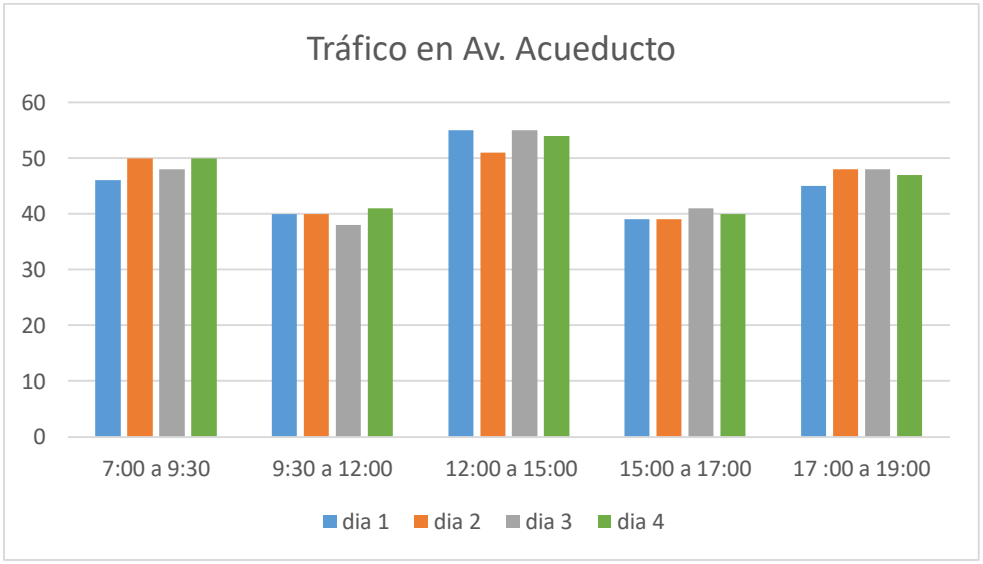


Figura 3. 10: Tráfico vehicular en la Av. Acueducto.

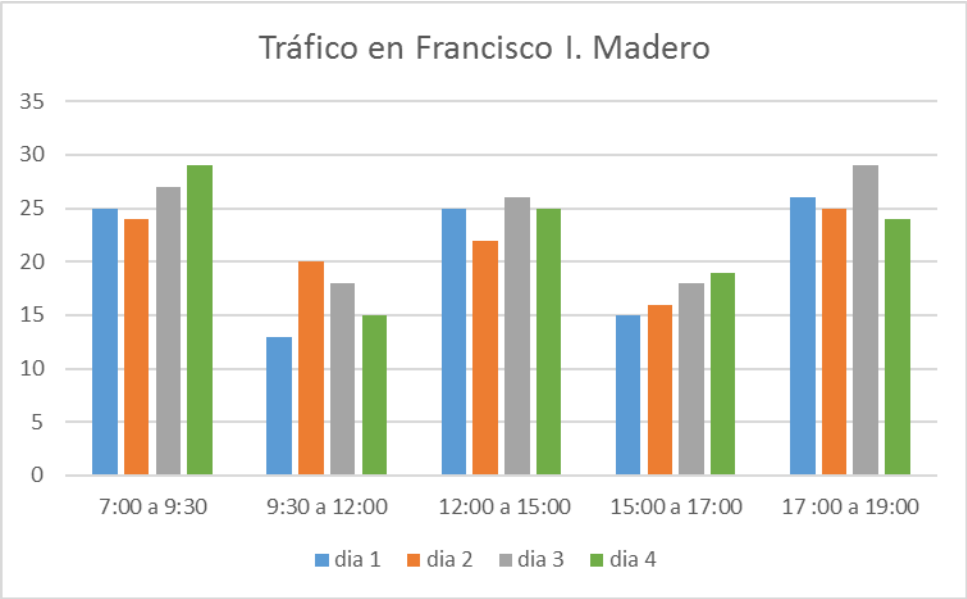


Figura 3. 11. Tráfico vehicular en la calle Francisco I. Madero.

Al realizar la investigación de campo se notó que algunos semáforos no están sincronizados, ya que algunos semáforos se ponen en verde primero o en rojo lo cual no le da mucho desahogue a la hora del tráfico, los semáforos que están ubicados en la avenida Francisco I. Madero con las esquinas Virrey de Mendoza y Vasco de Quiroga se ponen primero en rojo y 2 segundos a 5 segundos después los semáforos de las esquinas Morelos Norte y Belisario Domínguez se ponen en rojo, en el caso que cambian de color verde los cuatro semáforos se ponen casi al mismo tiempo ya que solo están de sincronizados por 2 segundos entre semáforos.

En los semáforos que están ubicados en la calle M. Villalongin hasta la esquina de Dr. Miguel Silva también no están sincronizados esto provoca que los automovilistas que pasan de las avenidas Acueducto y Francisco I. Madero se queden en los semáforos ya que no pasan todos los autos, lo cual provoca que se genere el tráfico. También se notó que una cuarta parte de los carros que provienen de estas avenidas tomen la calle Aquiles Serdán para no tener que lidiar con el tráfico vehicular y ahorrarse un tiempo en cruzar la avenida Madero de la altura de las Tarascas hasta el Mercado de Dulces.

3.4. Implementación

El sistema automatizado del semáforo inteligente estará diseñado para darle una mejor movilidad al tráfico vehicular, en la Figura 3.12, se muestra la avenida donde estará implementado el semáforo automatizado.

Figura 3. 11: Ubicación del sistema de automatización semáforo inteligente.

3.5. Descripción del sistema

El sistema de automatización de semaforización inteligente tiene como objetivo principal controlar el tiempo de las fases de los diferentes semáforos para darle un desahogue al tráfico de vehículos en la vía.

El sistema se pone en marcha con el botón ejecutar en la interfaz, entonces se generan los nuevos tiempos de fases para cada uno de los semáforos, después de generarse los nuevos tiempos los semáforos se recalibrarán, dependiendo en la hora que se active el sistema este emperezará a funcionar en horas normal o en la hora pico, en donde las horas normales es cuando no hay tanto tráfico vehicular y los tiempos de los semáforos será un poco más corto que cuando sea hora pico, los tiempos en las horas pico será más duradero debido a que el trafico aumenta y será necesario darle un mayor desahogue al tráfico vehicular.

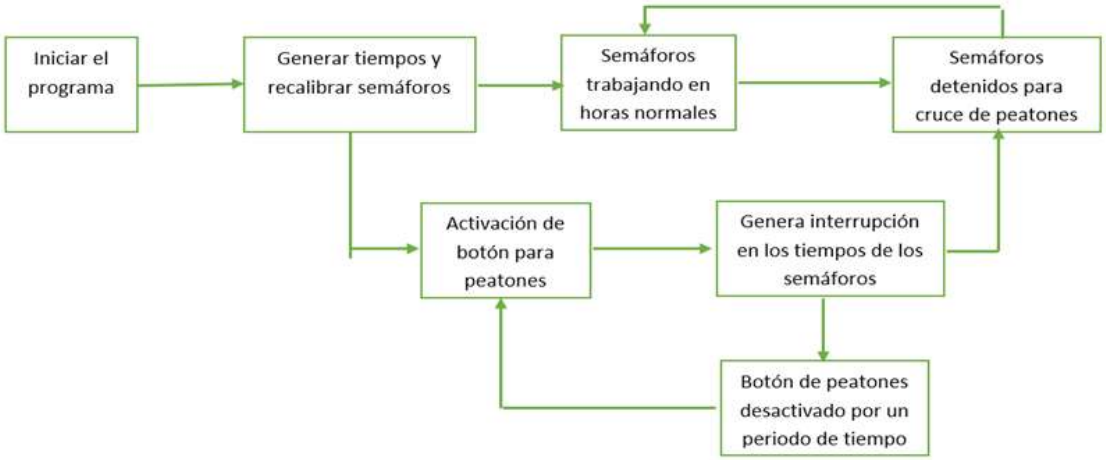


Figura 3. 12: Diagrama de bloques del sistema horas normales.

El PLC cuanta con instrucciones de reloj y calendario lo cual facilita al sistema de automatización de semaforización inteligente ya que puede programarse el calendario lo que permite que la programación hecha dure todo el año, el sistema por si solo puede cambiar de un modo a otro a las instrucciones de reloj que se puede programar que a cierta hora empieza las horas pico y terminan a cierta hora.

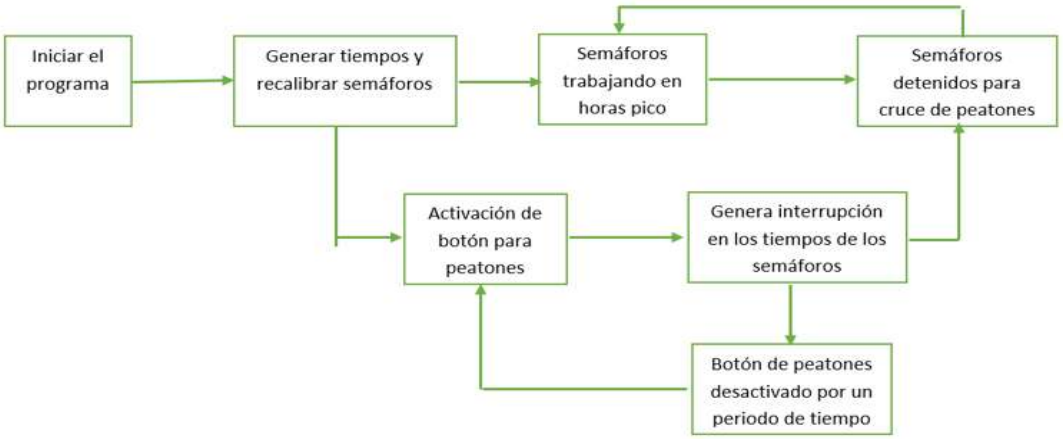


Figura 3. 13: Diagrama de bloques del sistema horas pico.

Para los peatones que circulen en la vía entre los semáforos existirá un botón el cual dará una indicación la cual se interrumpirá la secuencia de los semáforos para darle pase a los peatones, el tiempo que se dará para el cruce de los peatones será de 10 segundos, después de ese tiempo se volverá a iniciar la secuencia de los semáforos en donde se quedó, una vez activado el botón de peatones este no podrá ser activado hasta cierto tiempo después, en el tiempo que el botón no este activo aunque se vuelva a presionar este no hará nada debido a que si se está presionando el botón se estaría interrumpiendo muy frecuentemente la secuencia de los semáforos y esto causaría más tráfico.

En caso de que sea un día especial como son los domingos que es cuando se cierra la avenida Francisco I. Madero desde la altura de las Tarascas hasta la esquina de la calle Cuautla, se les dará más tiempo a los semáforos secundarios para que se desahogue el tráfico, otro día especial es el sábado por las noches, los sábados por las noches se cierra la avenida Francisco I. Madero para que la gente pueda caminar por la avenida desde la esquina de Virrey de Mendoza hasta la esquina de la calle El Nigromante.

Para estos casos se decidió hacer un subsistema para activarlos estos días lo cual podría aplicarse en otro día de la semana que puede haber algún evento no planeado lo cual generan mucho tráfico y congestionan gran parte de las zonas aledañas de donde se produce o se genera dicho evento.

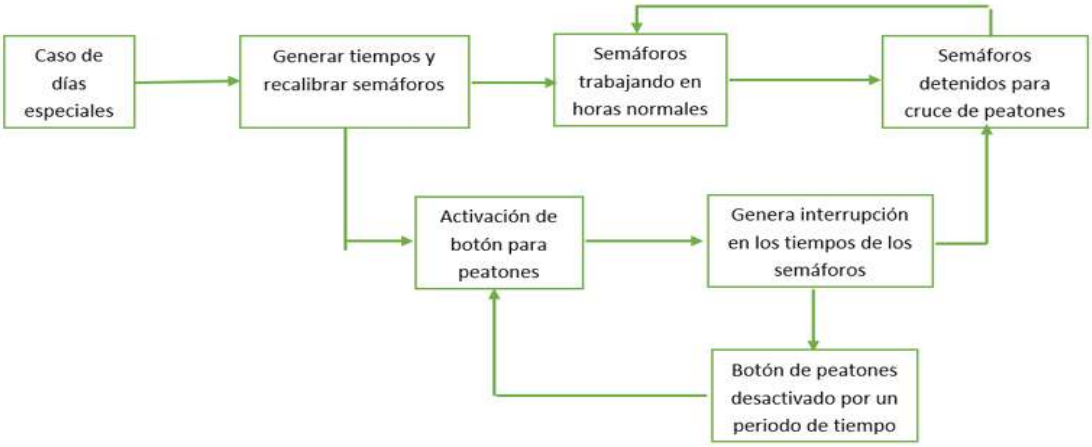


Figura 3. 14: Diagrama de bloques del sistema día especial.

Para los casos especiales, se generó una secuencia especial, debido a que no se sabe cuándo podrá haber una manifestación, una toma, algún evento de exhibición o concierto, algún percance automotriz o civil, donde cualquiera de estas circunstancia genera tráfico vehicular, en la zona y en las zonas aledañas se genera este evento el cual solo será activado por un usuario el cual detendrá la secuencia de modo automático y pasarlo a modo manual esta secuencia durará el tiempo que tome esta situación de caso especial, una vez terminada la situación el usuario pondrá en marcha el modo automático y este automáticamente reconocerá el día y hora en el que está y procederá a cambiar a los casos de hora pico, hora normal o día especial.

Figura 3. 15: Diagrama de bloques del sistema caso especial.

Capítulo 4 Descripción del software

4.1. Vista del portal

El Totally Integrated Automation Portal (TIA Portal) ofrece dos vistas diferentes de las herramientas disponibles: distintos portales orientados a tareas organizados según las funciones de las herramientas (vista del portal) e la Figura 4.1, o una vista orientada a los elementos del proyecto (vista del proyecto) en la Figura 4.2 [9].

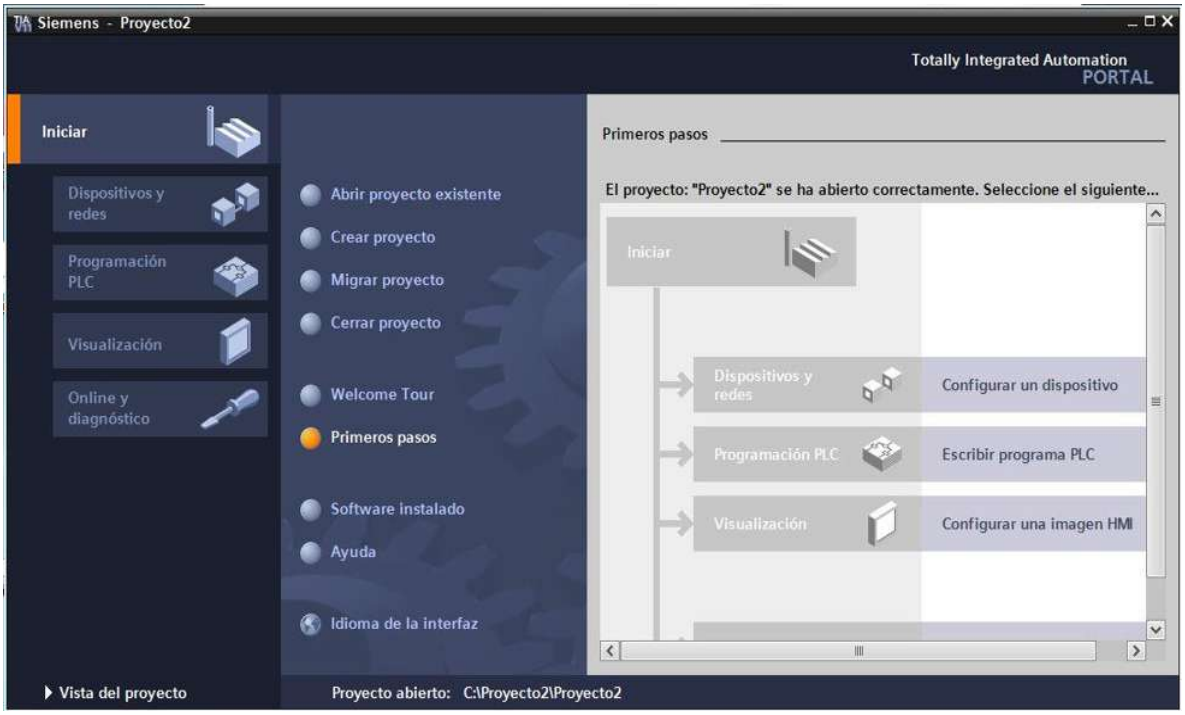


Figura 4. 1: Vista TIA Portal.

En la vista del portal se muestran las siguientes funciones [9]:

- **Portales para diferentes tareas:** ofrecen los tipos de acciones de configuración de los componentes.
- **Tareas del portal seleccionado:** ofrecen los objetivos específicos de la configuración seleccionada.
- **Panel de selección para la acción seleccionada:** muestra el contenido de la ventana de la acción a realizar
- **Cambia a la vista del proyecto:** es un acceso directo a la ventana del proyecto.

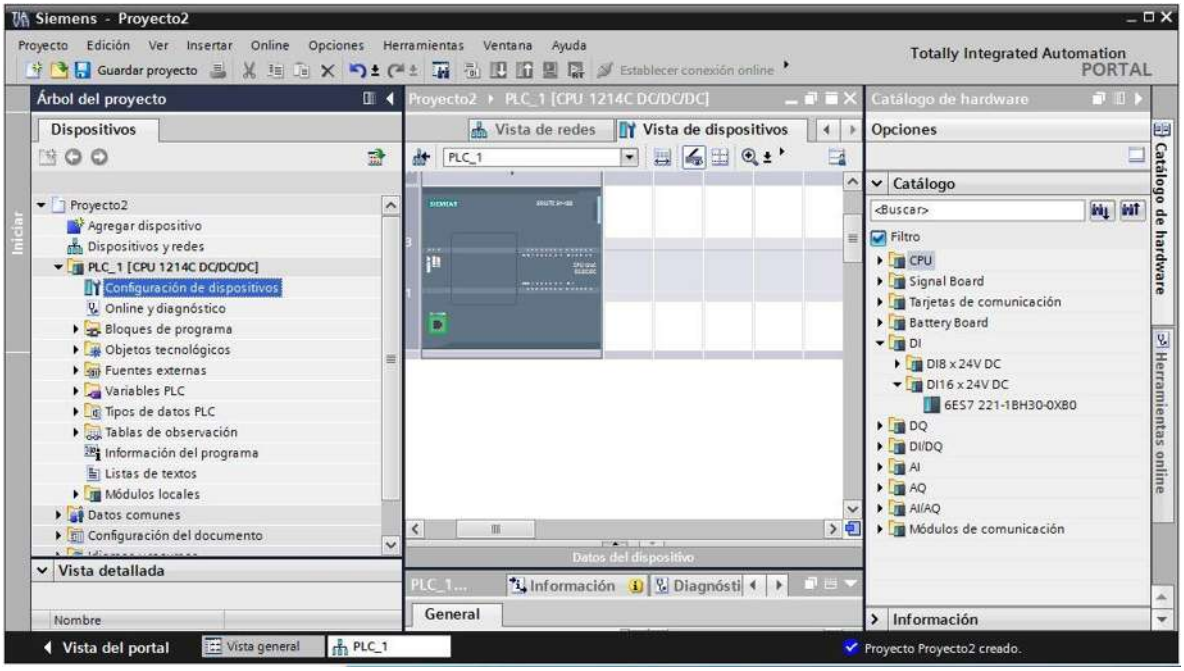


Figura 4. 2: Vista del proyecto.

En la vista del proyecto se muestran las siguientes funciones [9]:

- **Menús y barra de herramientas:** ofrecen las funciones más utilizadas para el trabajo del software y herramientas a utilizar.
- **Árbol del proyecto:** muestra los proyectos abiertos, como es posible agregar bloques, variables, configuración del autómata y agregar expansiones del autómata.
- **Área de trabajo:** se visualiza la programación realizada
- **Task cards:** ofrece la vista de las librerías de los componentes a utilizar.
- **Ventana de inspección:** muestra las propiedades e información acerca del objeto seleccionado en el área de trabajo.
- **Cambia a la vista del portal:** nos permite regresar a la vista del portal.
- **Barra de editor:** agiliza el trabajo y mejora la eficiencia, ya que nos muestra todos los editores que están abiertos.

4.2. PLCSIM V13

Es el Software encargado de la simulación del PLC, este software permite simular las variables creadas en el PLC, puede también controlar las variables creadas en el autómata.

4.2.1. Vista principal del PLCSIM V13

El PLCSIM V13 es fácil de manejar ya que posee un menú básico para poderlo manejar de manera sencilla, cómoda y entendible. En la Figura 4.3 se muestra la vista principal del PLCSIM V13 [9].

La simulación del PLCSIM V13 consiste en la visualización de las variables, las cuales se tiene el poder de manipularlas para que cambie de estado 0 a 1 o de 1 a 0, como se muestra en la Figura 4.4.

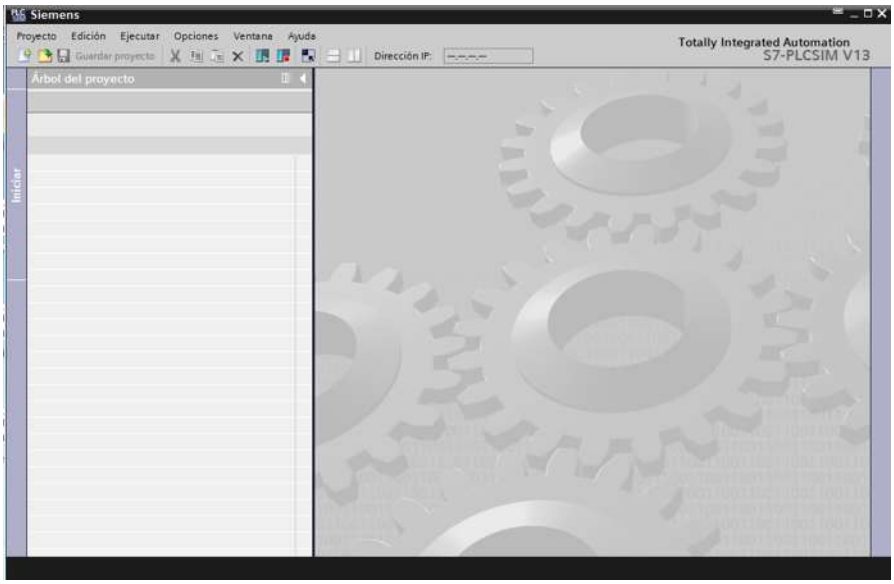


Figura 4. 3: Vista general PLCSIM V13.

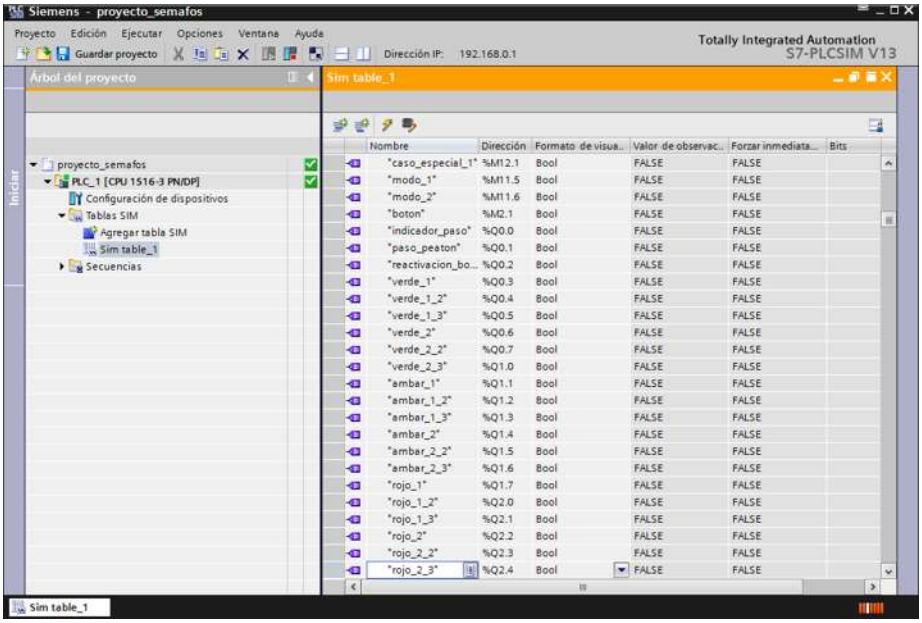


Figura 4. 4: Vista del proyecto PLCSIM V13.

4.3. Configuración de la comunicación HMI

Para la pantalla HMI hay que revisar que esté funcionando correctamente y esté haciendo comunicación con el PLC, hay que verificar la comunicación que se está teniendo para esto es necesario seguir los siguientes pasos:

- En Windows es necesario ir al apartado “inicio > panel de control”.
- Una vez dentro de panel de control donde dice “ver por: categorías” seleccionamos la opción “iconos grandes”.
- Todas las opciones que estaban en las diferentes rutas de acceso se desplegaran por opción, allí se busca la opción “ajustar interface PG/PC (32 bits)”, Figura 4.5.
- Una vez dentro de “ajustar interface PG/PC (32 bits)”, la opción que aparece por defecto es “PLCSIM.TCPIP.1”.
- Hay que buscar la opción “PLCSIM S7-1200/S7-1500.TCPIP.1” y aceptar, Figura 4.6, con esta opción se garantiza que se tenga comunicación entre la PC a PLC, con esto el PLC tendrá comunicación con la pantalla HMI.

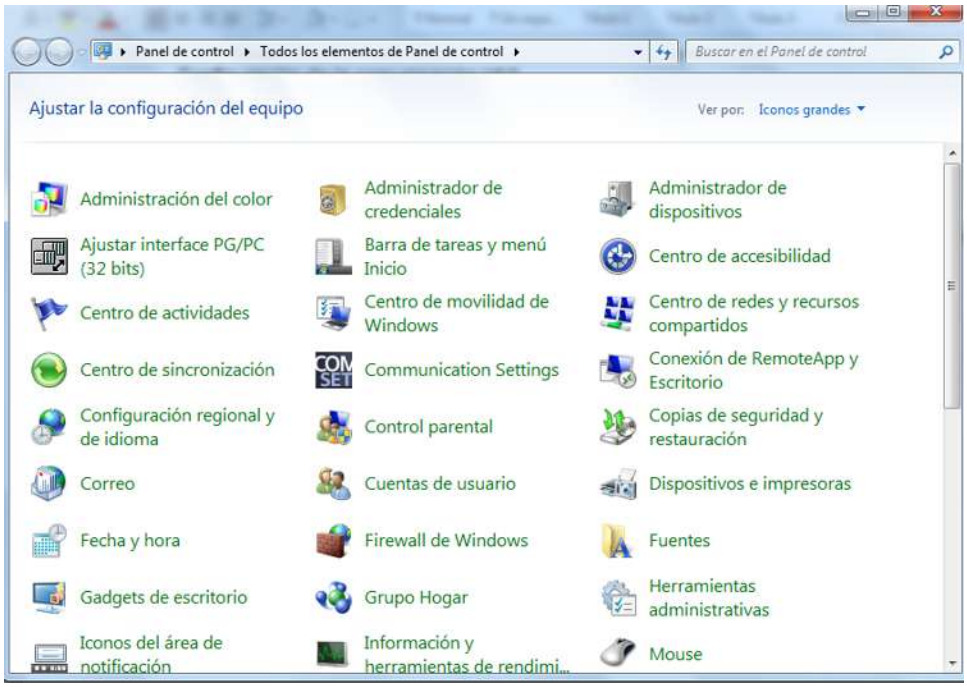


Figura 4. 5: Panel de control.

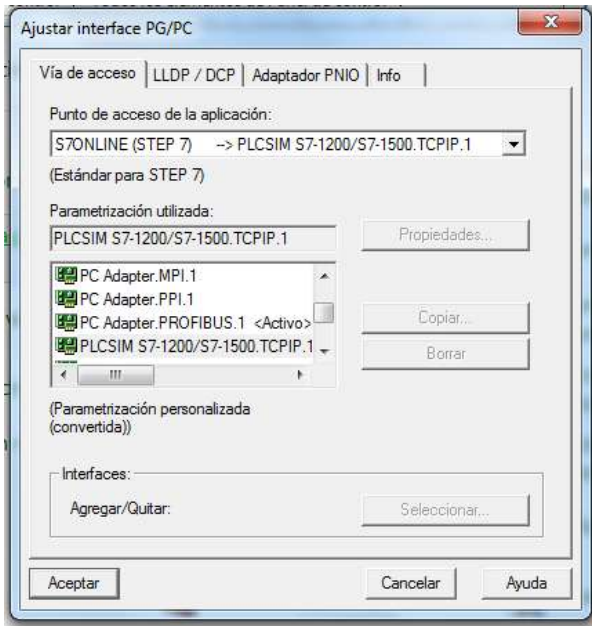


Figura 4. 6: Comunicación PLC-HMI "PLCSIM S7-1200/S7-1500.TCPIP.1".

Capítulo 5 Implementación semáforo inteligente

5.1. Estructura semáforos inteligentes

Tomando como ejemplo la Figura 5.1, se realizó la automatización de los semáforos inteligentes en la cual se implementó dos modos de funcionamiento, el primero para el modo automático, el cual con las diferentes secuencias detecta fecha y hora en las que debe funcionar entre las secuencias de hora pico, hora normal y días especiales. El modo manual, solo se selecciona cuando hay un caso especial, los cuales pueden ser marchas, desfiles, eventos y tomas. Al seleccionar el modo manual automáticamente el modo automático deja de funcionar.

Figura 5. 1: Semáforo.

5.2. Bloque de datos (DB)

Lo que se quiere evitar en un principio es que el programa genere más memoria, por lo cual es necesario utilizar los bloques de datos DB (Apéndice A.1), debido a que se utilizarán varios temporizadores los cuales al agregarlos se pueden crear como DB instancias, al utilizar varios DB instancia se estará ocupando memorias diferentes en el programa, en la Figura 5.2 se muestra la diferencia entre DB globales y DB instancias.

Un bloque de datos globales (DB), sirve para almacenar valores que se escribe durante la ejecución del programa.

Al contrario de los bloques lógicos, los DB, solo contienes declaraciones de variables, no contienen segmentos ni instrucciones, las declaraciones de variables definen la estructura del bloque de datos.

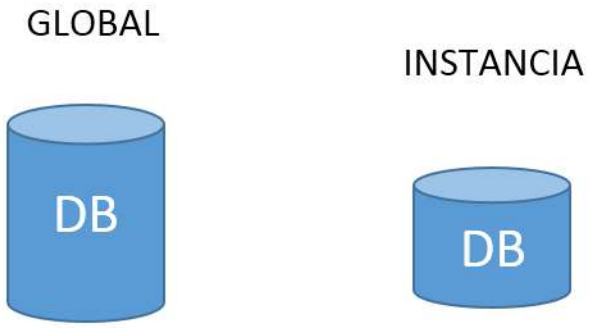


Figura 5. 2: Memoria de bloque de datos (DB) global e instancia.

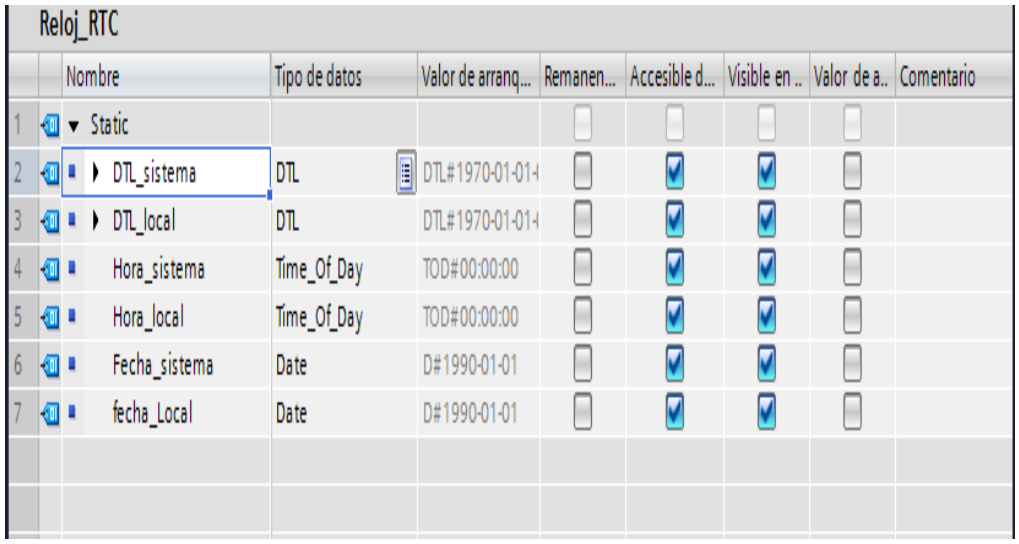
- **Un DB global** almacena los datos de los bloques lógicos en el programa. Cualquier OB, FB o FC puede acceder a los datos en un DB global.
- **Un DB instancia** almacena los datos de un FB especifico. La estructura de los datos en un DB instancia refleja los parámetros (Input, Output e Inout) y los datos estáticos del FB. (la memoria temporal del FB no se almacena en el DB instancia).

En el programa se crearon dos DB globales, una para para guardar las diferentes variables para luego ser llamadas en el programa.

En la DB global de reloj_RTC se crearon las variables donde se guardarán los datos DTL (Apéndice A.2.4.), time_of_day y date.

- El tipo de datos DTL es una estructura de datos de 12 bytes que almacena información de fecha y hora en una estructura predefinida. Un DTL se puede definir en la memoria temporal del bloque o en un DB.

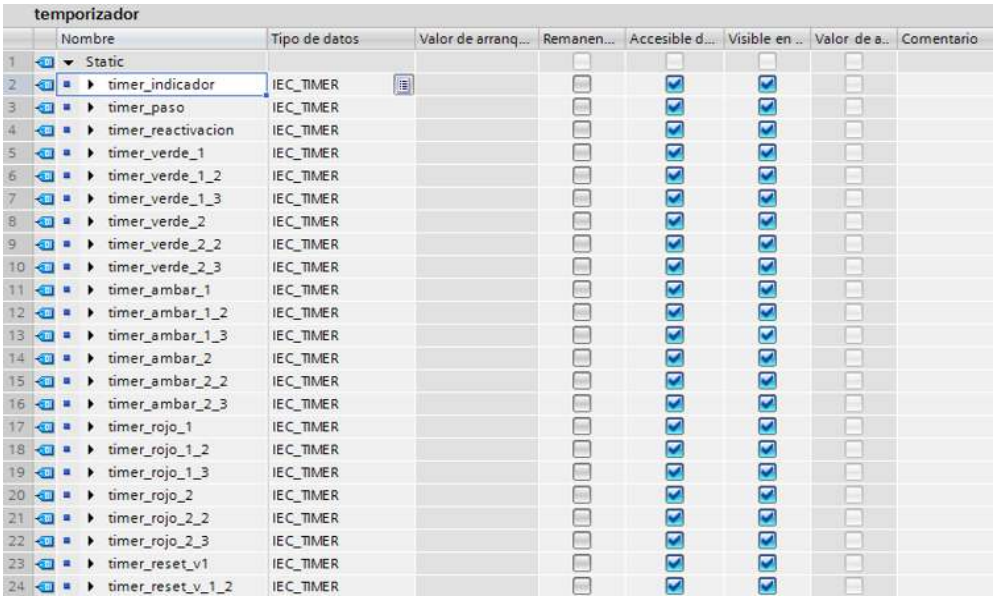
En la DB global de temporizador se crearon las variables donde se guardarán los datos IEC_TIMER, el programa TIA Portal al momento de agregar un temporizador asigna su nombre para los temporizadores IEC, timer para saber que los datos que se utilizarán será de tiempo.



Reloj_RTC

	Nombre	Tipo de datos	Valor de arranq...	Remanen...	Accesible d...	Visible en ...	Valor de a...	Comentario
1	Static							
2	DTL_sistema	DTL	DTL#1970-01-01-(					
3	DTL_local	DTL	DTL#1970-01-01-(					
4	Hora_sistema	Time_Of_Day	TOD#00:00:00					
5	Hora_local	Time_Of_Day	TOD#00:00:00					
6	Fecha_sistema	Date	D#1990-01-01					
7	fecha_Local	Date	D#1990-01-01					

Figura 5. 3: Vista del DB global Reloj_RTC.



temporizador

	Nombre	Tipo de datos	Valor de arranq...	Remanen...	Accesible d...	Visible en ...	Valor de a...	Comentario
1	Static							
2	timer_indicador	IEC_TIMER						
3	timer_paso	IEC_TIMER						
4	timer_reactivacion	IEC_TIMER						
5	timer_verde_1	IEC_TIMER						
6	timer_verde_1_2	IEC_TIMER						
7	timer_verde_1_3	IEC_TIMER						
8	timer_verde_2	IEC_TIMER						
9	timer_verde_2_2	IEC_TIMER						
10	timer_verde_2_3	IEC_TIMER						
11	timer_ambar_1	IEC_TIMER						
12	timer_ambar_1_2	IEC_TIMER						
13	timer_ambar_1_3	IEC_TIMER						
14	timer_ambar_2	IEC_TIMER						
15	timer_ambar_2_2	IEC_TIMER						
16	timer_ambar_2_3	IEC_TIMER						
17	timer_rojo_1	IEC_TIMER						
18	timer_rojo_1_2	IEC_TIMER						
19	timer_rojo_1_3	IEC_TIMER						
20	timer_rojo_2	IEC_TIMER						
21	timer_rojo_2_2	IEC_TIMER						
22	timer_rojo_2_3	IEC_TIMER						
23	timer_reset_v1	IEC_TIMER						
24	timer_reset_v_1_2	IEC_TIMER						

Figura 5. 4: Vista del DB global temporizador.

5.3. IMPLEMENTACION CON DIAGRAMAS DE ESCALERAS

Para la implementación con diagramas de escaleras y el funcionamiento del programa, una vez que el programa se ha ejecutado no hará nada ya que para el

funcionamiento se tienen que elegir entre dos modos, el primero es el modo automático el cual funciona mediante la detección de hora y fecha, una vez obtenida hora y fecha local está la comprueba para ver en que secuencia está funcionando de las cuales puede trabajar en hora normal, hora pico y días especiales. El modo dos es el modo manual y esta solo se puede activar cuando se tengan casos especiales como tomas, manifestaciones y eventos.

5.3.1 Detección de hora y fecha

La CPU soporta un reloj en tiempo real. La función de esta parte es la de obtener una lectura de la hora y fecha del sistema y local, se dispone de instrucciones que permiten leer la hora del sistema (RD_SYS_T, Apéndice A.9.) y la hora local (RD_LOC_T, Apéndice A.9.), la hora local se calcula utilizando la diferencia con respecto a la zona horaria y al horario de verano que se ajusta en la configuración "configuración de dispositivos" del reloj del PLC.

En la Figura 5.5 se muestra la implementación.

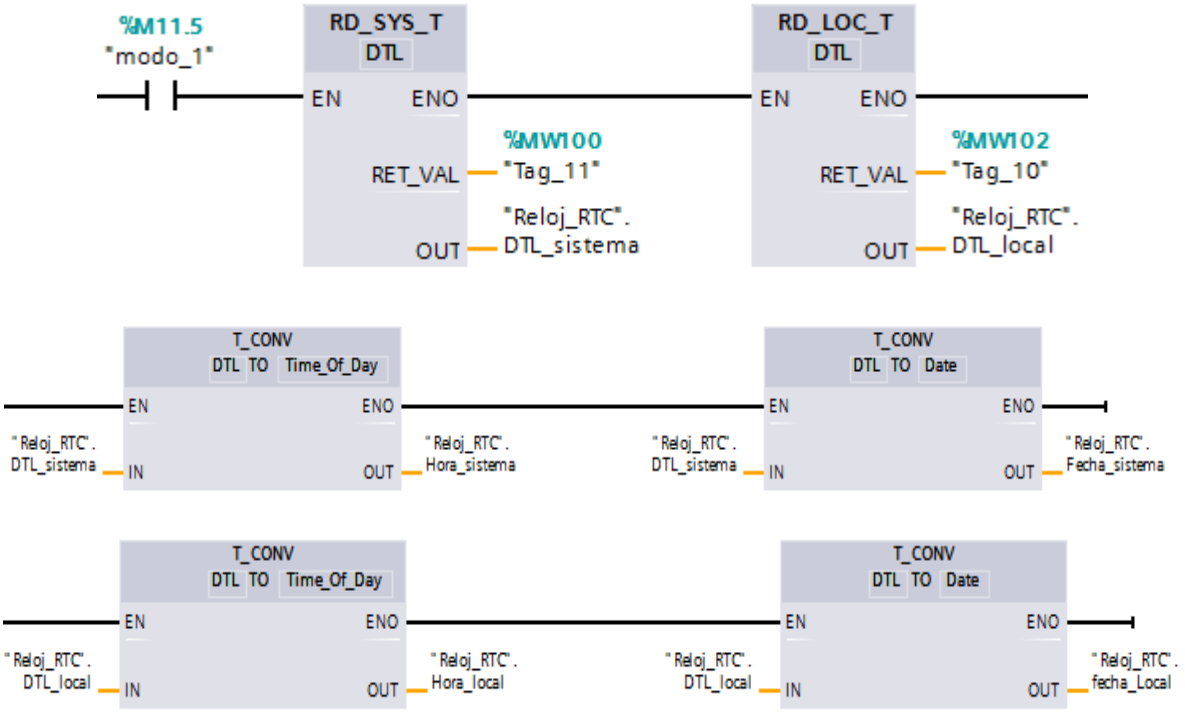


Figura 5. 5: Implementación de modo automático y detección de fecha y hora.

5.3.2. Detección de fecha y hora de secuencias

Una vez obtenido la hora y fecha local, se prosigue a detectar mediante comparadores (Apéndice A.8.), el día de la semana y la hora en la cual se va a trabajar, entre la secuencia en las que se puede trabajar son hora normal, hora pico y día especial, los días que entra la secuencia de días especiales son los sábados a partir de las 19:30 a las 23:59 y los domingos a partir de las 07:00 a las 13:00, los demás días y horarios cambian de hora normal a hora pico.

En la Figura 5.6 se muestra la implementación del día domingo.

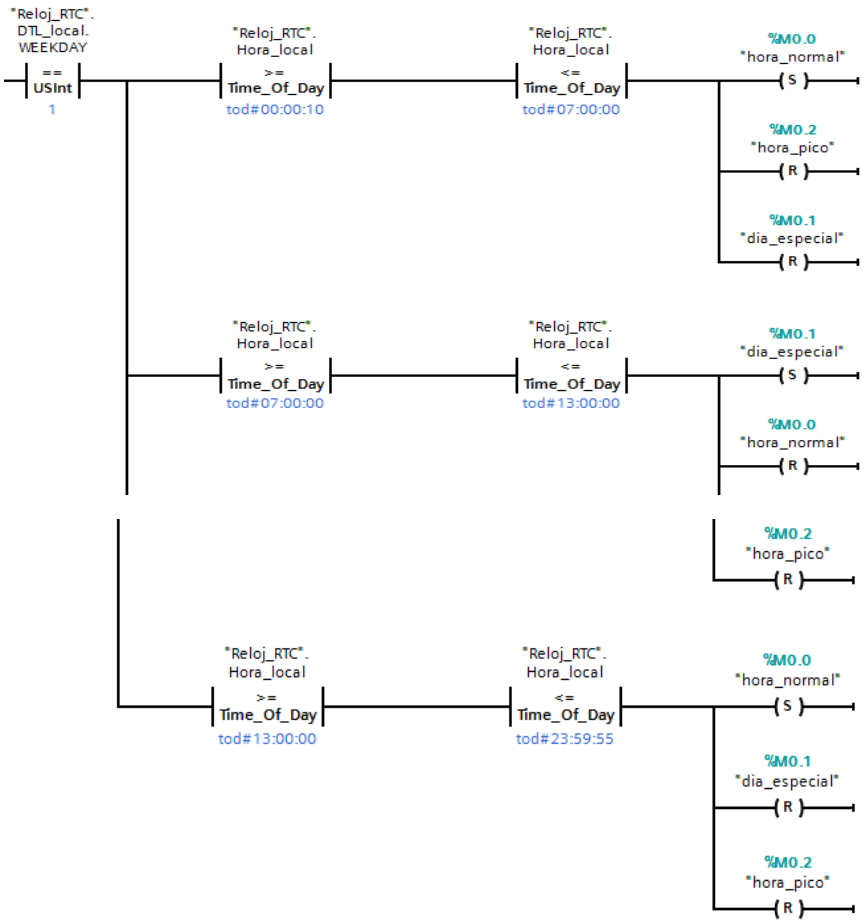


Figura 5. 6: Implementación de día domingo.

En las Figuras 5.7 y 5.8 se muestran la implementación de los días lunes a viernes.

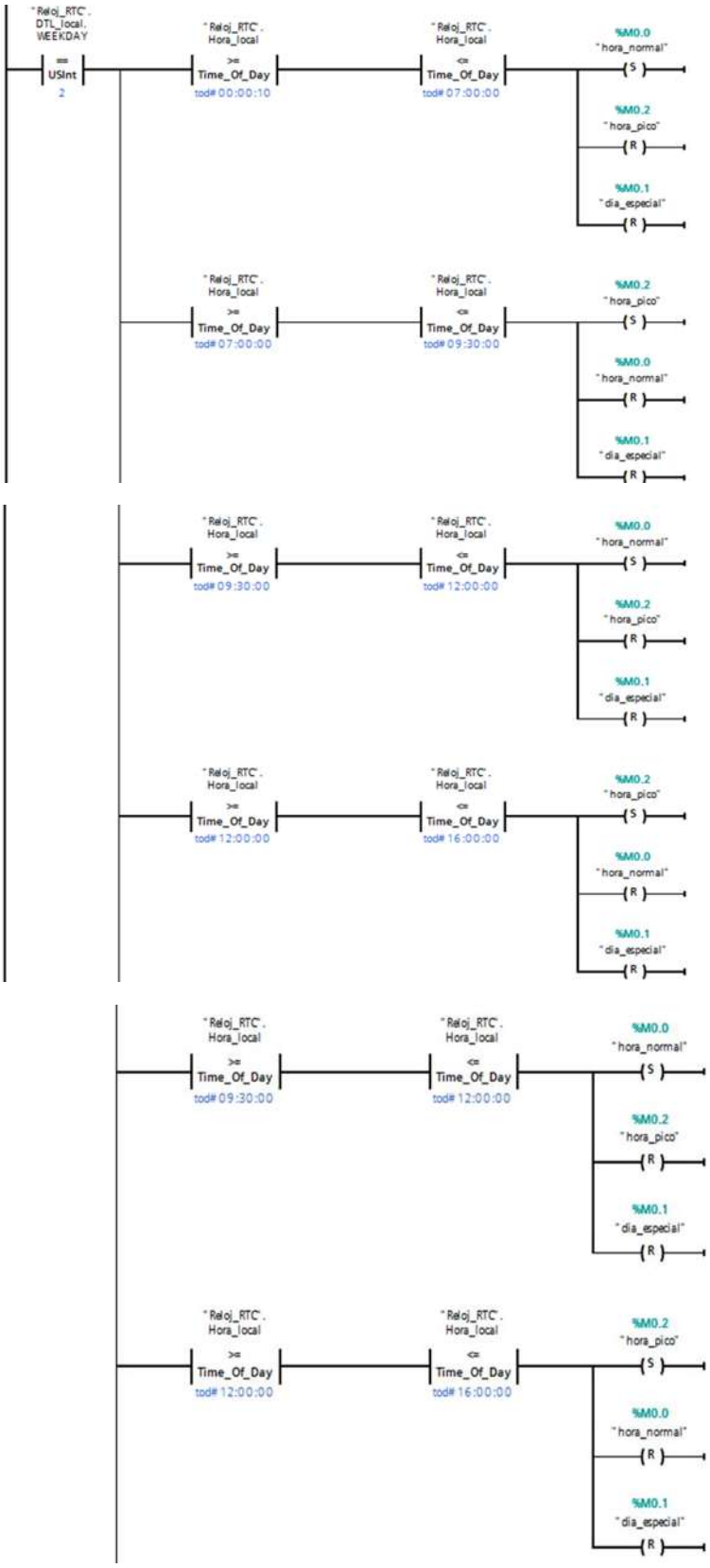


Figura 5. 7: Implementación de detección de días lunes a viernes.

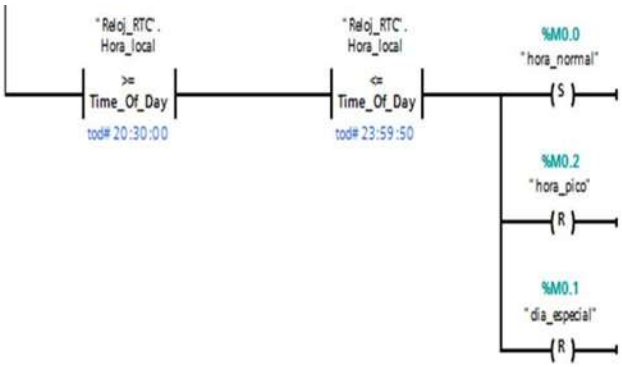


Figura 5. 8: Implementación de detección de días lunes a viernes.

En las Figuras 5.9 y 5.10 se muestra la implementación del día sábado.

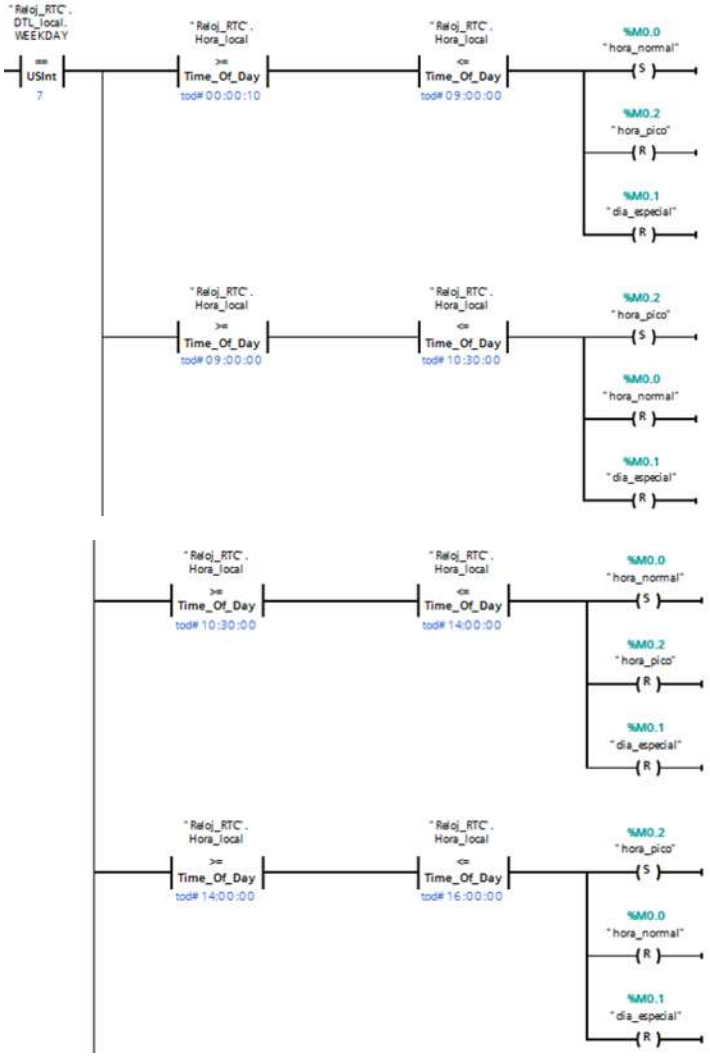


Figura 5. 9: Implementación de detección de día sábado.

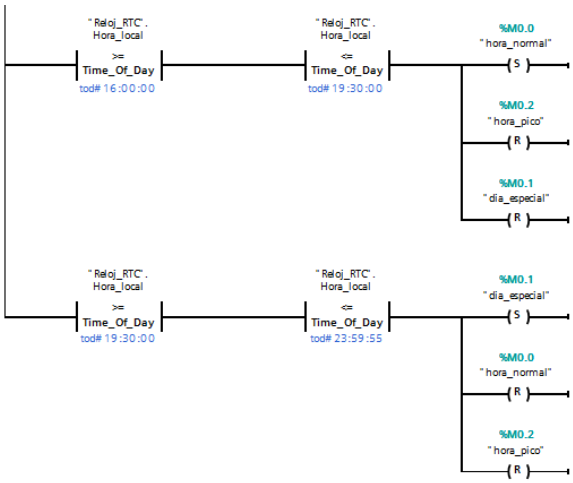


Figura 5. 10: Implementación de detección de día sábado.

5.3.3. Paso para los peatones

El objetivo de esta parte es que el peatón tenga el control del pase peatonal sin estar interrumpiendo frecuentemente el flujo de los automóviles, por este motivo el paso de los peatones solo puede ser activado cada cierto tiempo, para esto se utilizó un flipflop de tipo SR (Apéndice A.6.2.), sin importar cuantas veces se presione el botón para el paso peatonal este no hará nada hasta que se activa el indicador de reactivación.

En la Figura 5.11 y 5.12 se muestra la implementación.

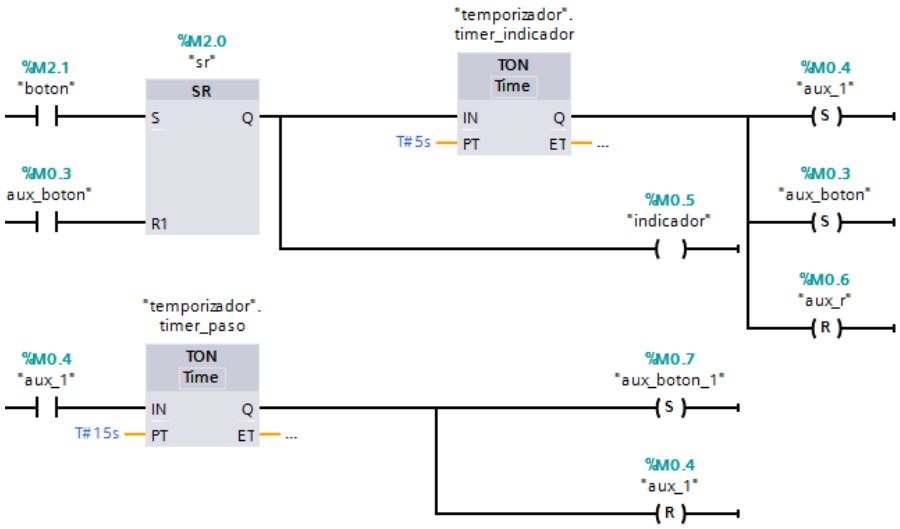


Figura 5. 11: Implementación de secuencia para el paso de los peatones.

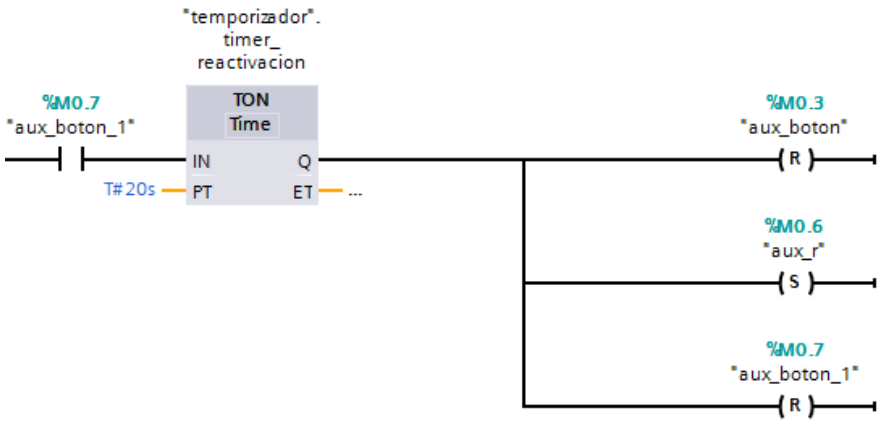


Figura 5. 12: Implementación de secuencia para el paso de los peatones.

En la Figura 5.13 se muestra la implementación para la visualización de los indicadores del paso para los peatones.

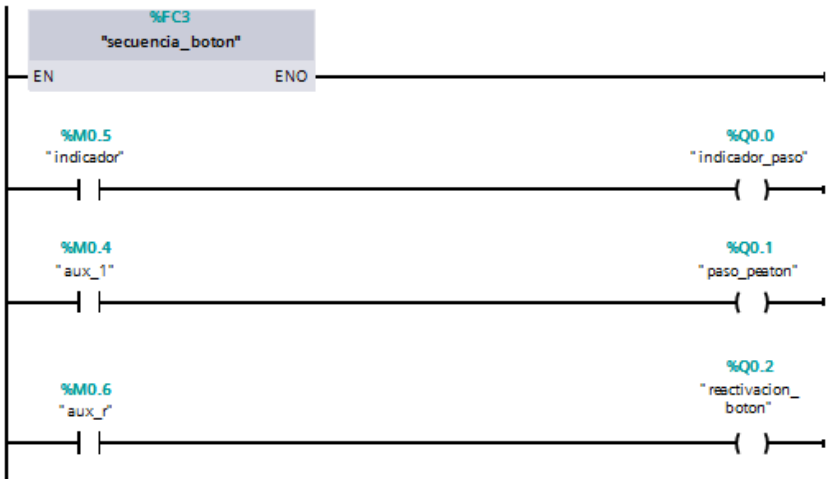


Figura 5. 13: Visualización para el paso de los peatones.

5.3.4. Secuencia de hora normal y hora pico

El objetivo de estas dos secuencias es la de poder dar mejor flujo al tráfico vehicular y así se tenga mejor circulación vial, para dar un mejor fluidez a la aglomeración, se decidió que los semáforos estén de sincronizados ya que así los automovilistas podrán transitar la avenida Madero en un tiempo más corto, para esta parte se utilizaron los temporizadores tipo TONR(apéndice A.7.), el cual tiene la ventaja sobre el TON(apéndice A.7.), ya que si se le interrumpe este se detiene y cuando se quita la interrupción este empezara desde el valor de tiempo acumulado. Las

bobinas set/reset (apéndice A.5) manda la señal de activación a los temporizadores de los semáforos.

En la Figura 5.14 se muestra la implementación.

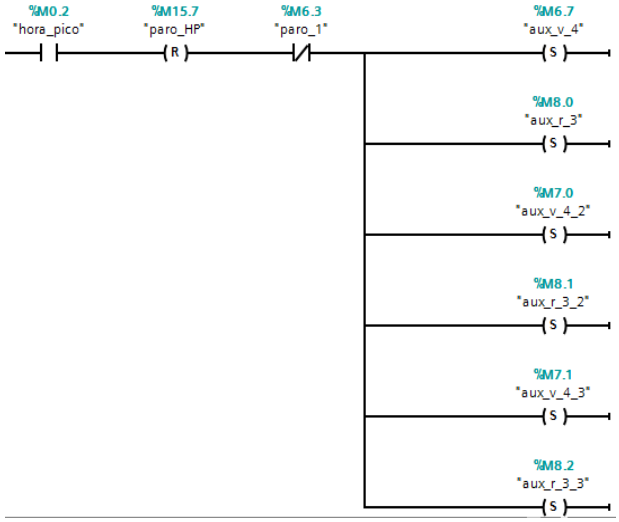


Figura 5. 14: Implementación de activadores secuencia hora pico.

En las Figuras 5.15 a la 5.17 se muestra la implementación de la avenida Francisco I. Madero como principal.

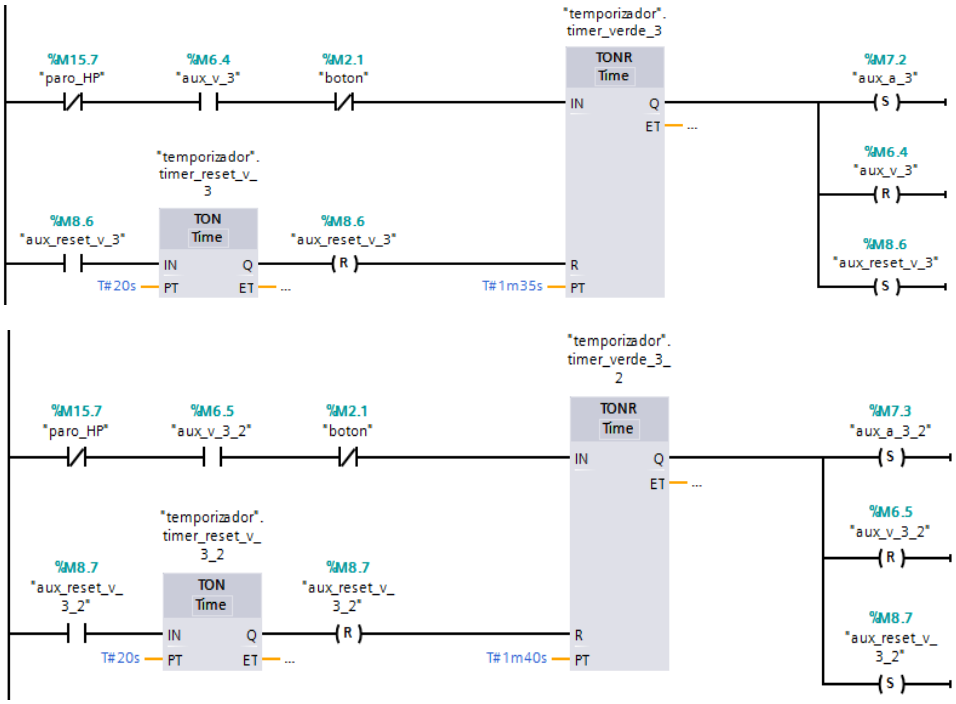


Figura 5. 15: Implementación de secuencia hora pico, semáforos principales.

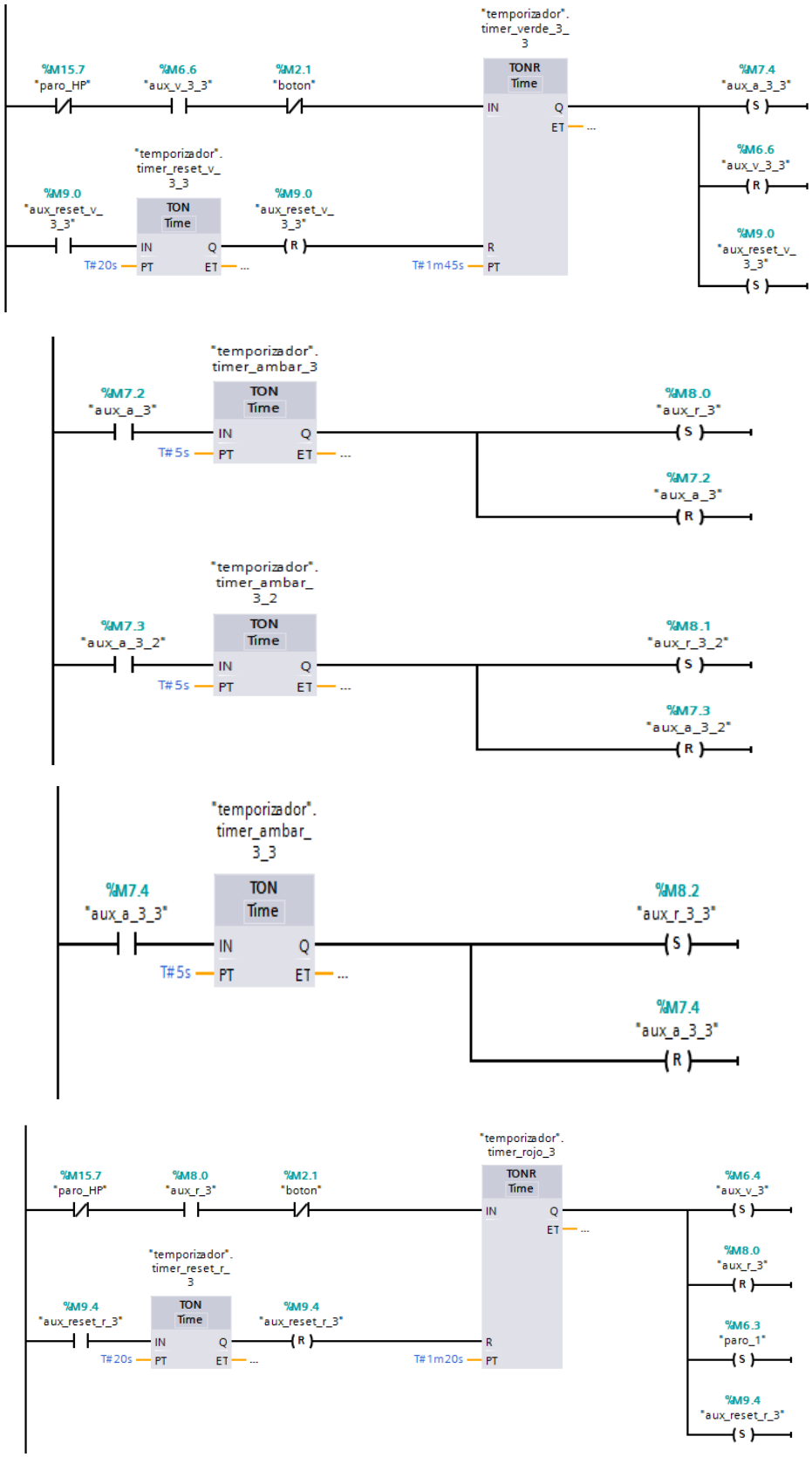


Figura 5. 16: Implementación de secuencia hora pico, semáforos principales.

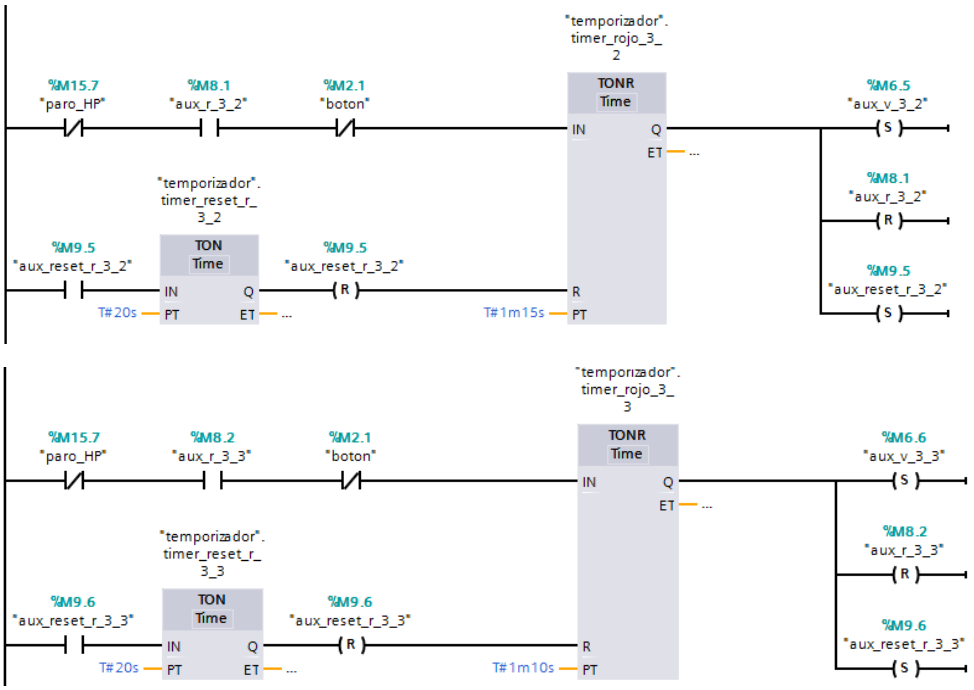


Figura 5. 17: Implementación de secuencia hora pico, semáforos principales.

En las Figuras 5.18 a la 5.21 se muestra la implementación de las calles aledañas como secundarios.

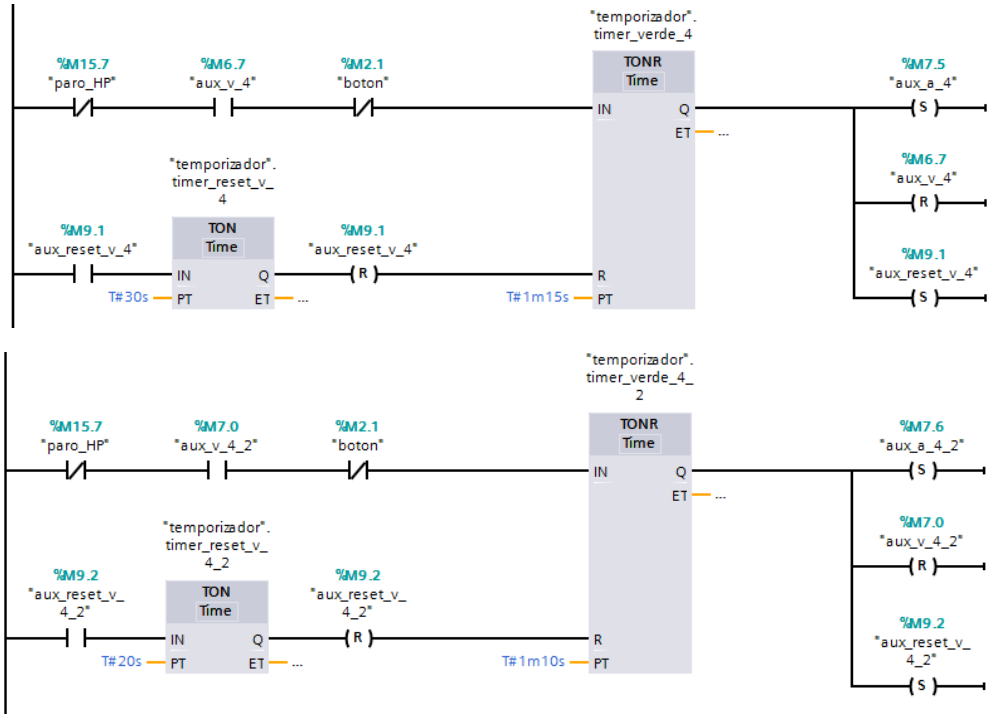


Figura 5. 18: Implementación de secuencia hora pico, semáforos secundarios.

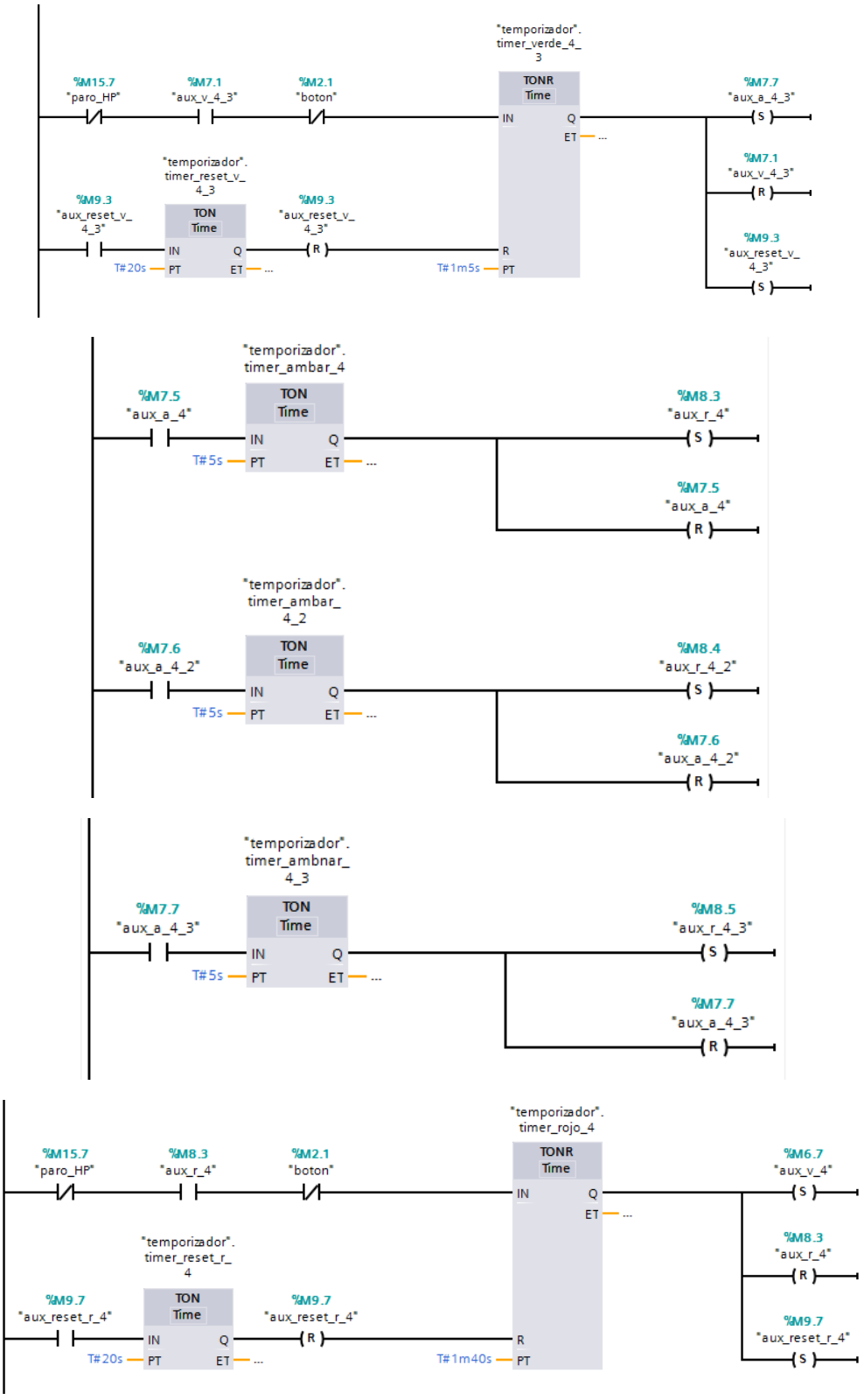


Figura 5. 19: Implementación de secuencia hora pico, semáforos secundarios.

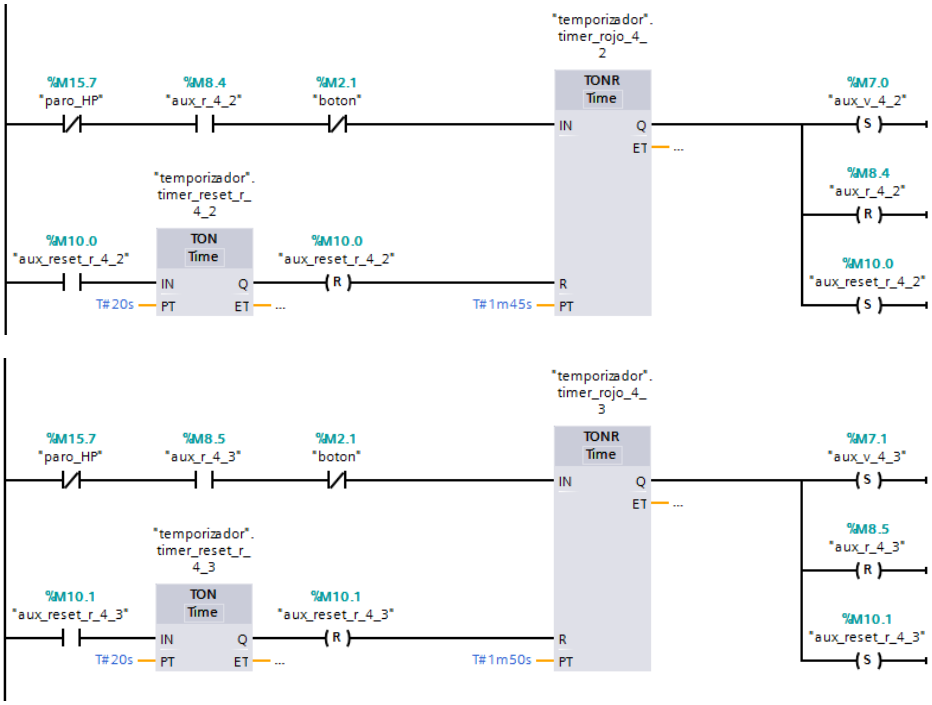


Figura 5. 20: Implementación de secuencia hora pico, semáforos secundarios.

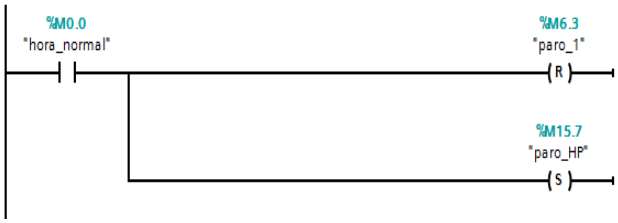


Figura 5. 21: Implementación para el paro de secuencia hora pico.

En las secuencias de hora normal y hora pico lo que cambia son los tiempos que se le dan a los semáforos principal y secundarios, los tiempos en la secuencia de hora pico fueron determinados mediante el estudio que se realizó sobre la situación del tráfico en la zona en la ciudad de Morelia.

El último segmento agregado se realizó pensando en el paro de la secuencia cuando este cambie de hora normal en hora pico y viceversa, si no se realiza este proceso este seguiría funcionando a pesar de que se haya cambiado de secuencia lo cual va a provocar problemas ya que no se sabría si estaría funcionando bien.

En las Figuras 5.22 y 5.23 se muestra la implementación para la visualización.

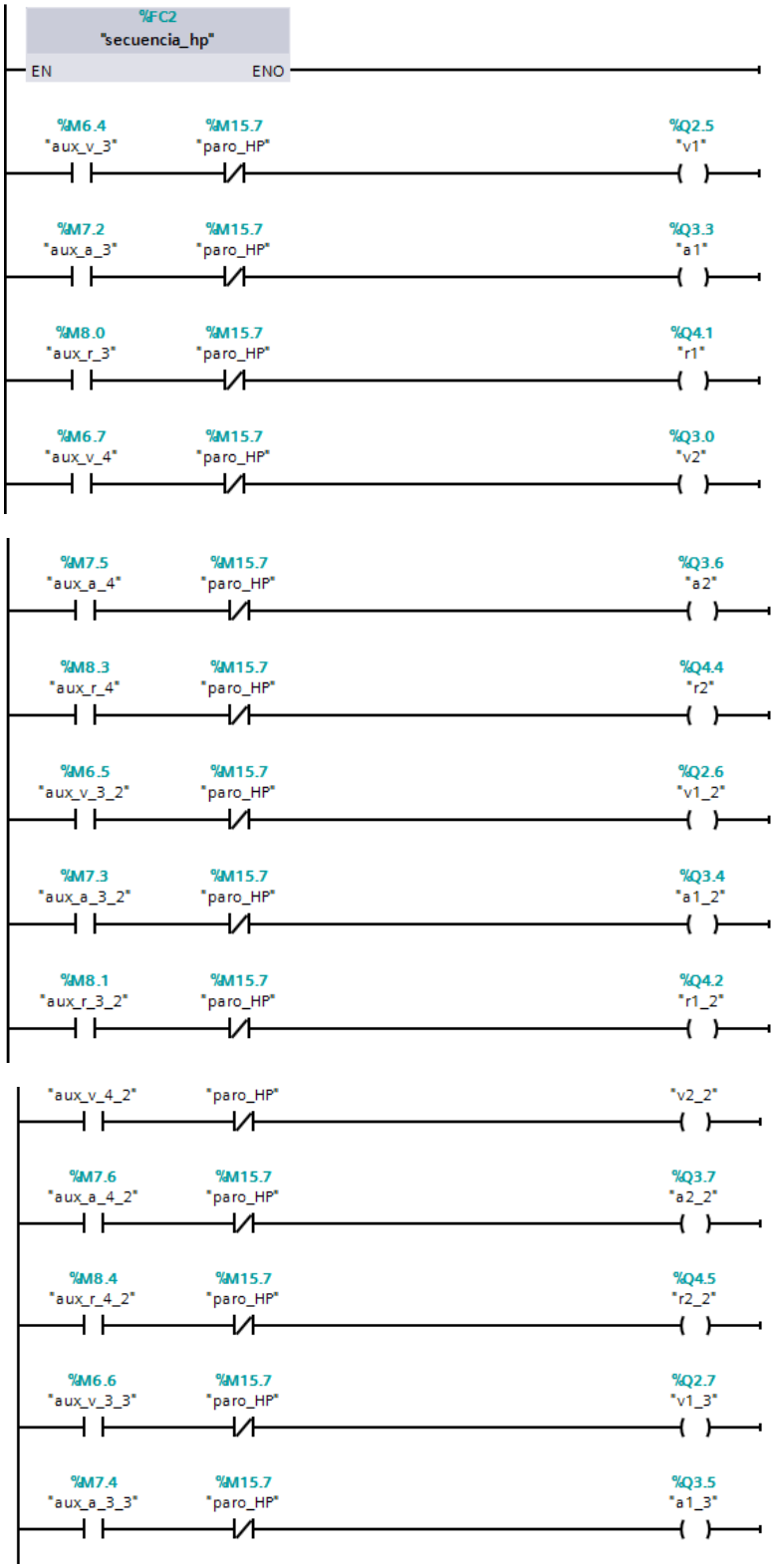


Figura 5. 22: Visualización para la hora pico.

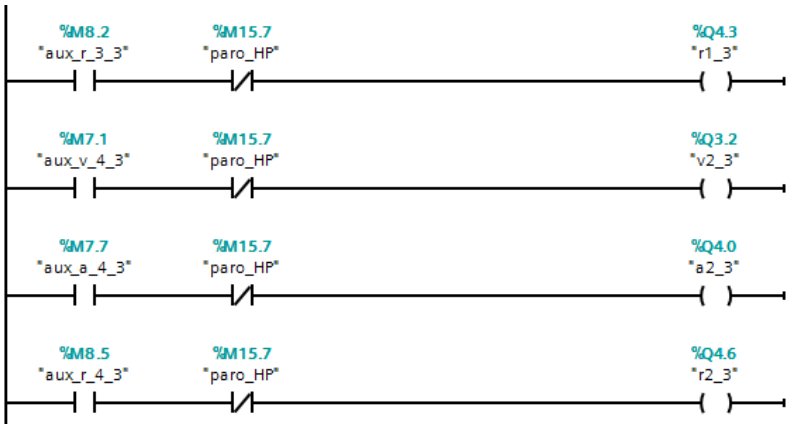


Figura 5. 23: Visualización para la hora pico.

5.3.5. Secuencia de día especial

Los días sábados en la noche y domingos en la mañana cierran la Av. Madero, los sábados para que la gente pueda pasear y admirar el espectáculo del encendido de la catedral y los domingos la cierran para los paseos en bicicletas y carreras deportivas. Los eventos que se realizan en estos días generan aglomeración automotriz en av. Madero como en las calles aledañas por lo cual es difícil transitar por la zona, se decidió que los semáforos estén sincronizados dándole más tiempo a los semáforos secundarios para darle mayor fluidez al tráfico vehicular.

En las Figuras 5.24 a la 5.27 se muestra la implementación.

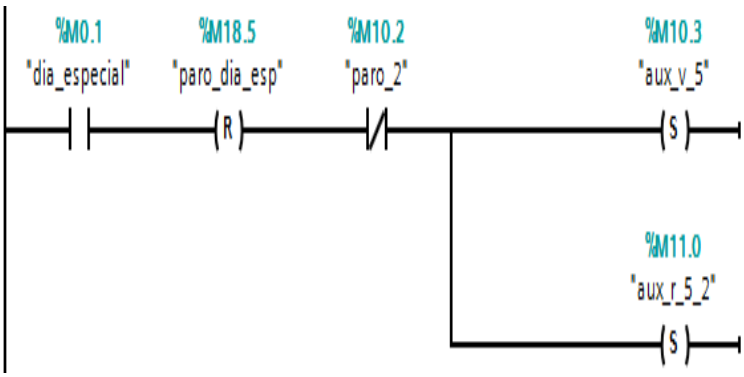


Figura 5. 24: Implementación de activadores de día especial.

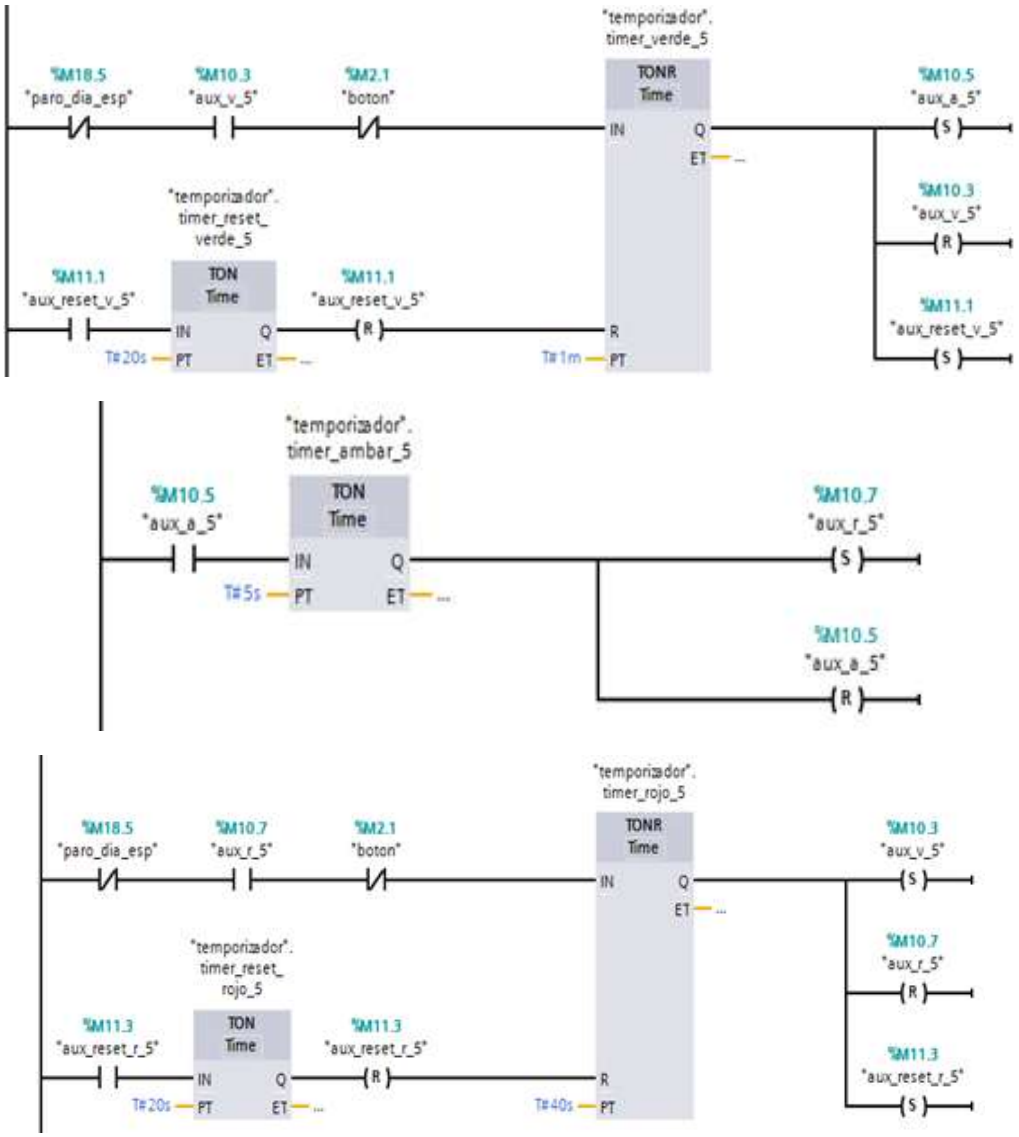


Figura 5. 25: Implementación de secuencia día especial, semáforos secundarios.

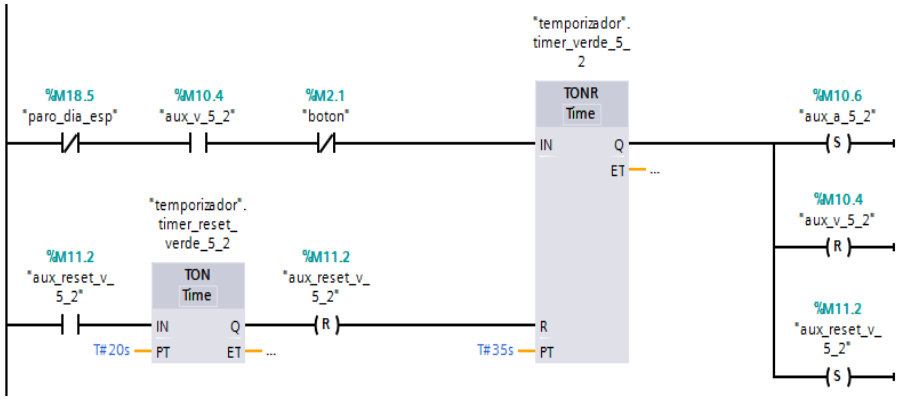


Figura 5. 26: Implementación de secuencia día especial, semáforos principal.

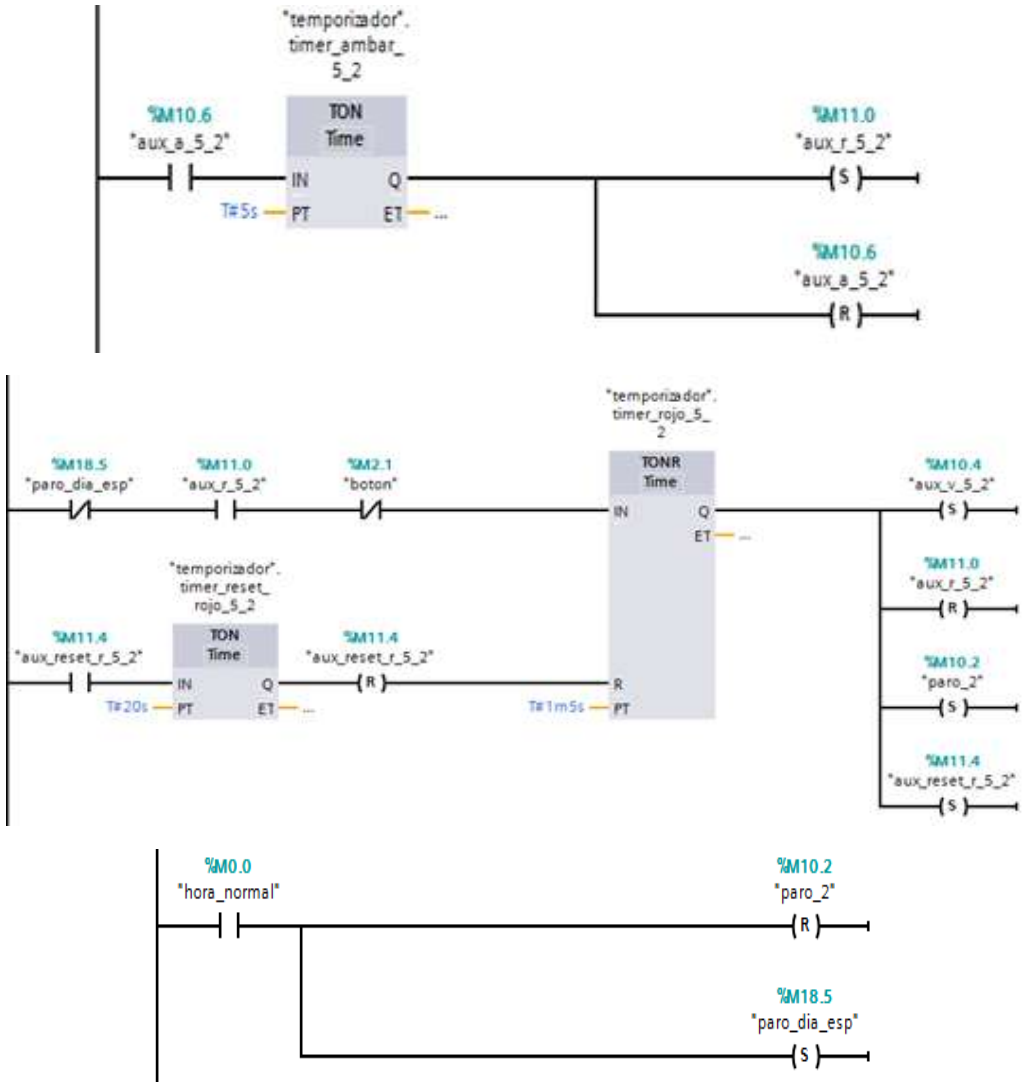


Figura 5. 27: Implementación de secuencia día especial, semáforos principales.

En la Figura 5.28 se muestra la implementación para la visualización.

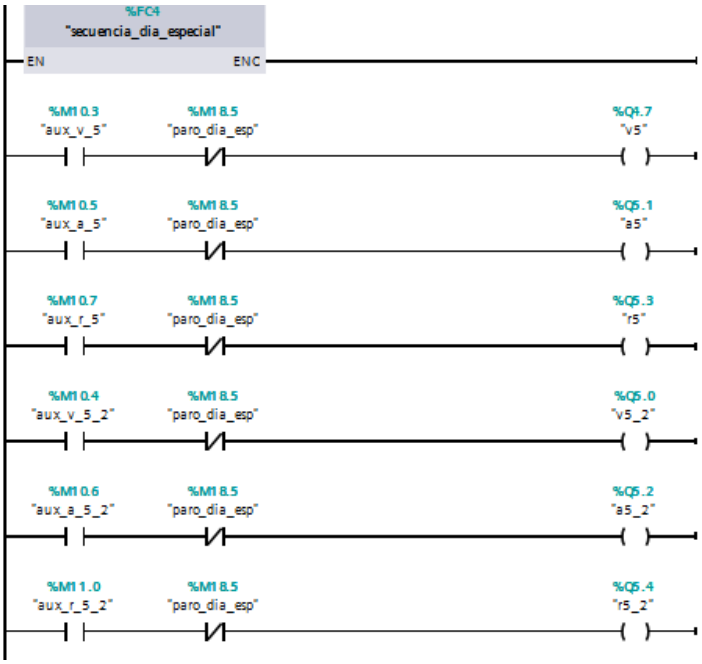


Figura 5. 28: Visualización para días especiales.

5.3.6. Selección modo manual

El objetivó de esta parte del proyecto, es la implementación de una secuencia especial para casos especiales los cuales pueden ser, tomas, marchas, eventos especiales (carreras, conciertos y exposiciones), al seleccionar el modo manual, el modo automático dejará de funcionar, este a su vez permitirá seleccionar el caso especial que se pondrá en marcha. En la Figura 5.29 se muestra la implementación.

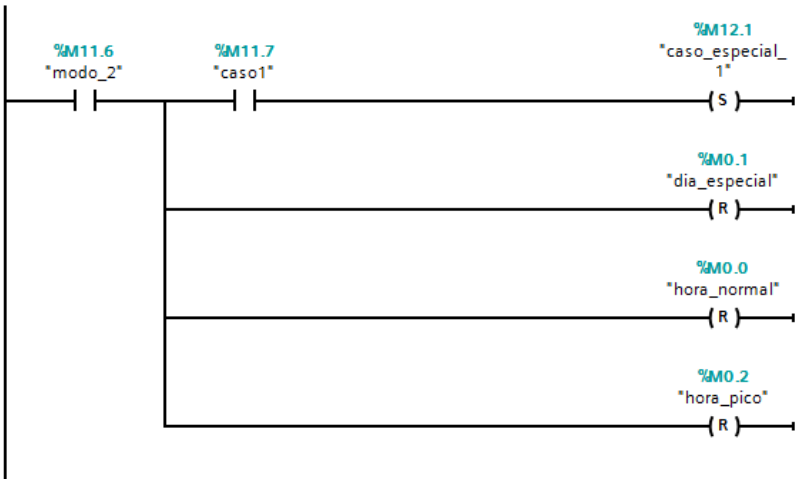


Figura 5. 29: Implementación de modo manual y caso especial.

Antes de poner en marcha el caso especial, el usuario deberá de poner los tiempos de los semáforos, teniendo en cuenta que al poner los tiempos se debe de cumplir que el tiempo en verde y amarillo del semáforo principal debe de ser el tiempo en rojo del secundario y viceversa. El tiempo que se introducirá debe de ser el adecuado para que el tráfico vehicular pueda fluir con más rapidez. Si el tiempo no es el suficiente este se podrá modificar para aumentarlo. La duración de esta secuencia será la duración del caso especial, una vez terminado el caso especial el usuario deberá de regresar al modo automático. En las Figuras 5.30 a la 5.34 se muestra la implementación de la secuencia del caso especial.

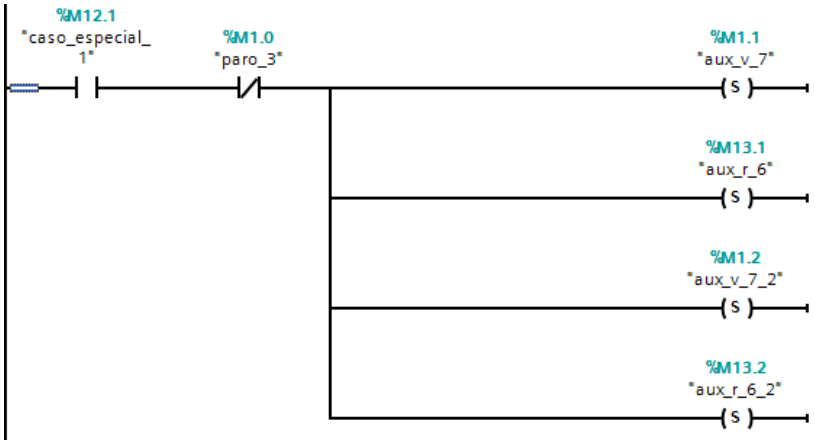


Figura 5. 30: Implementación de activadores de caso especial.

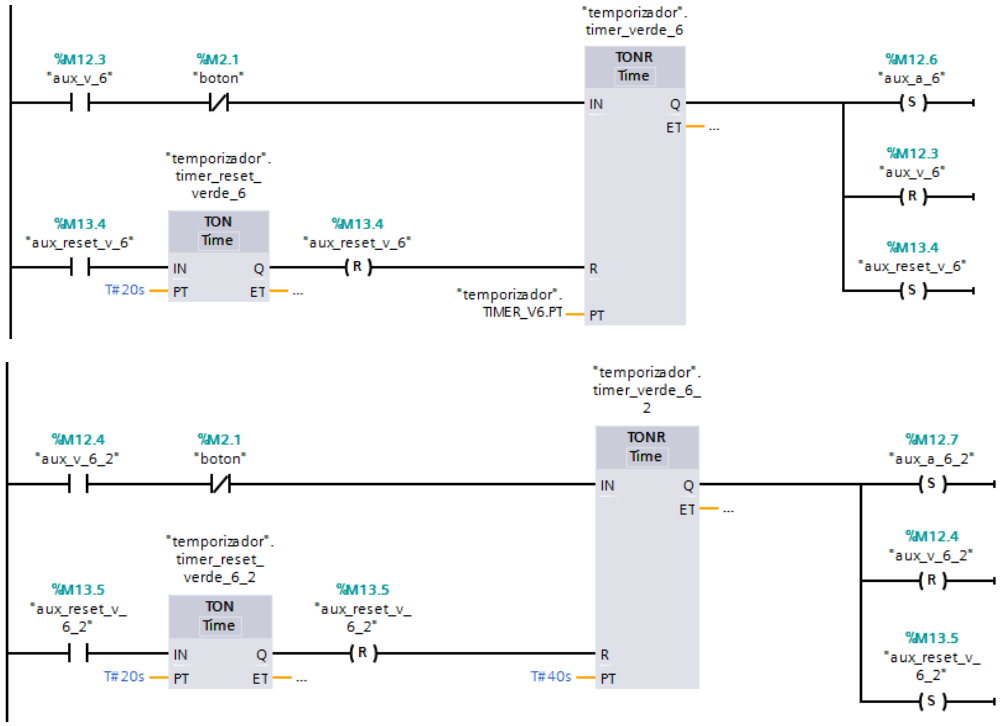


Figura 5. 31: Implementación de secuencia caso especial, semáforos grupo 1.

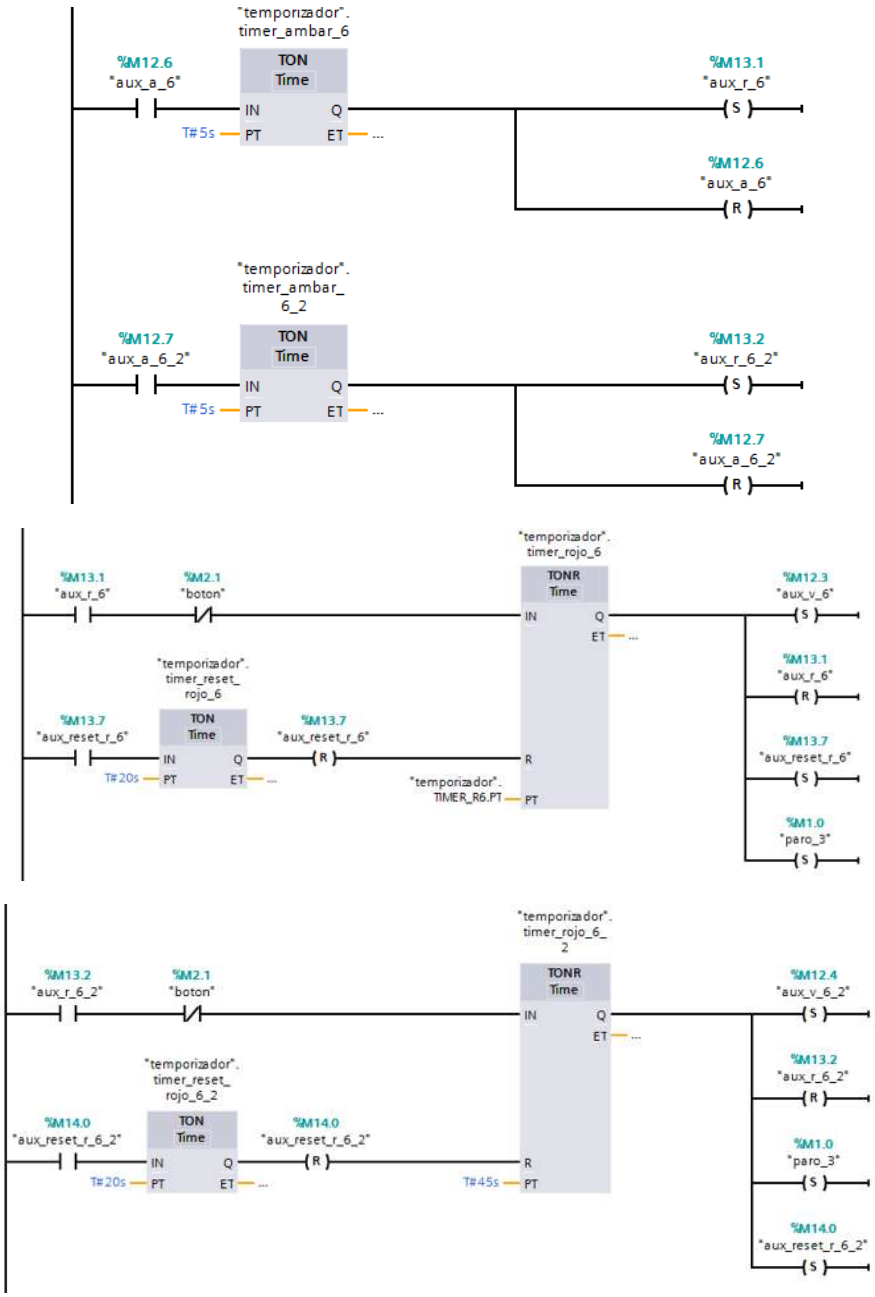


Figura 5. 32: Implementación de secuencia caso especial, semáforos grupo 1.

El grupo 1 de semáforos está conformado por los semáforos de la calle Belisario Domínguez hasta el semáforo de la calle El Nigromante, el grupo 2 está conformado de los semáforos de la av. Acueducto hasta la calle Dr. Miguel Silva.

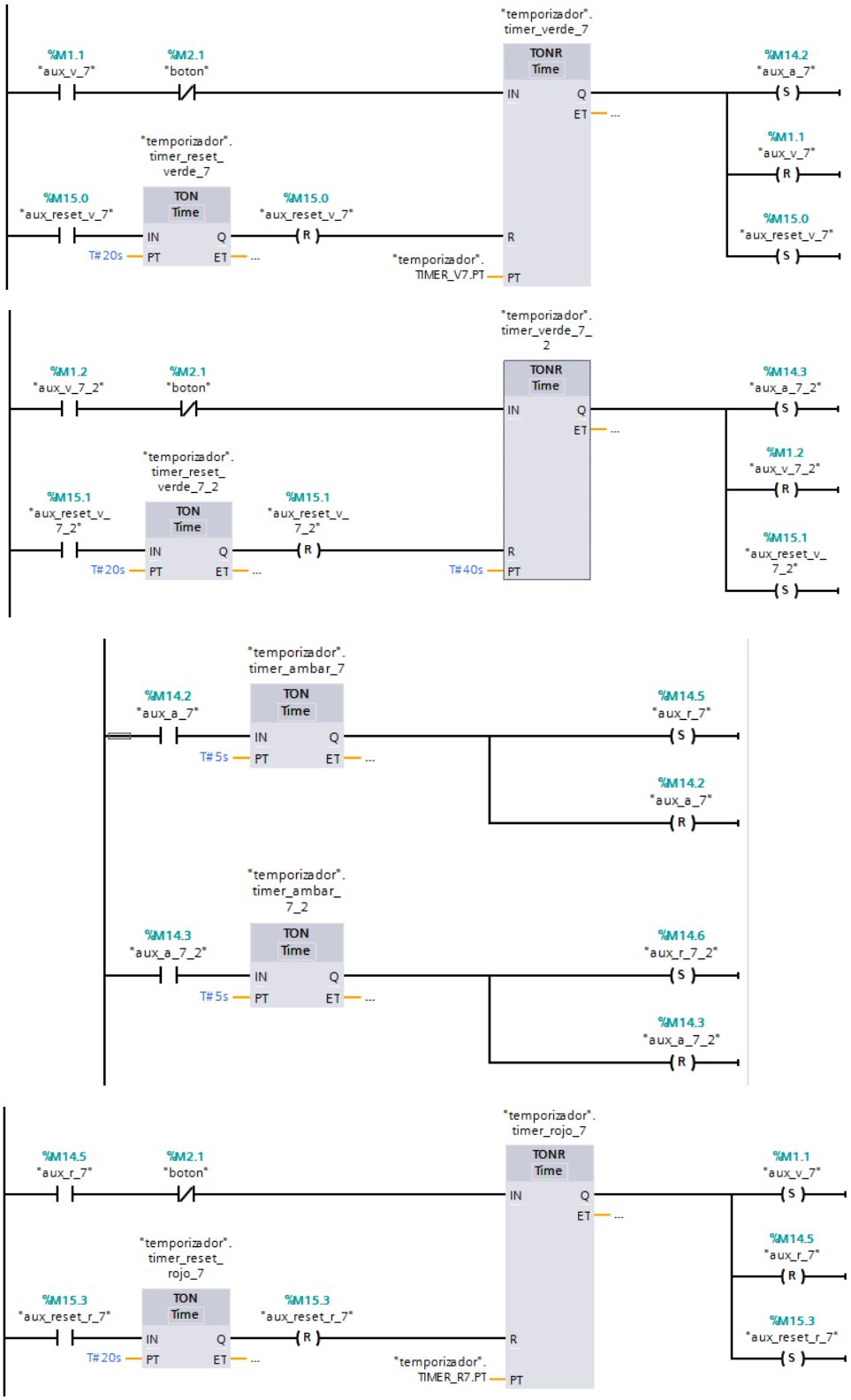


Figura 5. 33: Implementación de secuencia caso especial, semáforos grupo 2.

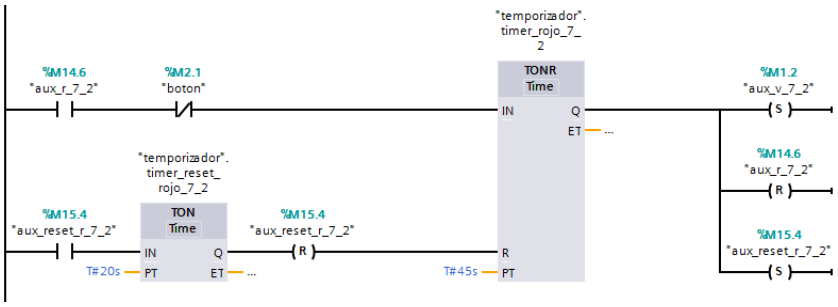


Figura 5. 34: Implementación de secuencia caso especial, semáforos grupo 2.

En la Figura 5.35 se muestra la visualización de la secuencia del caso especial.

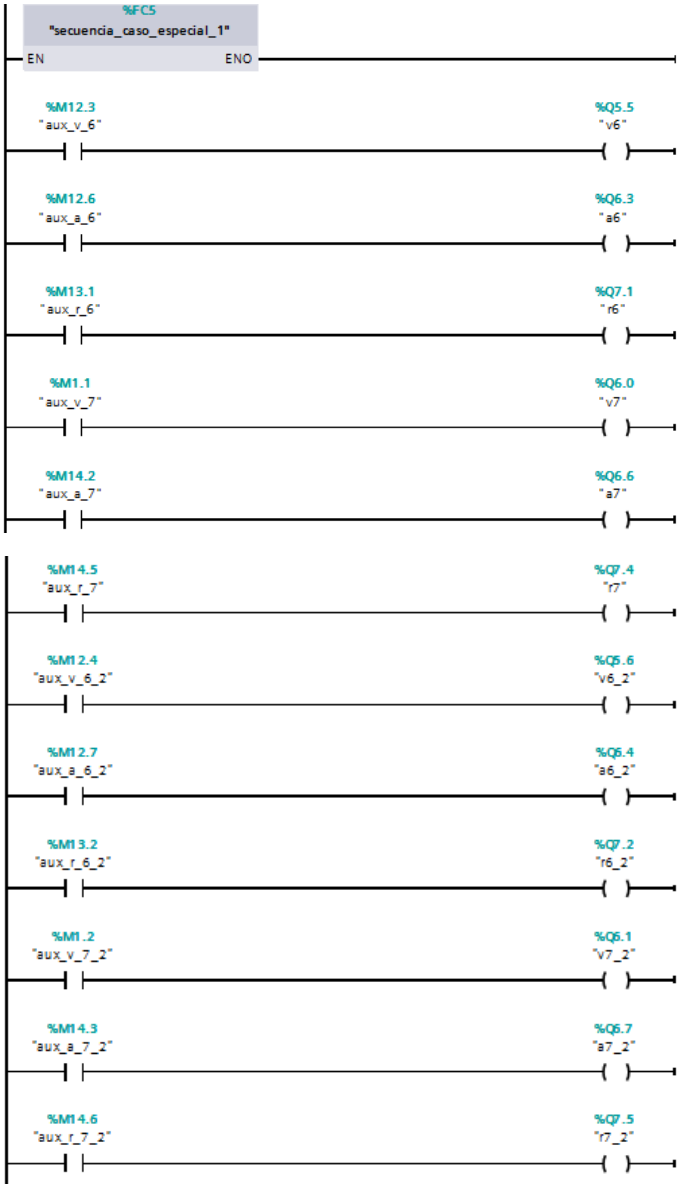


Figura 5. 35: Visualización caso especial.

5.4. Simulación HMI

Debido a que no se cuenta con el hardware para poder realizar la implementación física se llevó a cabo la simulación con la HMI. En este caso se utilizó el PLCSIM V13 para simular las variables del software TIA Portal V13, en la Figura 5.36 se muestra la simulación de las variables.

Nombre	Dirección	Formato de visua..	Valor de observac..	Forzar inmediata...	Bits	Forzar coherente...
"hora_normal"	%M0.0	Bool	FALSE	FALSE		FALSE
"hora_pico"	%M0.2	Bool	FALSE	FALSE		FALSE
"dia_especial"	%M0.1	Bool	FALSE	FALSE		FALSE
"caso_especial_1"	%M12.1	Bool	FALSE	FALSE		FALSE
"modo_1"	%M11.5	Bool	FALSE	FALSE		FALSE
"modo_2"	%M11.6	Bool	FALSE	FALSE		FALSE
"boton"	%M2.1	Bool	FALSE	FALSE		FALSE
"indicador_paso"	%Q0.0	Bool	FALSE	FALSE		FALSE
"paso_peaton"	%Q0.1	Bool	FALSE	FALSE		FALSE
"reactivacion_bo...	%Q0.2	Bool	FALSE	FALSE		FALSE
"verde_1"	%Q0.3	Bool	FALSE	FALSE		FALSE
"verde_1_2"	%Q0.4	Bool	FALSE	FALSE		FALSE
"verde_1_3"	%Q0.5	Bool	FALSE	FALSE		FALSE
"verde_2"	%Q0.6	Bool	FALSE	FALSE		FALSE
"verde_2_2"	%Q0.7	Bool	FALSE	FALSE		FALSE
"verde_2_3"	%Q1.0	Bool	FALSE	FALSE		FALSE
"ambar_1"	%Q1.1	Bool	FALSE	FALSE		FALSE
"ambar_1_2"	%Q1.2	Bool	FALSE	FALSE		FALSE
"ambar_1_3"	%Q1.3	Bool	FALSE	FALSE		FALSE
"ambar_2"	%Q1.4	Bool	FALSE	FALSE		FALSE
"ambar_2_2"	%Q1.5	Bool	FALSE	FALSE		FALSE
"ambar_2_3"	%Q1.6	Bool	FALSE	FALSE		FALSE
"rojo_1"	%Q1.7	Bool	FALSE	FALSE		FALSE
"rojo_1_2"	%Q2.0	Bool	FALSE	FALSE		FALSE
"rojo_1_3"	%Q2.1	Bool	FALSE	FALSE		FALSE
"rojo_2"	%Q2.2	Bool	FALSE	FALSE		FALSE
"rojo_2_2"	%Q2.3	Bool	FALSE	FALSE		FALSE
"rojo_2_3"	%Q2.4	Bool	FALSE	FALSE		FALSE

Figura 5. 36: Tabla de variables en PLCIM.

5.4.1. Pantalla inicio

En la pantalla inicio se tendrá la vista principal que aparecerá al usuario al encender la HMI, en ella se muestra una presentación del proyecto, donde se observa el mapa de implementación con los diferentes modos de operación, como se ve en la Figura 5.37. Cuando esté operando en modo automático se podrá observar el modo de hora que está trabajando como se muestra en las Figuras 5.38 y 5.39.

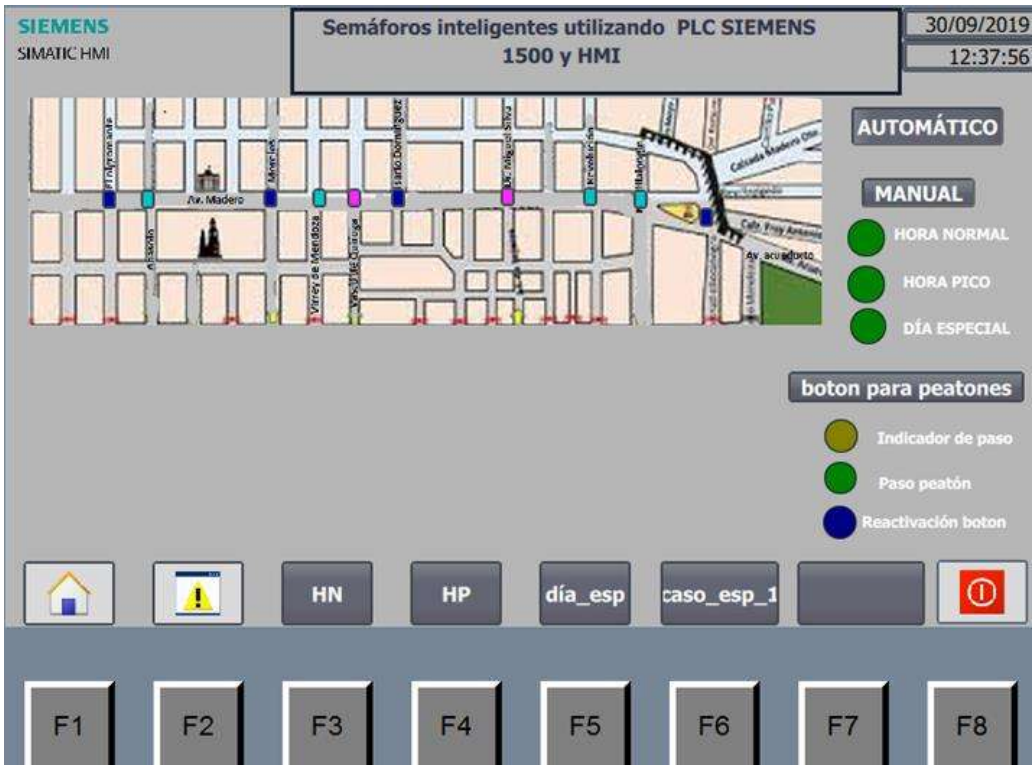


Figura 5. 37: Pantalla inicio.

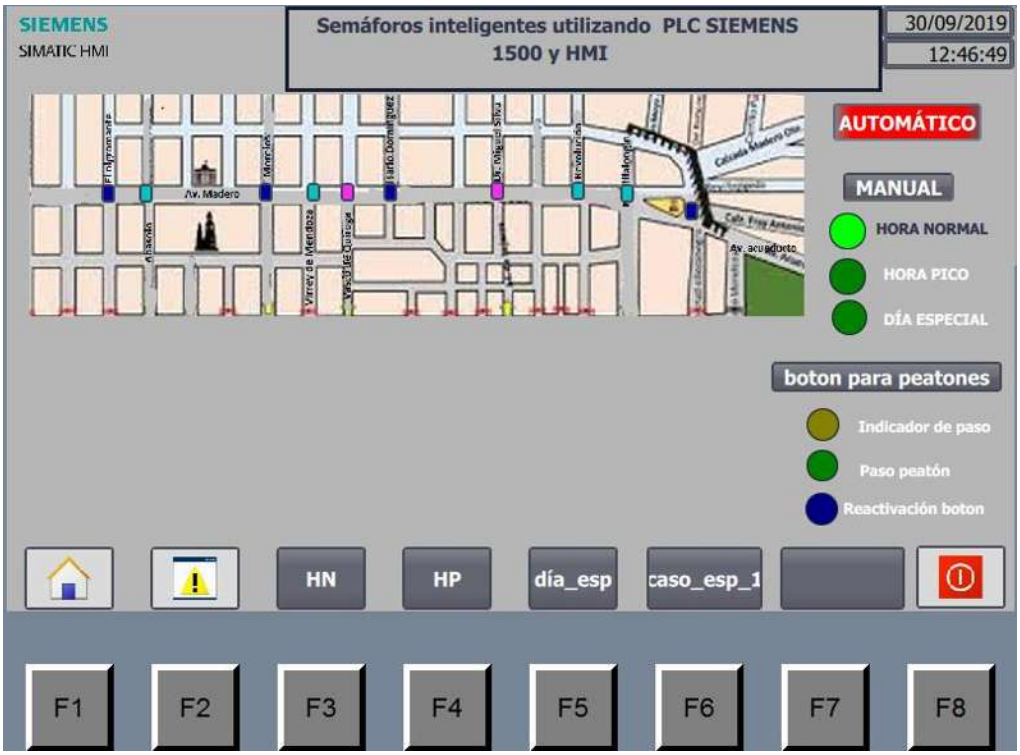


Figura 5. 38: Modo automático funcionado en hora normal.

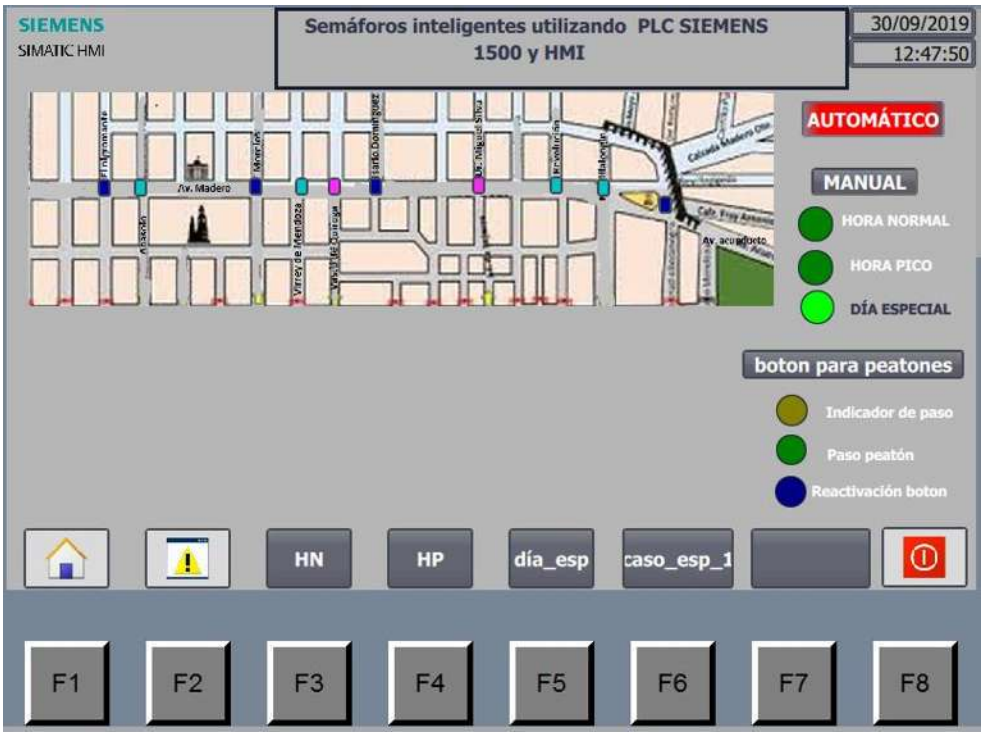


Figura 5. 39: Modo automático funcionando en día especial.

En el mapa de pantalla de inicio se agregaron botones para que el usuario pueda colaborar con el funcionamiento de cierto semáforo en específico, al presionar dicho botón se desplegará en la parte inferior una pequeña muestra de los semáforos, en esta muestra se observan los tres modos de funcionamiento que existen en modo automático, el primero es de la hora normal, el segundo es de la hora pico y el tercero es el de días especiales. Además de que se tiene la visualización del caso de cruce de peatones. En la Figura 5.40 y 5.41 se muestra el funcionamiento de los semáforos, en la Figura 5.42 se ve el funcionamiento del cruce de peatones.

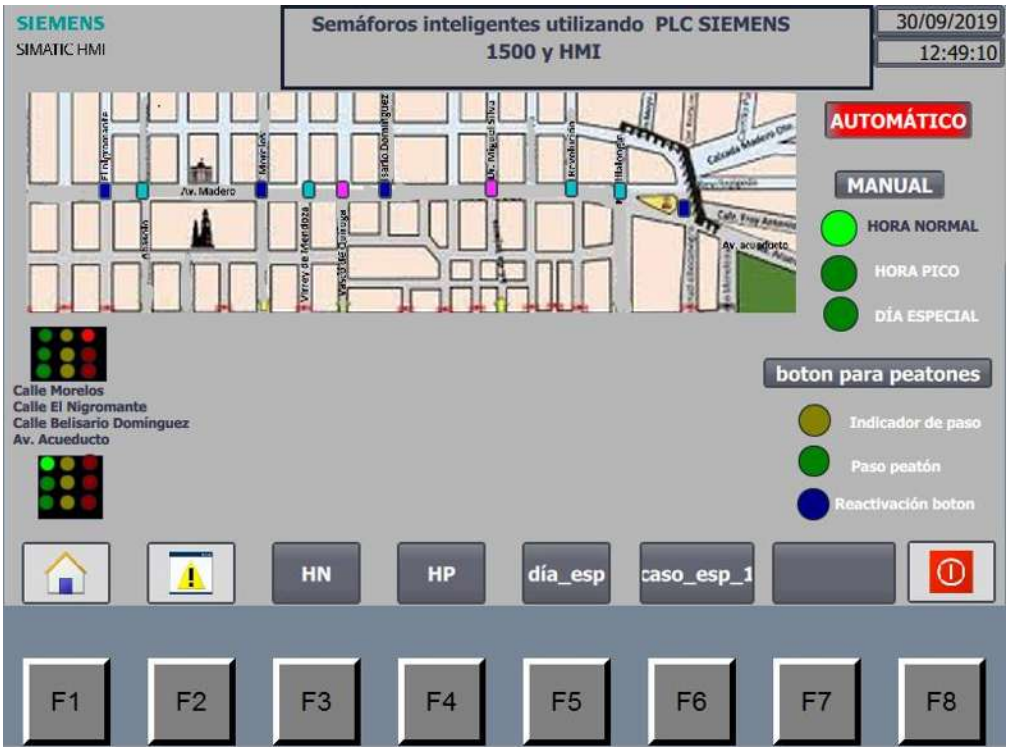


Figura 5. 40: Muestra del funcionamiento de los semáforos del grupo 1.

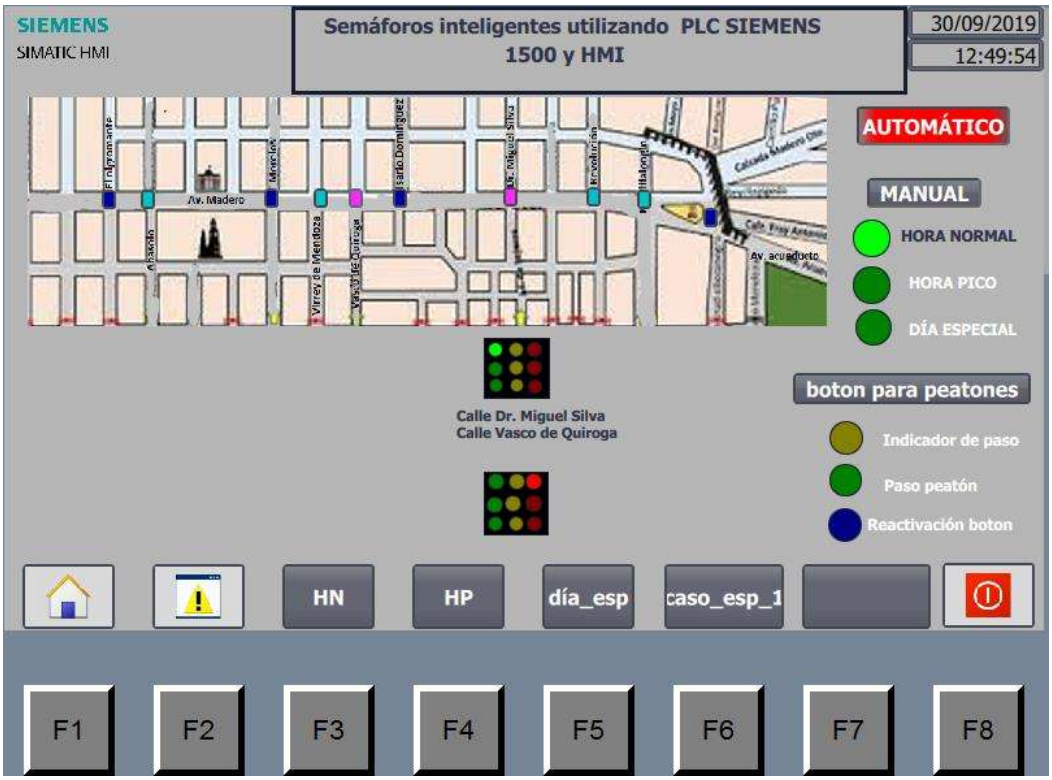


Figura 5. 41: Muestra del funcionamiento de los semáforos del grupo 3.

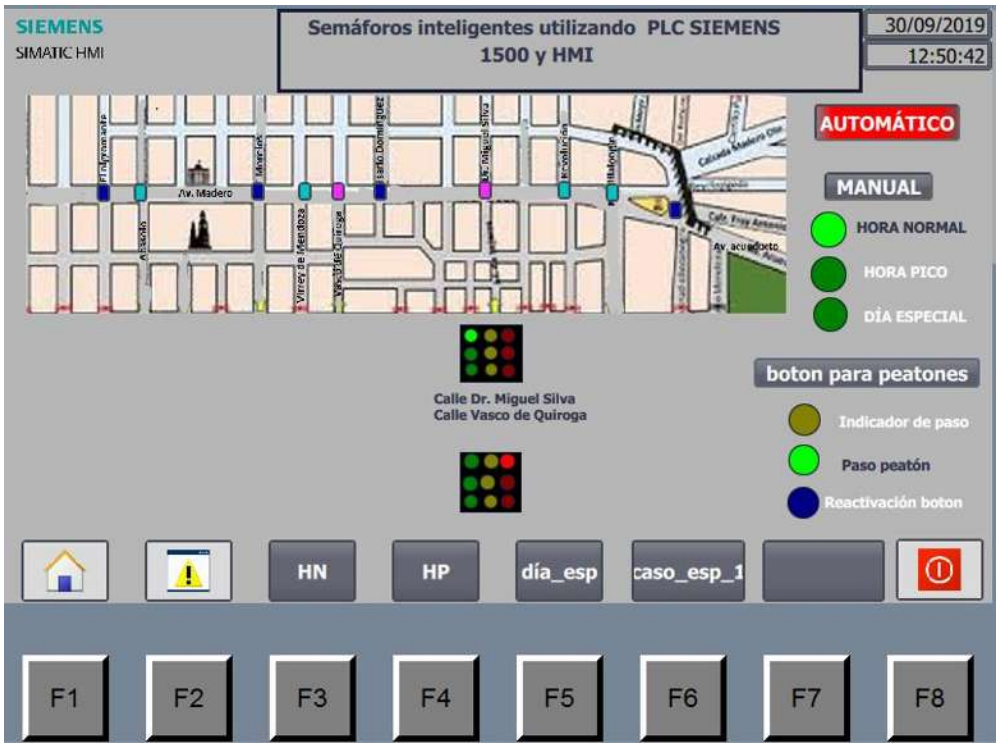


Figura 5. 42. Funcionamiento del cruce de peatones.

5.4.2. Hora normal

Una vez presionado el boton F3 se cambia de pantalla y se muestra la pantalla del funcionamiento Hora Normal, donde se puede observar el funcionamiento de los semáforos en hora normal, en la pantalla se observan los nombres de las calles que controlan los semáforos, del lado izquierdo se tienen los semáforos principales de la Av. Madero y del lado derecho los semáforos secundarios. Estos semáforos fueron agrupados de esta manera para que el automovilista pueda cruzar la avenida en menor tiempo ya que los semaforos estan desincronizados lo cual proporciona un avance más rápido y en un menor tiempo.

En la Figura 5.43 se muestra el funcionamiento de la secuencia de la hora normal con los semáforos secundarios en verde, después de transcurrir el tiempo puesto en el semáforo del grupo 3, en la Figura 5.44 se observa como empieza la desincronizacion de los semáforos del grupo 3, esto se hace para que los automóviles que estan en estos semáforos avancen y para cuando los demás grupos de semáforos cambien del secundario al primario avancen de manera uniforme, en la Figura 5.45 se muestra la desincronizacion de los semáforos del grupo 2, en la Figura 5.46 se ve la desincronizacion del grupo 1, en la Figura 5.47 se observan los semáforos principales en verde.

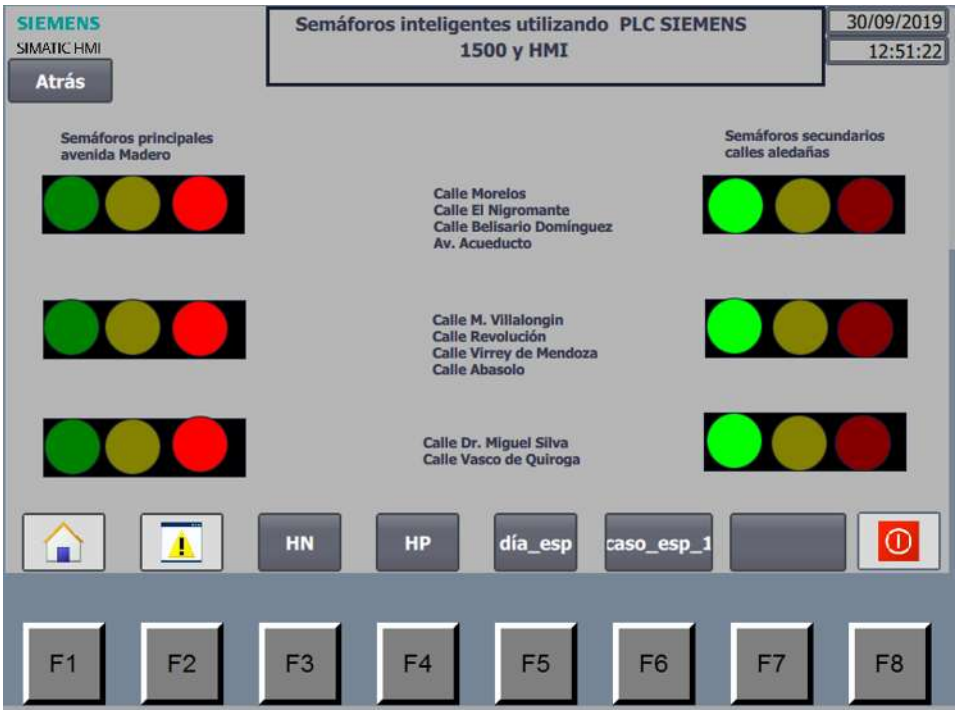


Figura 5. 43: Semáforos secundarios en verde.

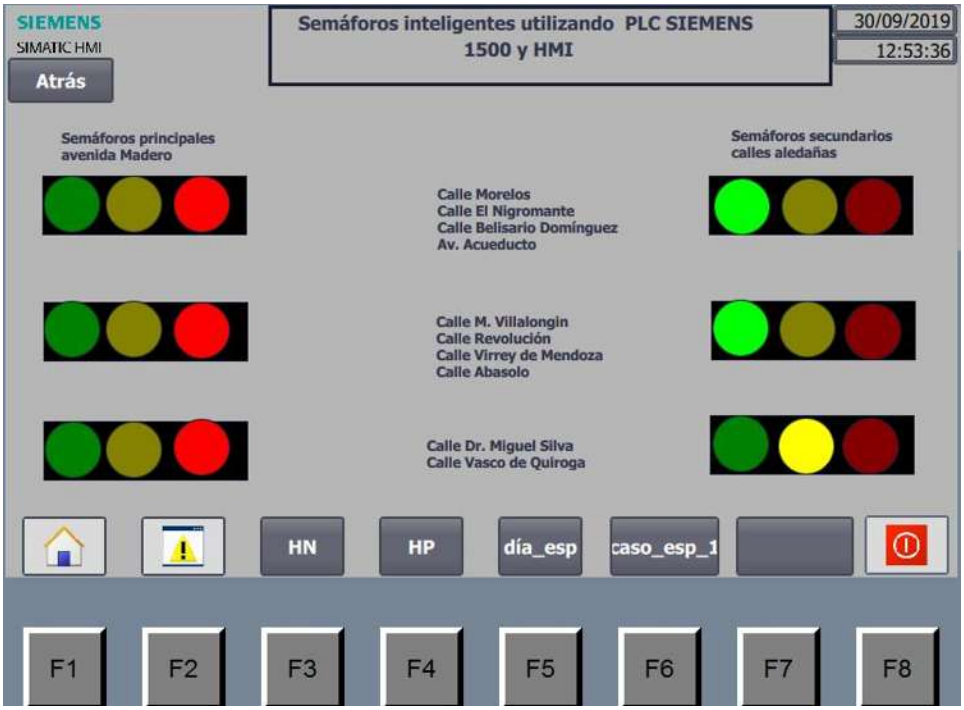


Figura 5. 44: Desincronización de los semáforos del grupo 3.

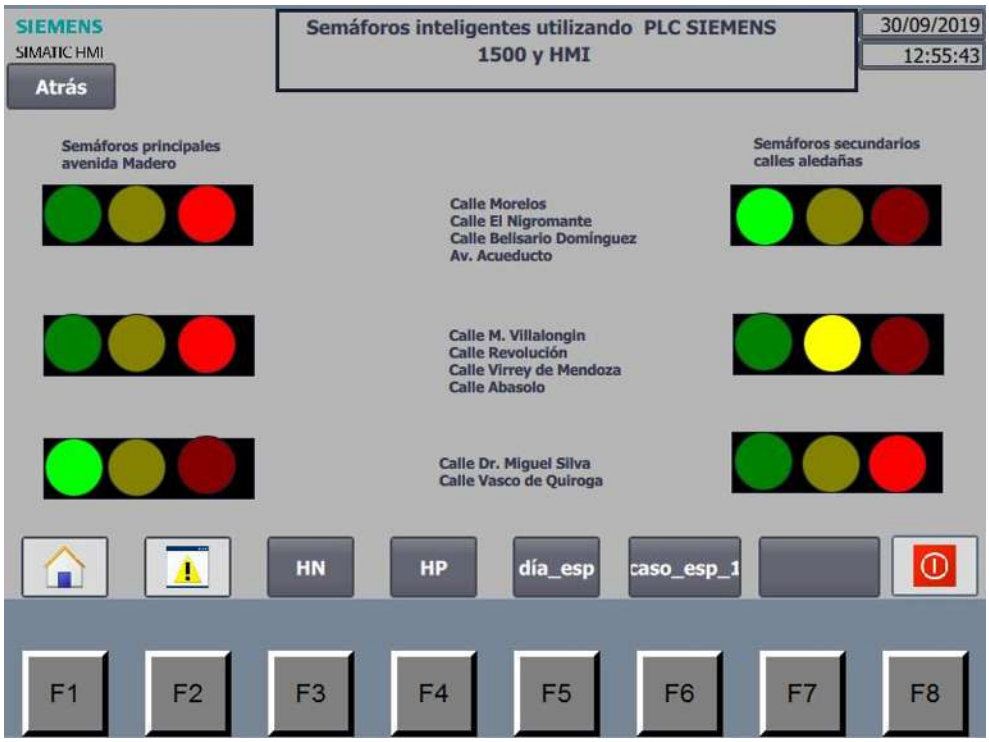


Figura 5. 45: Desincronización de los semáforos del grupo 2.

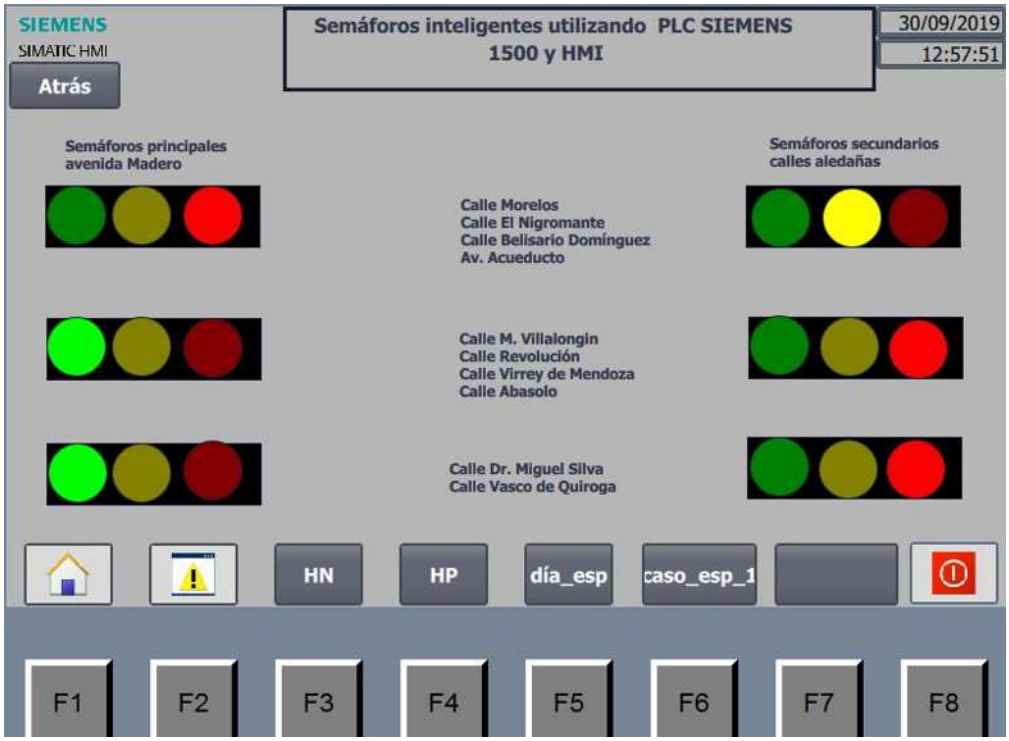


Figura 5. 46: Desincronización de los semáforos del grupo 1.

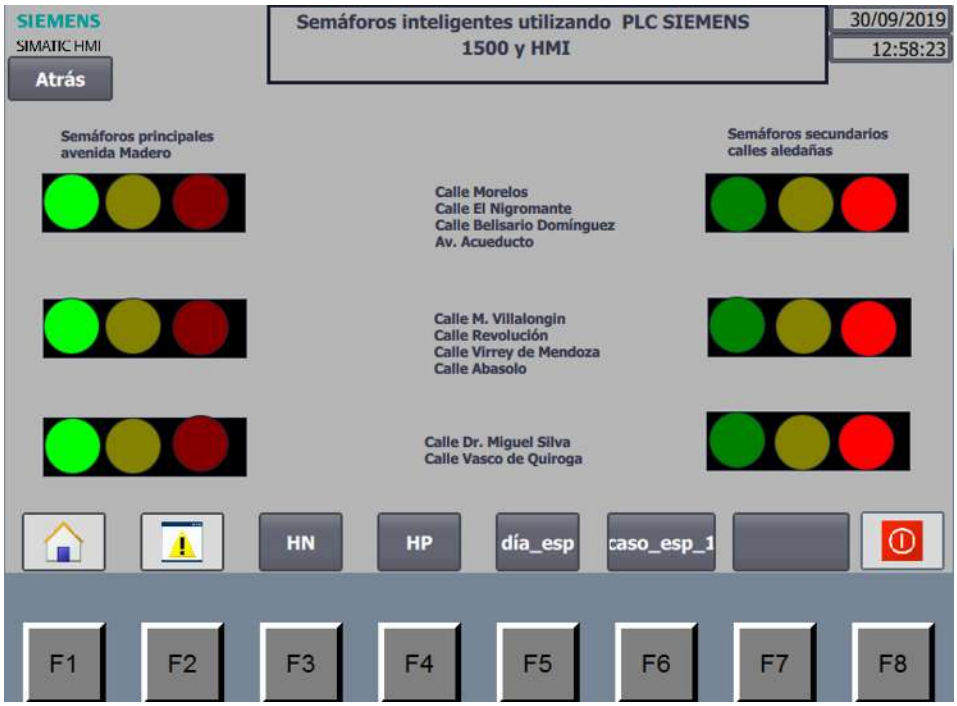


Figura 5. 47: Semáforos principales en verde.

5.4.3. Hora pico

Una vez presionado el botón F4 se cambia de pantalla y se muestra la pantalla del funcionamiento hora pico, donde se puede observar el funcionamiento de los semáforos en esta hora del día, esta secuencia es diferente a la hora normal, cuenta con tiempos más grandes, estos tiempos fueron tomados del estudio del tráfico realizado en la zona de implementación (Sección 3.3.1.), al igual que los semáforos de hora normal estos fueron organizados en grupos para hacer la desincronización y los automovilistas puedan cruzar la avenida en menor tiempo, los tiempos fueron mayores ya que cuando trabaja la hora pico hay mayor cantidad de vehículos por la zona.

En la Figura 5.48 se muestra el funcionamiento de la secuencia de la hora pico con los semáforos secundarios en verde, después de transcurrir el tiempo puesto en el semáforo del grupo 3 ,en la Figura 5.49 se observa como empieza la desincronizacion de los semáforos del grupo 3, esto se hace para que los automóviles que estan en estos semáforos avancen y para cuando los demás grupos de semáforos cambien del secundario al primario avancen de manera uniforme , en la Figura 5.50 se muestra la desincronizacion del grupo 2, en la Figura 5.51 se ve la desincronizacion del grupo 1 y en la Figura 5.52 se observan los semáforos principales en verde.

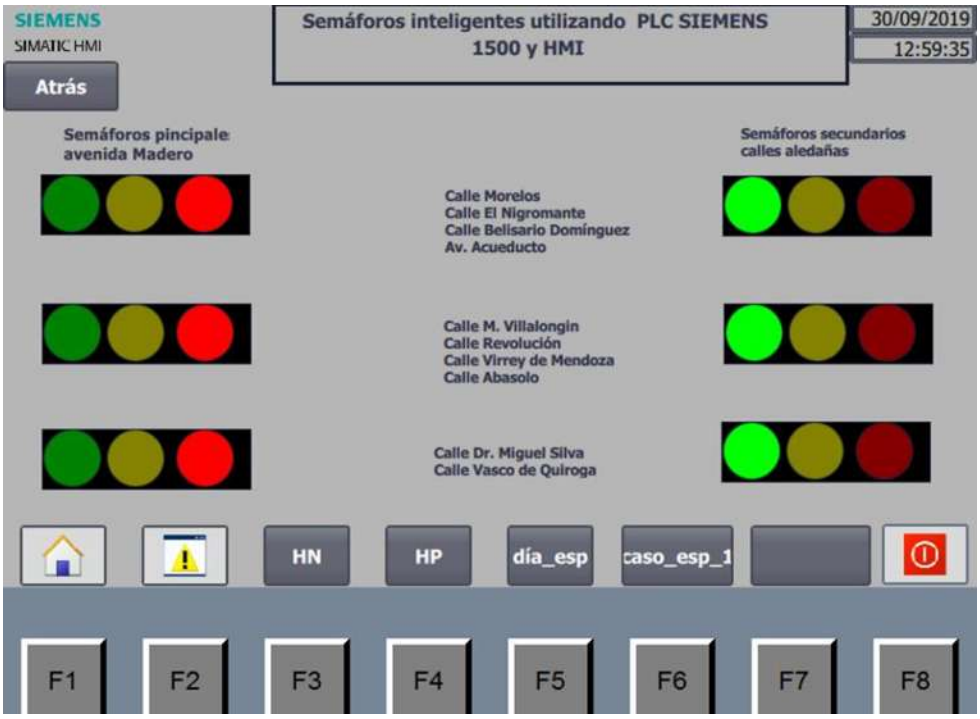


Figura 5. 48: Semáforos secundarios en verde.

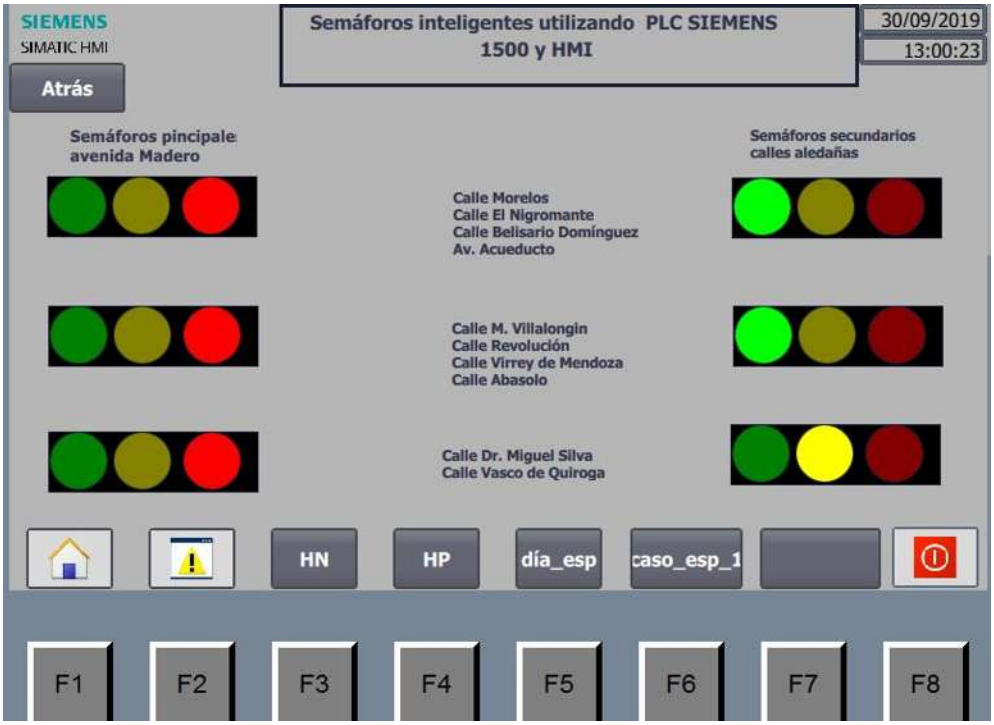


Figura 5. 49: Desincronización de semáforos del grupo 3.

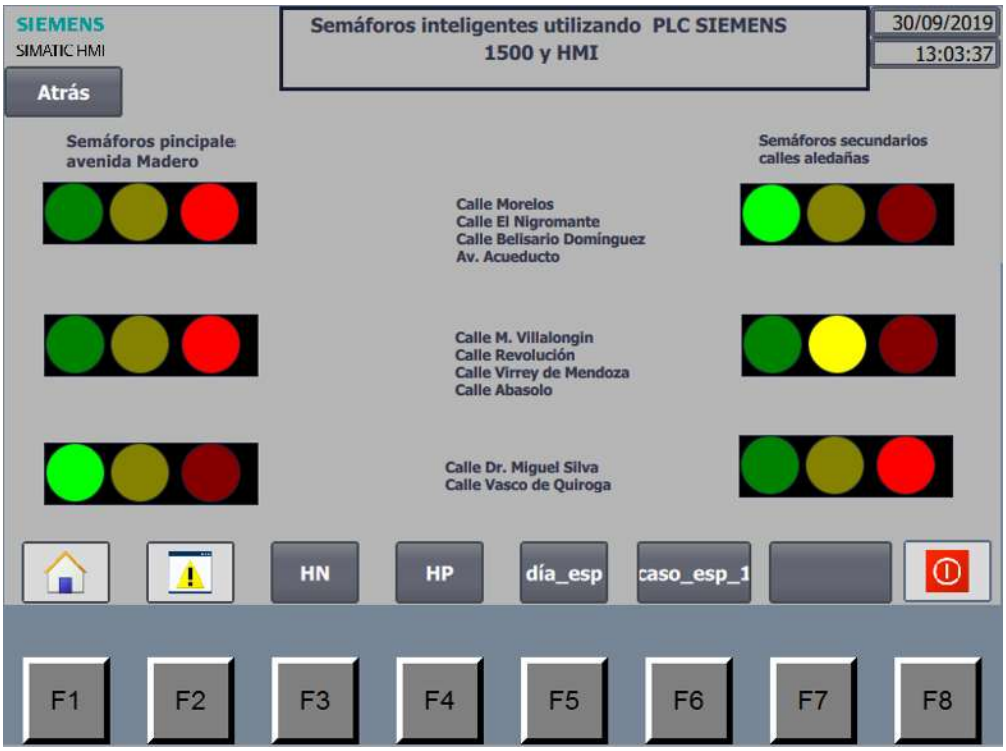


Figura 5. 50: Desincronización de semáforos del grupo 2.

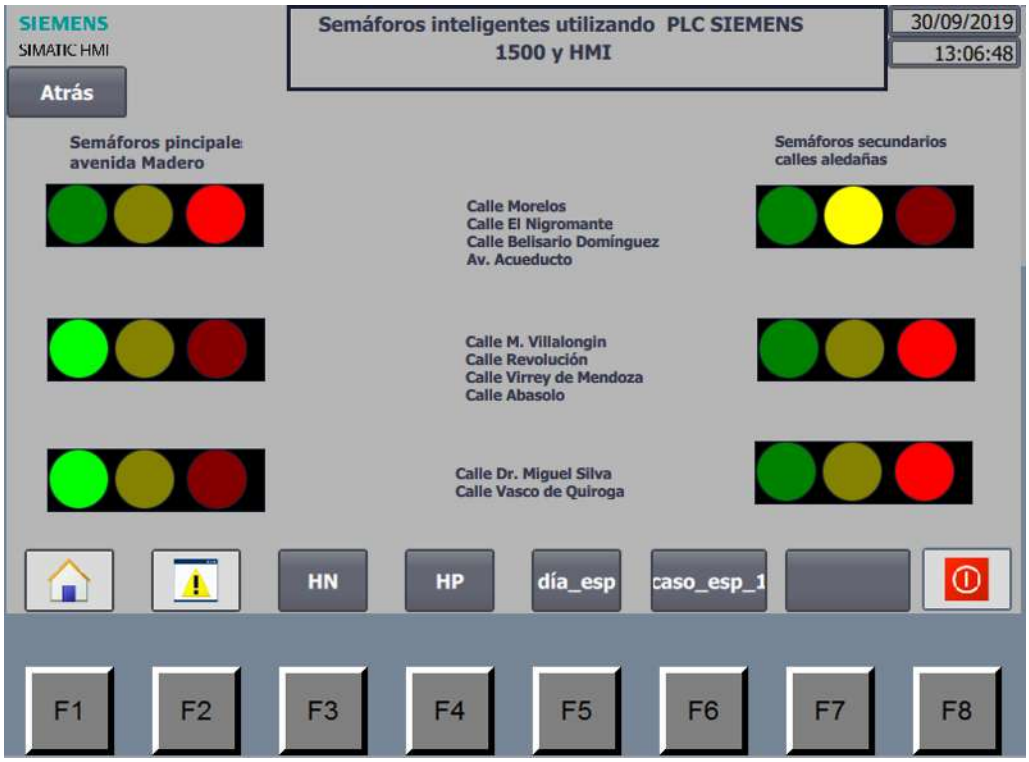


Figura 5. 51: Desincronización de los semáforos del grupo 1.

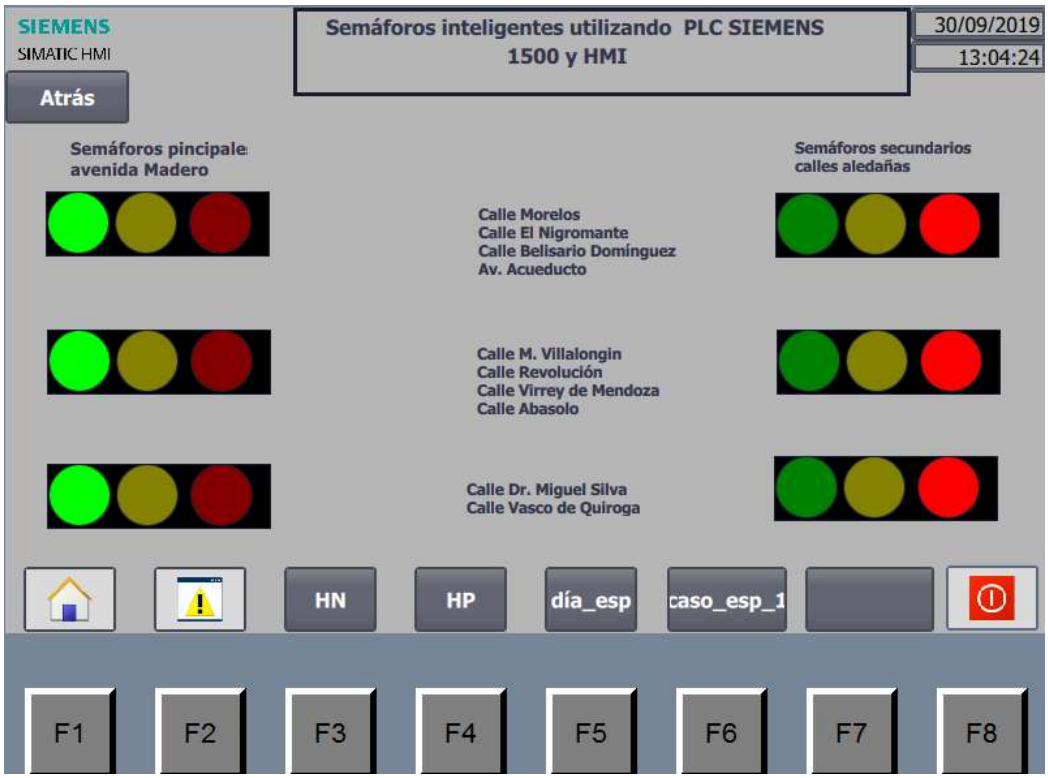


Figura 5. 52: Semáforos principales en verde.

5.4.4. Día especiales

Cuando entra en funcionamiento la secuencia de día especial (sábados y domingos), presionando el botón F5 se cambia la pantalla y nos muestra el funcionamiento de días especiales, donde se puede observar la secuencia de día especial, donde la prioridad la tienen los semáforos secundarios, ya que estos días cierran la avenida Madero, en las calles de los semáforos secundarios se hace mayor el tráfico, así que estos semáforos están sincronizados y tienen mayor tiempo que los semáforos principales de la avenida principal.

En la Figura 5.53 se muestra los semáforos de la avenida Madero en verde con un tiempo menor, en la Figura 5.54 se ven los semáforos secundarios en verde con mayor tiempo que los de la avenida Madero.

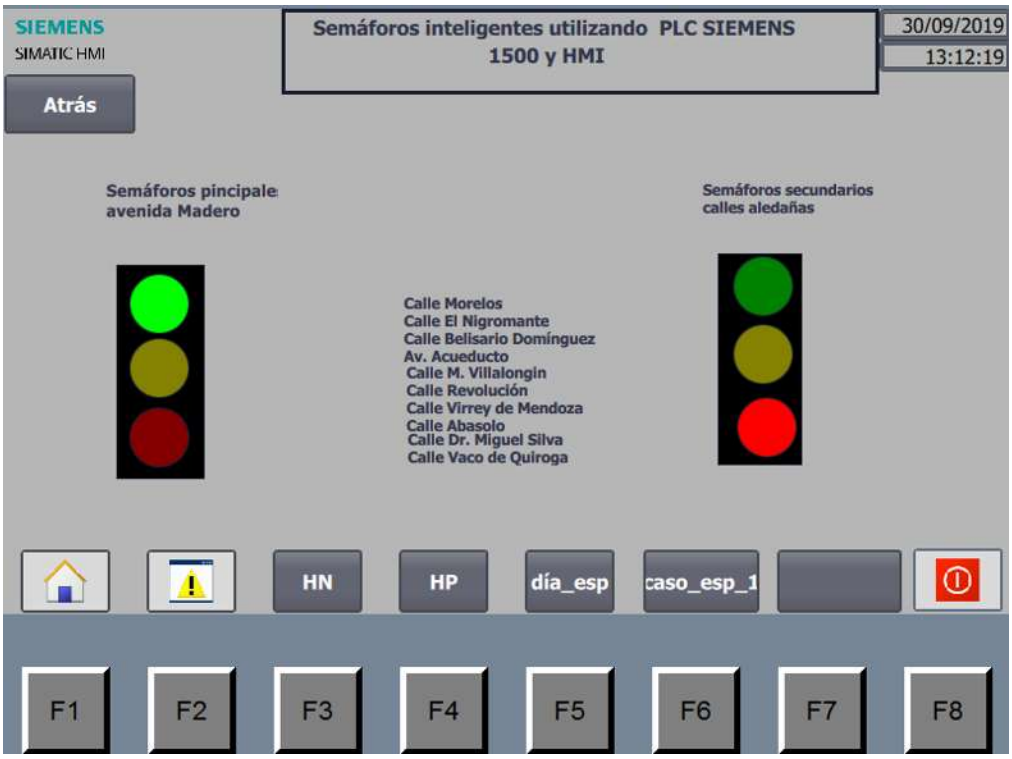


Figura 5. 53: Semáforos principales sincronizados en verde.

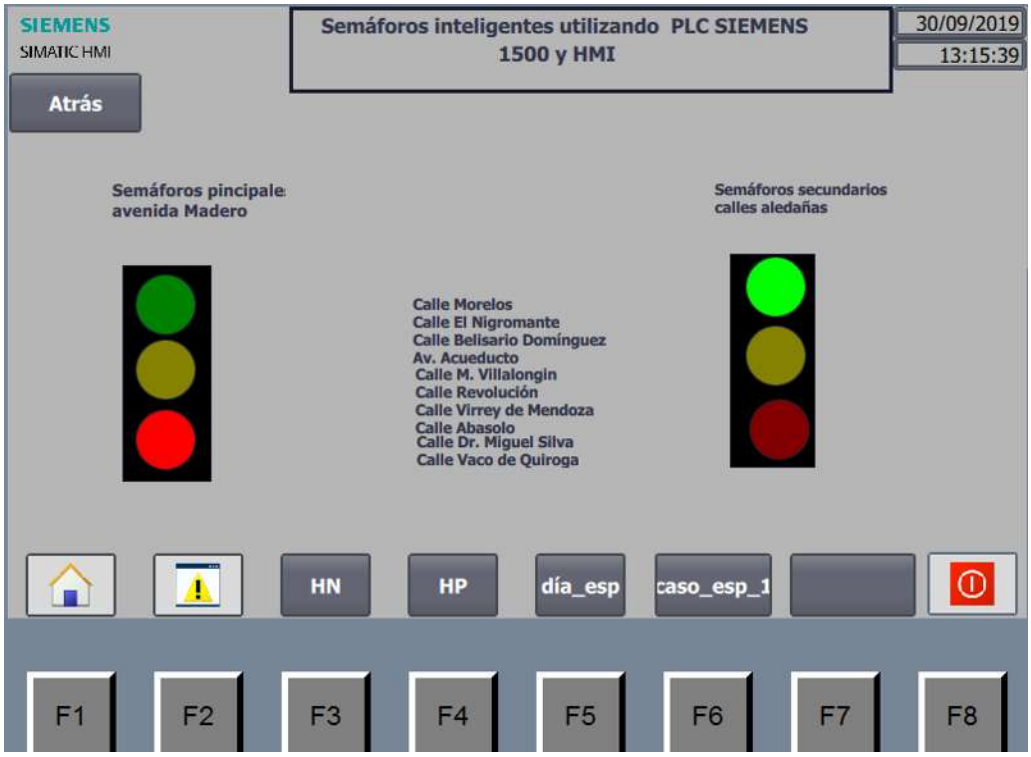


Figura 5. 54: Semáforos secundarios sincronizados en verde.

5.4.5. Caso especial

Para la selección de caso especial se puede observar en la pantalla inicio, se tienen las dos opciones, el modo automático, el cual el usuario la seleccionará, esta se encarga de leer la hora, fecha y trabajo bajo las horas de las secuencias y el modo manual, el cual el usuario podrá elegir presionando el botón F6 para cambiar la pantalla al caso especial, en esta pantalla el usuario tendrá que poner los tiempo de modo manual, los tiempos introducidos tienen que ser suficientes para que el tráfico vehicular pueda disminuir, para los tiempos se tienen que meter de las siguiente manera, para poner 50 segundos, el usuario tendrá que multiplicar el tiempo deseado por 100, para su funcionamiento óptimo el tiempo puesto en el tiempo verde del principal se le debe sumar el tiempo en ámbar (el tiempo del ámbar son 5 segundos) y el resultado se debe de poner en el tiempo en rojo del secundario y viceversa.

En la Figura 5.55 se muestra el modo manual trabajando, en la Figura 5.56 y 5.57 se observa cómo se insertan los valores de los semáforos y en las figuras 5.58 y 5.59 se ve funcionando el caso especial.

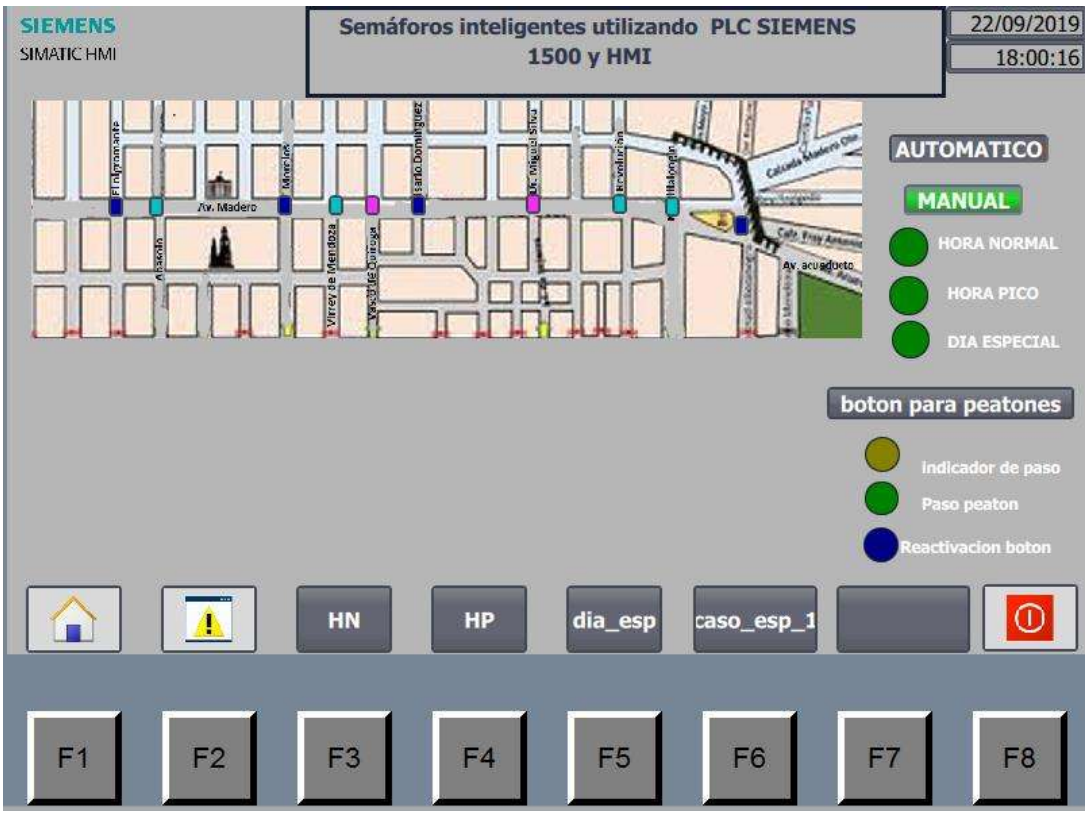


Figura 5. 55: Modo manual funcionando.

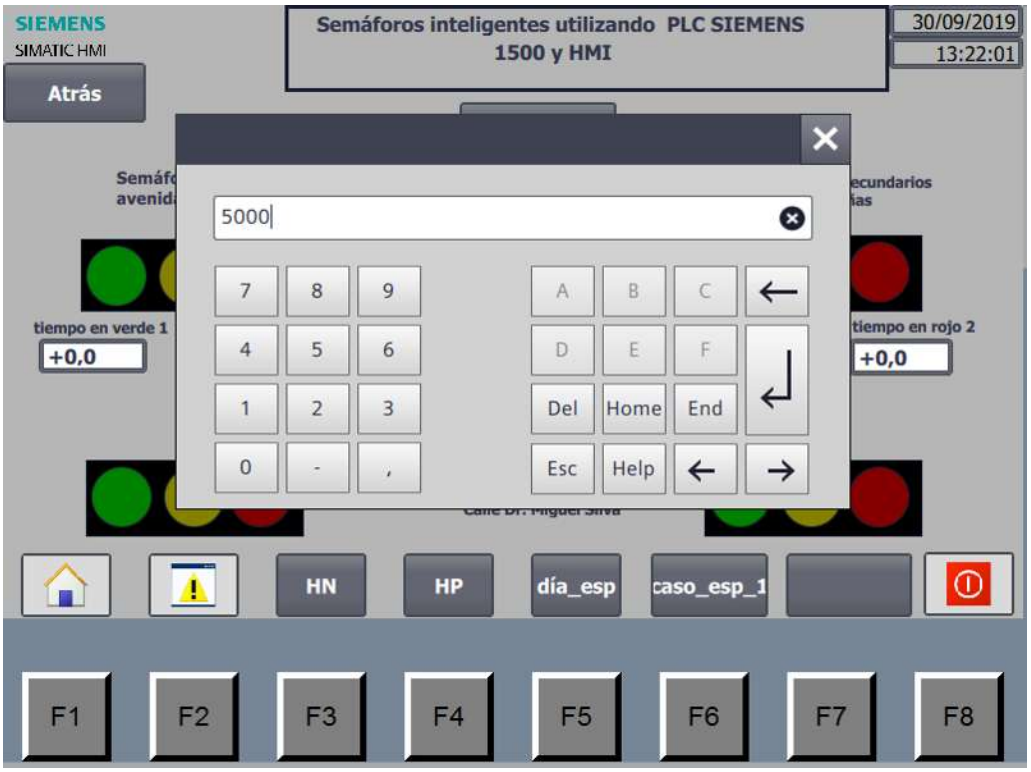


Figura 5. 56: Muestra de cómo se agrega el tiempo en caso especial.

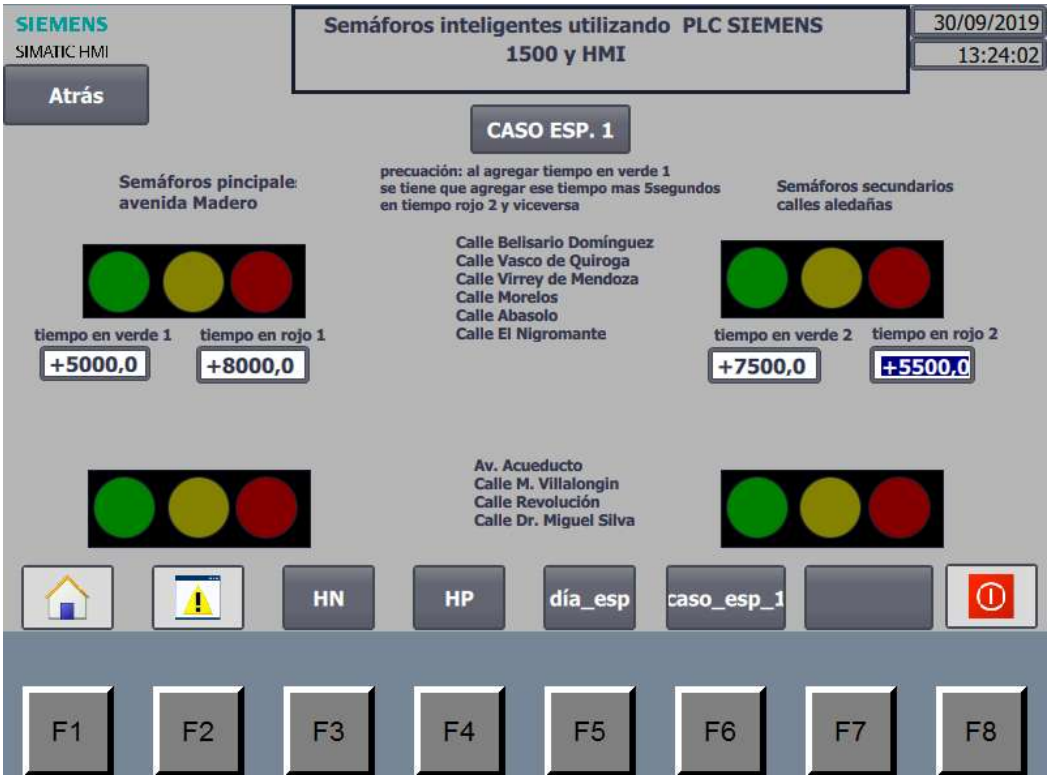


Figura 5. 57: Tiempos puestos en caso especial.

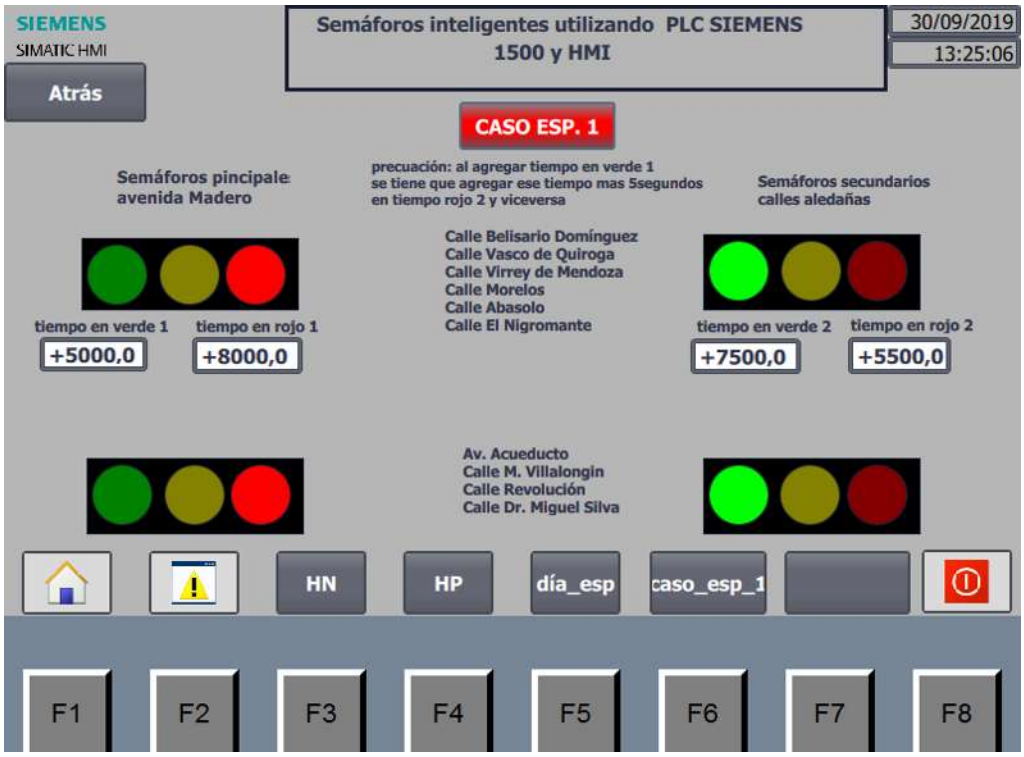


Figura 5. 58: Funcionamiento de caso especial.

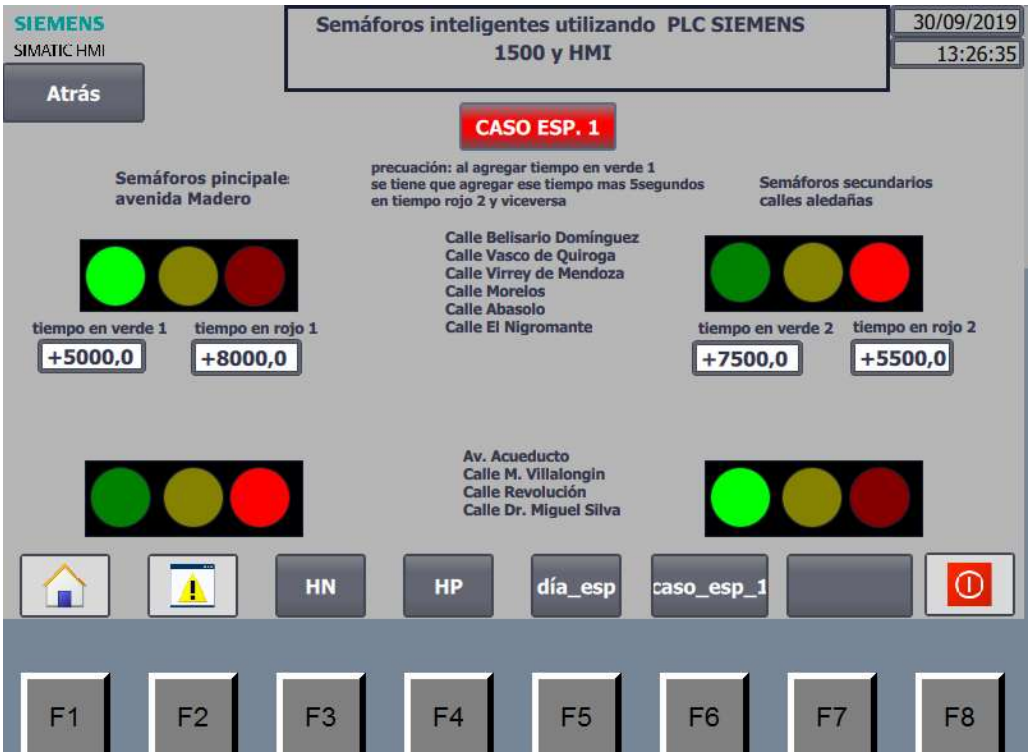


Figura 5. 59: Funcionamiento de caso especial.

Capítulo 6 Conclusiones y trabajos futuros

6.1. Conclusiones

Una ventaja que tiene el PLC sobre los microcontroladores es la gran potencia que tiene, el PLC tiene mayor capacidad de entradas y salidas, los PLC son utilizados para proyectos grandes ya que es relativamente fácil de utilizar, el PLC se puede adaptar a las necesidades del usuario.

Para hacer la automatización de los semáforos con el PLC fue necesario la utilización de diferentes secuencias para el funcionamiento óptimo, así como para hacer la comparación de los días de trabajo, como trabajar con un conjunto de semáforos en los cuales funcionan en dos modos diferentes, lo cual facilita el manejo de los mismos, para la fluidez de la aglomeración automotriz, esto resulta eficiente a la hora de mejorar la flujo de los automóviles llegando a reducir en un 25 % el tráfico automotriz.

6.2. Trabajos futuros

En trabajos futuros se podría hacer la unión de algunos de los distintos semáforos inteligentes con el proyecto realizado.

La ventaja de este proyecto que maneja varios semáforos en conjunto a comparación de otros semáforos inteligentes los cuales solo están diseñados para funcionar con un solo semáforo. Teniendo los datos de los automóviles que circulan por avenidas y calles se podría complementar con sensores para detectar cuantos automóviles pasan y así determinar un algoritmo para las diferentes secuencias, así como también se podrían utilizar los semáforos inteligentes con procesamiento de imágenes, el reto viene a ser el juntar dos diferentes softwares ya que los otros semáforos inteligentes utilizan el microcontrolador como herramienta principal y software como Matlab o labview.

Apéndice

A.1. Bloque de datos (DB)

Los bloques de datos (DB) se crean en el programa de usuario para almacenar los datos de los bloques lógicos. Todos los bloques del programa de usuario pueden acceder a los datos en un DB global. En cambio, un DB instancia almacena los datos de un bloque de función (FB) especifico [9].

Los datos almacenados en un DB no se borran cuando finaliza la ejecución del bloque lógico asociado, hay dos tipos de DBs, a saber [9]:

- Un DB global almacena los datos de los bloques lógicos en el programa. Cualquier OB, FB o FC puede acceder a los datos en un DB global.
- Un DB instancia almacena los datos de un FB especifico. La estructura de los datos en un DB instancia refleja los parámetros (Input, Output e Inout) y los datos estáticos del FB (la memoria temporal del FB no se almacena en el DB instancia).

Nota: aunque el DB instancia refleja los datos de un FB específico, cualquier bloque lógico puede acceder a los datos de un DB instancia.

A.2. Tipos de datos fecha y hora

Tabla A. 1: Tipos de datos fecha y hora.

Tipo de datos	Tamaño	Rango	Ejemplos de entrada de constantes
Hora	32 bits	T#-24d_20h_31m_23s_648ms a T#24d_20h_31m_23s_647ms Almacenado como: -2.147.483.648 ms hasta +2.147.483.647 ms	T#5m_30s T#1d_2h_15m_30s_45ms TIME#10d20h30m20s630ms 500h10000ms 10d20h30m20s630ms
Fecha	16 bits	D#1990-1-1 a D#2168-12-31	D#2009-12-31 DATE#2009-12-31 2009-12-31
Hora	32 bits	TOD#0:0:0.0 a TOD#23:59:59.999	TOD#10:20:30.400 TIME_OF_DAY#10:20:30.400 23:10:1
DTL (Fecha y hora largo)	12 bytes	Mín.: DTL#1970-01-01-00:00:00.0 Máx.: DTL#2554-12-31-23:59:59.999 999 999	DTL#2008-12-16-20:30:20.250

A.2.1. hora

El dato TIME se guarda como entero doble con signo y se interpreta como milisegundos. El formato del editor puede utilizar información para día (d), horas (h), minutos (m), segundos (s) y milisegundos (ms).

No es necesario especificar todas las unidades de tiempo. Son válidos por ejemplo T#5h10s y 500h.

El valor combinado de todos los valores de unidad especificados no puede superar los límites superior o inferior en milisegundos para el tipo de datos Time (-2.147.483,648 ms a +2.147.483,647 ms) [9].

A.2.2. Fecha

DATE se guarda como valor entero sin signo y se interpreta como número de días agregados a la fecha patrón 01/01/1990 para obtener la fecha específica. El formato del editor debe especificar un año, un mes y un día [9].

A.2.3. Tod

TOD (TIME_OF_DAY) se guarda como entero sin signo y se interpreta como el número en milisegundos desde medianoche para obtener la hora específica del día (medianoche = 0 ms). Deben especificarse hora (24h/día), minuto y segundo. Las fracciones de segundo son opcionales [9].

A.2.4. Datos DTL

Este tipo de datos DTL (fecha y hora largo) utiliza una estructura de 12 bytes para guardar información sobre la fecha y hora. DTL se puede definir en la memoria temporal de un bloque o en un DB. Debe indicarse un valor para todos los componentes de la columna “valor inicial” del editor de DB [9].

Tabla A. 2: Tamaño y rango para un DTL.

Longitud (bytes)	Formato	Rango de valores	Ejemplo de un valor de entrada
12	Reloj y calendario Año-Mes-Día:Hora:Minuto: Segundo.Nanosegundos	Mín.: DTL#1970-01-01-00:00:00.0 Máx.: DTL#2554-12-31-23:59:59.999 999 999	DTL#2008-12-16-20:30:20.250

Todo componente de DTL contiene un tipo de datos y un rango de valores diferentes. El tipo de datos de un valor especificado debe concordar con el tipo de datos de los componentes correspondientes [9].

Tabla A. 3: Elementos de la estructura DTL.

Byte	Componente	Tipo de datos	Rango de valores
0	Año	UINT	1970 a 2554
1			
2	Mes	USINT	1 a 12
3	Día	USINT	1 a 31
4	Día de la semana [1]	USINT	1(domingo) a 7(sábado) [1]
5	Hora	USINT	0 a 23
6	Minuto	USINT	0 a 59
7	Segundo	USINT	0 a 59
8	Nanosegundos	UDINT	0 a 999 999 999
9			
10			
11			

El día de la semana no se tiene en cuenta en la entrada del valor.

A.3. Contactores

Contactos normalmente abiertos y normalmente cerrados: Los contactos se pueden conectar a otros contactos, creando así una lógica combinacional propia. Si el bit de entrada indicado utiliza el identificador de memoria I (entrada) o Q (salida), el valor de bit se lee de la memoria imagen de proceso. Las señales de los contactos físicos del proceso controlado se cablean con los bornes de entrada del PLC. La CPU consulta las señales de entrada cableadas y actualiza continuamente los valores de estado correspondientes en la memoria imagen de proceso de las entradas [9].

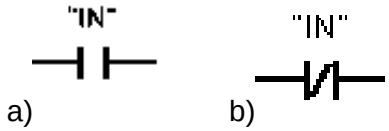


Figura A. 1: Símbolo de contadores, a) normalmente cerrado, b) normalmente abierto.

A.4. Bobina de relé

En la programación FUP, las bobinas KOP se transforman en cuadros de asignación (= y /=), en los que se indica una dirección de bit para la salida del cuadro. Es posible

conectar las entradas y salidas del cuadro con otros cuadros lógicos, o bien introducir una dirección de bit [9].

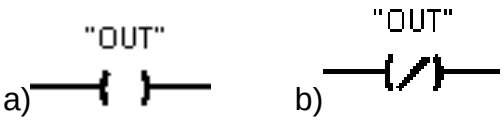


Figura A. 2: Símbolo de Bobinas, a) bobina de salida, b) bobina de salida negada.

A.5. Instrucciones activar/desactivar salida

Si se activa S (Set) el valor de datos de la dirección OUT se pone a 1. Si S no es está activado, OUT no cambia [9].

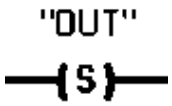


Figura A. 3: Símbolo de bobina con salida set.

Si se activa R (Reset), el valor de datos de la dirección de salida OUT se pone a 0.

Si no se activa R, no se modifica OUT [9].

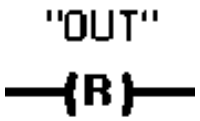


Figura A. 4: Símbolo de bobina con salida reset.

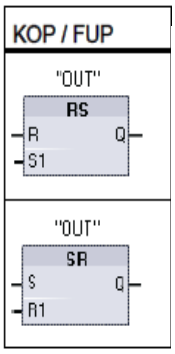
A.6. Flipflop de RS/SR

A.6.1. RS

RS es un flipflop en el que domina la activación. Si las señales de activación (S1) y desactivación (R) son verdaderas, la dirección de salida OUT se pone a 1 [9].

A.6.2. SR

SR es un flipflop en el que denomina la desactivación. Si las señales de activación (S) y desactivación (R1) son verdaderas, la dirección de salida OUT se pone a 0 [9].



Instrucción	S1	R	Bit "OUT"
RS	0	0	Estado anterior
	0	1	0
	1	0	1
	1	1	1
	S	R1	
SR	0	0	Estado anterior
	0	1	0
	1	0	1
	1	1	0

Figura A. 5. Símbolos de flipflop RS/SR.

A.7. Temporizador

Las instrucciones con temporizadores se utilizan para crear retardos programados. El número de temporizadores que pueden utilizarse en el programa de usuario está limitado sólo por la cantidad de memoria disponible en la CPU. Cada temporizador utiliza una estructura de DB del tipo de datos IEC_Timer de 16 bytes para guardar la información del temporizador especificada encima de la instrucción de cuadro o bobina. STEP 7 crea automáticamente el DB al introducir la instrucción.

Existen diferentes temporizadores los cuales son [9]:

- El temporizador TP genera un impulso con una duración predeterminada.
- El temporizador TON pone la salida Q a ON tras un tiempo de retardo predeterminado.
- El temporizador TOF pone la salida Q a OFF tras un tiempo de retardo predeterminado.
- El temporizador TONR pone la salida Q a ON tras un tiempo de retardo predeterminado. El tiempo transcurrido se acumula a lo largo de varios periodos de temporización hasta que la entrada R inicializa el tiempo transcurrido.

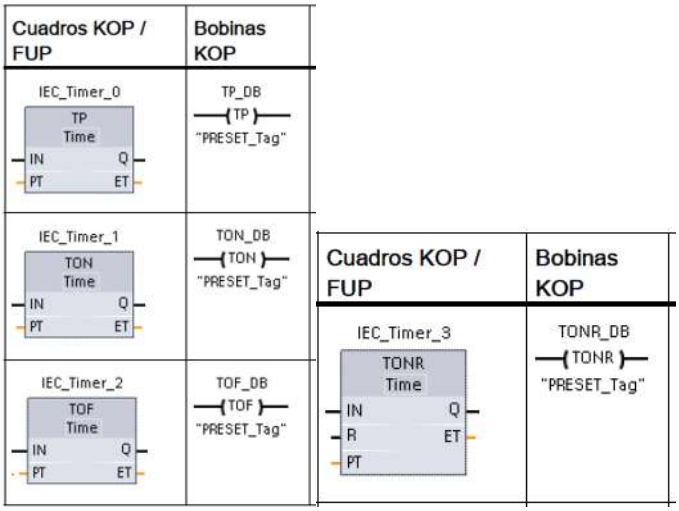


Figura A. 6: Símbolos de los temporizadores.

Tabla A. 4: Tamaño y rango del tipo de datos TIME.

Tipo de datos	Tamaño	Rangos válidos[1]
TIME	32 bits, almacenados como datos DInt	T#-24d_20h_31m_23s_648ms hastaT#24d_20h_31m_23s_647ms Almacenado como -2.147.483.648 ms hasta +2.147.483.647 ms

A.8. Comparación

Compara varios elementos del mismo tipo de datos. Si la comparación de contactos KOP es TRUE (verdadera), se activa el contacto. Si la comparación de cuadros FUP es TRUE (verdadera), la salida del cuadro es TRUE [9].

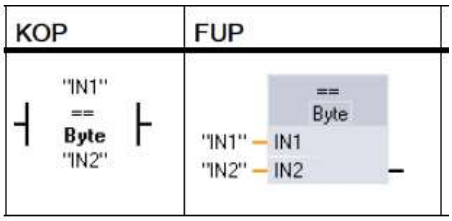


Figura A. 7: Símbolo de comparación.

Tabla A. 5: Tipos de datos para los parámetros.

Parámetro	Tipo de datos	Descripción
IN1, IN2	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, String, Char, Time, DTL, constante	Valores que deben compararse

A.9. Instrucciones avanzadas

A.9.1. Ajustar y leer el reloj del sistema

Las instrucciones de reloj se utilizan para ajustar y leer el reloj del sistema de la CPU. El tipo de datos DTL se utiliza para proporcionar valores de fecha y hora [9].

RD_SYS_T (leer la hora) lee la hora actual del sistema de la CPU. Este valor de hora no incluye la diferencia con respecto a la hora local ni tampoco al horario de verano [9].

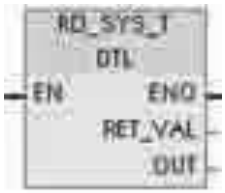


Figura A. 8: Símbolo RD_SYS_T.

RD_LOC_T (leer hora local) proporciona la hora local actual de la CPU como tipo de datos DTL. Este valor de tiempo refleja la zona horaria local ajustada correctamente según el cambio de horario de verano/invierno (si está configurado) [9].

Figura A. 9: Símbolo de RD_LOC_T.

Tabla A. 6: Tipos de datos para los parámetros.

Parámetro y tipo		Tipo de datos	Descripción
IN	IN	DTL	Hora que debe ajustarse en el reloj del sistema de la CPU
RET_VAL	OUT	Int	Código de condición de ejecución
OUT	OUT	DTL	RD_SYS_T: Hora actual del sistema de la CPU RD_LOC_T: Hora local actual incluido cualquier ajuste correspondiente al cambio de horario de verano/invierno, si está configurado

- La hora local se calcula aplicando la zona horaria y las variaciones correspondientes al horario de verano/invierno ajustadas en los parámetros de la ficha de configuración del dispositivo "Hora" [9].
- La configuración de la zona horaria es un offset respecto a la hora UTC o GTM [9].
- La configuración del horario de verano especifica el mes, semana, día y hora de comienzo del horario de verano [9].
- La configuración del horario estándar especifica el mes, semana, día y hora de comienzo del horario estándar [9].
- La diferencia con respecto a la zona horaria se aplica siempre al valor de hora del sistema. La diferencia con respecto al horario de verano se aplica únicamente si el horario de verano está en vigor [9].

Nota

Horario de invierno/verano y configuración del tiempo de inicio estándar

La propiedad "Hora del día" para el "Inicio del horario de verano" de la configuración de dispositivo de la CPU debe ser la hora local.

T_CONV convierte un tipo de datos Time en un tipo de datos DInt, o a la inversa, el tipo de datos DInt en el tipo de datos Time [9].

Figura A. 10: Símbolo T_CONV.

Tabla A. 7: Tipos de datos para los parámetros T_CONV.

Parámetro y tipo		Tipo de datos	Descripción
IN	IN	DInt, Time	Valor de entrada Time o valor DInt
OUT	OUT	DInt, Time	Valor convertido DInt o valor Time

A.10. Visualizador HMI

Puesto que la visualización se está convirtiendo cada vez más en un componente estándar de la mayoría de las máquinas, los SIMATIC HMI Basic Panels ofrecen dispositivos con pantalla táctil para tareas básicas de control y supervisión. Todos los paneles ofrecen el grado de protección IP65 y certificación CE, UL, cULus y NEMA 4x. En el proyecto se utilizó la pantalla hmi KTP 700 basic PN [9].

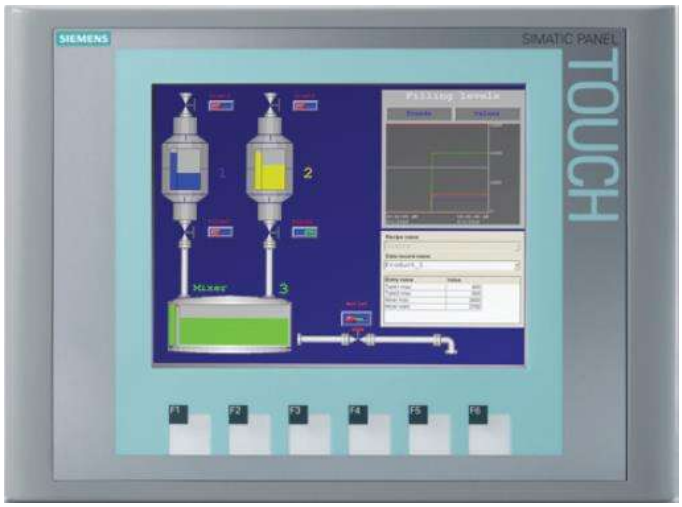


Figura A. 11: Pantalla HMI.

Pantalla táctil de 6 pulgadas con 6 teclas táctiles, nos ofrece las siguientes opciones:

- Color (TFT, 256 colores) o monocromo (STN, escala de grises).
- 115,2 mm x 86,4 mm (5,7") Horizontal o vertical.
- Resolución: 320 x 240.
- 500 variables.
- 50 sinópticos de proceso.
- 200 alarmas.
- 25 curvas.
- 32 KB memoria de recetas.
- 5 recetas, 20 registros, 20 entradas

A.11. Herramientas de visualización en la HMI

A.11.1. Circulo

El objeto "Círculo" es un objeto cerrado que se puede rellenar con un color o trama [10].

Figura A. 12: Visualizador de circulo.

A.11.2. Botón

El objeto "Botón" permite configurar el objeto con el que el operador ejecuta en runtime una función configurable cualquiera [10].

Figura A. 13: Visualizador de botón.

A.11.3. Campo de texto

El objeto "Campo de texto" es un objeto cerrado que puede rellenarse con un color [10].

Figura A. 14: Visualizador de campo de texto.

A.11.4. Visor de gráficos

El objeto "Visor de gráficos" permite visualizar gráficos [10].

Figura A. 15: Visualizador de visor de gráficos.

A.11.5. Campo E/S

El objeto "Campo E/S" permite la entrada y la visualización de valores de proceso [10].

Figura A. 16: Visualizador de campo E/S.

A.11.6. Rectángulo

El objeto "Rectángulo" es un objeto cerrado que puede rellenarse con un color [10].

Figura A. 17: Visualizador de rectángulo.

Bibliografía

[1] R. Valdés Fernández, Automatización de un sistema de climatización con PLC., 2012.

[2] E. Mandado Perez, J. Marcos Acevedo, C. Fernández Silva y J. Armesto Quiroga, automatas programables y sistemas de automatización, alfaomega.

[3] P. Aguilera Martinez, programacion de PLC´s, 2002.

[4] J. Celis Peñaranda, C. Escobar Amado, S. Sepulveda Mora, S. Castro Casadiego, B. Medina Delgado y J. Ramirez Mateus, Crontrol adaptativo para optimizar una intersección basado en un sistema embebido, 2016.

[5] H. D. Osorio Unda , Implementación de un sistema de automatización para el control de semaforos inteligentes, 2018.

[6] M. Martínez Anorozo, Semáforos inteligentes.

[7] C. Chavez, Sistema de semaforización inteligente para el control de flujo vehicular mediante el Procesamiento Digital de Imagenes, 2015.

[8] «https://lavozdemichoacan.com.mx/morelia/sin-control-crecimiento-de-parque-vehicular-en-morelia/,» [En línea].

[9] SIEMENS, «S7 Controlador programable S7-1200,» [En línea].

[10] «ayuda, TIA Protal V13, COmponentes,».