

HERRAMIENTA BASADA EN AUTÓMATAS PARA DISEÑO E
IMPLEMENTACIÓN CON PLC, DE SUPERVISORES DE SISTEMAS DE
EVENTOS DISCRETOS.
ANEXOS



Leydy Viviana Muñoz Robles
Lenin Galindez Trochez

UNIVERSIDAD DEL CAUCA
FACULTAD DE INGENIERÍA ELECTRÓNICA Y
TELECOMUNICACIONES
DEPARTAMENTO DE ELECTRÓNICA, INSTRUMENTACIÓN Y
CONTROL
POPAYÁN
2010

HERRAMIENTA BASADA EN AUTÓMATAS PARA DISEÑO E
IMPLEMENTACIÓN CON PLC, DE SUPERVISORES DE SISTEMAS DE
EVENTOS DISCRETOS.

ANEXOS

Leydy Viviana Muñoz Robles
Lenin Galindez Trochez

Director
PhD. Carlos Alberto Gaviria López

UNIVERSIDAD DEL CAUCA
FACULTAD DE INGENIERÍA ELECTRÓNICA Y
TELECOMUNICACIONES
DEPARTAMENTO DE ELECTRÓNICA, INSTRUMENTACIÓN Y
CONTROL
POPAYÁN
2010

TABLA DE CONTENIDO

PREFACIO	0
ANEXO B. MANUAL DE USUARIO SOFTWARE FSMLADDER.....	1
1. Introducción al programa FSMLADDER.....	1
1a. ¿Qué es FSMLADDER?.....	1
2. Requerimientos.....	1
3. Área de trabajo de FSMLADDER	2
3a. Opción Archivo	2
3b. Opción Ejecutar	4
3c. Opción Ladder.	6
3d. Opción Ayuda.....	7
4. Árbol indicador.	8
5. Ventana de asignación de variables.	9
5a. ¿Cómo asignar y sustituir variables?.....	10
6. Indicadores de direcciones entradas y salidas	10
7. Ventana de Consola y Problemas	11
8. Botón de inicio e indicadores.....	11
8. ¿Cómo implementar un autómata temporizado?.....	12
9. ¿Cómo implementar un contador?	15
10. Ejemplo práctico.	17
10a. Descripción del proceso.	18
Autómatas de los sensores	20
Autómata de las restricciones.....	21
Cargar archivos en FSMLADDER.....	22
Simulación en Uppal.....	23
Generar código ladder.....	25
ANEXO A. MANUAL DE USUARIO HERRAMIENTA DESUMA.....	29
1. Crear un autómata.....	29

2.	<i>Guardar un autómata creado y eliminar un evento o estado.....</i>	30
3.	<i>Eliminar y editar un evento o un estado.....</i>	30
3a.	<i>Eliminar una transición o evento de un autómata.....</i>	30
3b.	<i>Eliminar un estado de un autómata.</i>	31
3c.	<i>Editar las propiedades de los estados dentro de un autómata.</i>	31
3d.	<i>Editar las propiedades de los eventos dentro de un autómata.</i>	32
4.	<i>Operación de composición paralela de dos o más autómatas.....</i>	32
	<i>ANEXO C. METODOLOGIA DE MODELADO.....</i>	34

PREFACIO

El documento presentado a continuación, el manual de usuario del software FSMLADDER, una herramienta construida por los autores de este documento, utilizando el lenguaje de programación JAVA bajo el IDE ECLIPSE y utilizando la librería SWT para la interfaz grafica, esta herramienta ofrece una manera eficiente y segura de establecer una conversión entre la lógica que manejan las maquinas de estado finito (autómatas finitos) y el lenguaje de programación Ladder. En el Anexo A se presenta un manual de usuario del software DESUMA, una herramienta desarrollado en la Universidad de Mount Allison en el lenguaje de programación JAVA, que permite el diseño de autómatas de sistemas de eventos discretos. El anexo C por su parte contiene una metodología denominada SUPERVISIÓN BASADA EN UN MODELO COMPORTAMENTAL DEL PROCESO [26], diseñada en conjunto por la Universidad Politécnica de Cataluña ubicada en Barcelona – España, y el Laboratorio de Análisis y Arquitectura de Sistemas (LAAS) de Toulouse – Francia, la cual propone utilizar el conocimiento heurístico que el experto posee para modelar un proceso, mediante la construcción de varios autómatas finitos y a partir de ellos obtener un único autómata supervisor que determinara el comportamiento de un proceso.

Los Autores.

ANEXO B. MANUAL DE USUARIO SOFTWARE FSMLADDER

1. Introducción al programa FSMLADDER

1a. ¿Qué es FSMLADDER?

El programa FSMLADDER, cuyo nombre se deriva de las siglas en inglés “finite state machine” (maquinas de estados finito) y la palabra LADDER (escalera), ofrece una manera eficiente y segura de establecer una conversión entre la lógica que manejan las maquinas de estado finito (autómatas finitos) y el lenguaje de programación Ladder, en este caso en particular lo que se pretende con esta aplicación es traducir el lenguaje que genera la herramienta de maquinas de estado finito DESSUMA al lenguaje de lógica ladder para la aplicación software RS_logix 500. El presente documento pretende ser una guía para los usuarios en el que se busca tener un manual de uso rápido y sencillo que permita resolver las dudas más comunes en lo que se refiere a la herramienta de uso libre FSMLADDER.

2. Requerimientos

La herramienta FSMLADDER es una aplicación desarrollada en el entorno de trabajo SDK eclipse 3.2, por lo tanto es necesario que el ordenador tenga instalado una versión de java igual o superior a la 1.41, cumpliendo con este requisito se puede ejecutar FSMLADDER en ordenadores que usen plataformas Windows que tengan las especificaciones listadas a continuación.

- Windows 95 SE, Windows 98, Windows Me, Windows 2000, Windows XP o Windows 7.
- Procesador Pentium II 450 MHz como requerimiento mínimo.
- 50 MB de espacio en su disco duro.
- 64 MB de memoria RAM (Se recomienda 128 MB o más).
- Unidad lectora de CD (sólo para ejecutar FSMLADDER desde un CD).
- Resolución de pantalla de 800x600 o más.
- Entorno Java (JRE) versión 1.4.1_01 o más.
- **Nota:** Cumpliendo con estos requisitos mínimos el usuario solo tiene que ejecutar la aplicación, ya sea desde una ubicación de disco duro o desde una unidad de Cd en donde se encuentre el ejecutable de la herramienta.

3. Área de trabajo de FSMLADDER

Una vez que haya ingresado a FSMLADDER se podrá observar el siguiente espacio de trabajo.

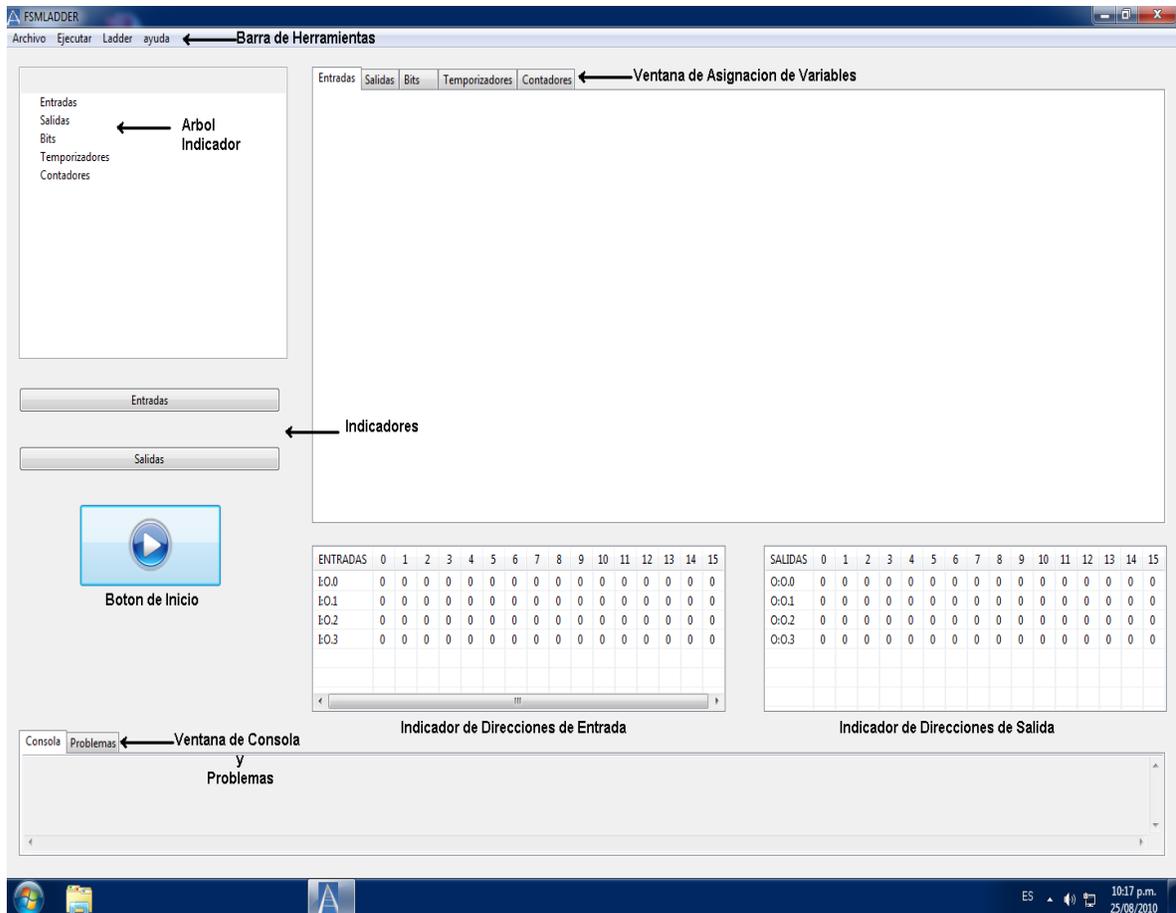


Figura 1: Espacio de trabajo (FSMLADDER).

A continuación se hará una descripción detallada de cada una de las partes que conforman el espacio de trabajo.

3a. Opción Archivo

En el momento que se selecciona la opción "archivo" se despliega un pequeño submenú en él que se encuentran funciones tales como:

- **Abrir:** función encargada de abrir los archivos (autómatas) que quieren procesar.
- **Directorio:** Función necesaria para establecer un directorio de trabajo.
- **Salir:** Función que cierra la aplicación.



Figura 2: Opción Archivo.

Al momento de iniciar el ítem “abrir”, el programa despliega una venta en la cual el usuario podrá seleccionar los archivos que desea convertir a la lógica ladder.

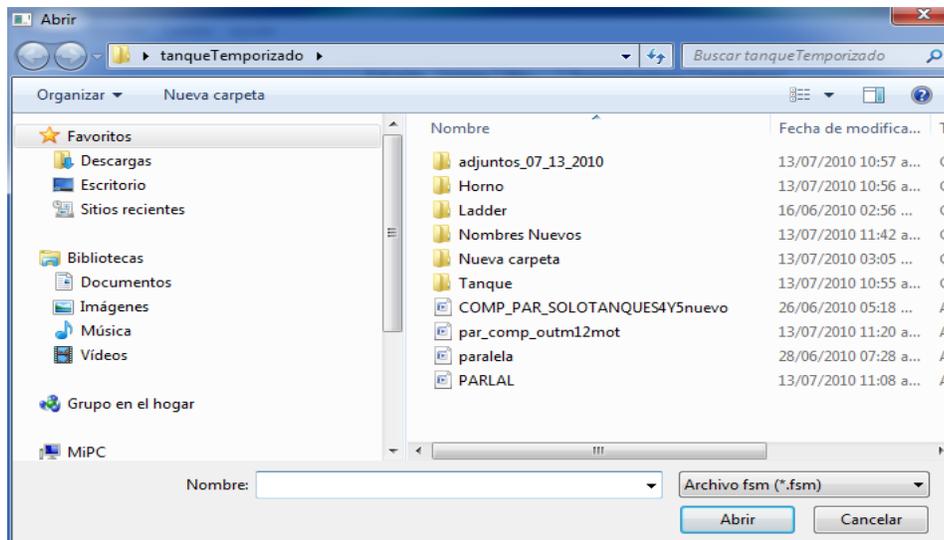


Figura 3: Opción Abrir.

Como se puede apreciar en la grafica (Figura 3) solo es posible abrir archivos que tengan extensión “fsm” (formato DESUMMA).

Cuando el usuario haya seleccionado los archivos y posteriormente tome la opción “Abrir” de la ventana, el programa iniciara la composición paralela con los archivos seleccionados y desplegara una nueva ventana en la que se pedirá que se le asigne un nombre al resultado de la composición paralela.

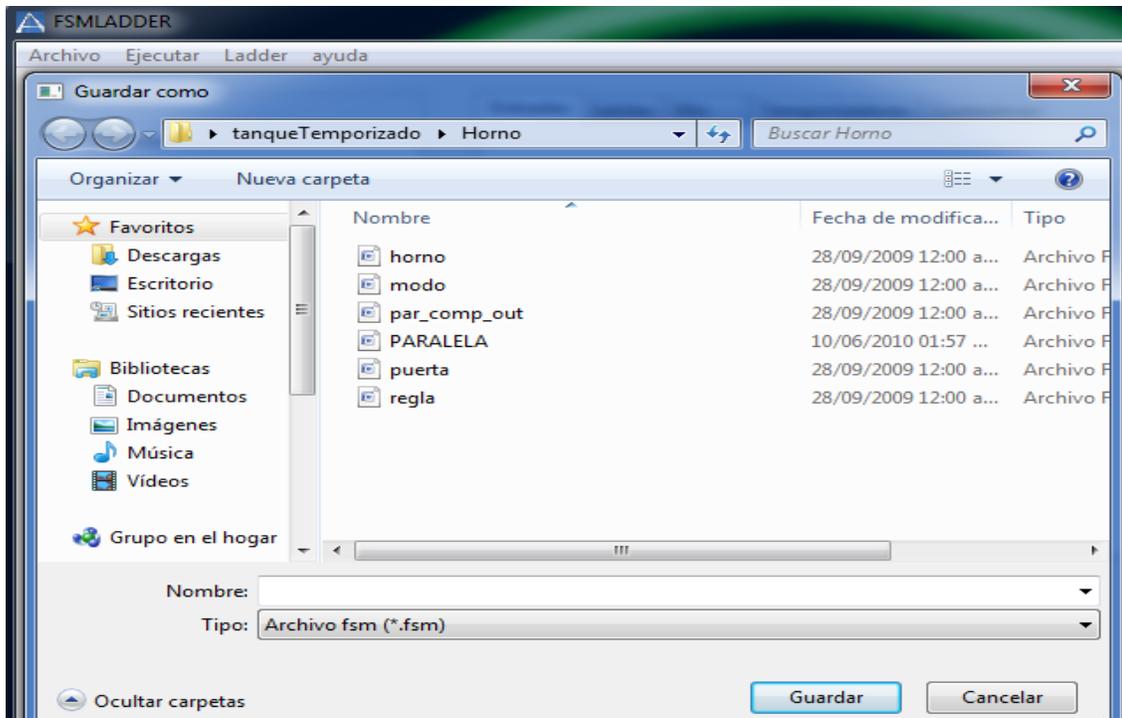


Figura 4: Almacenar composición paralela.

Si el usuario toma la opción “directorio” el programa habilita una ventana en la cual se podrá seleccionar una dirección en donde se desea que los archivos resultantes de la conversión (composición paralela) queden almacenados.

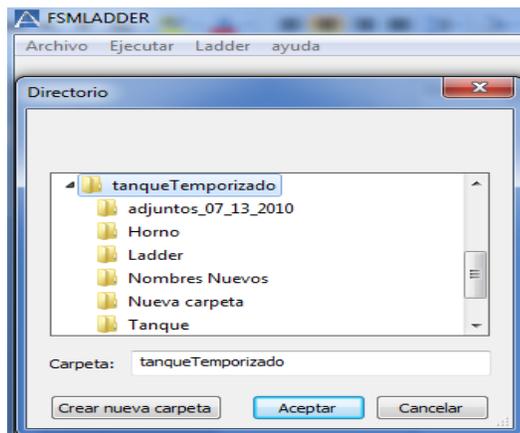


Figura 4: Opción Directorio.

3b. Opción Ejecutar

Esta sección de la aplicación consta de un submenú el cual se despliega en el momento que el usuario accede a la opción “Ejecutar”, dentro de este se muestra un ítem llamado “Composición paralela” y posteriormente si es seleccionado se mostrara otro menú el cual consta de 2 opciones las cuales son “nuevo” y “actual”.

La opción “Ejecutar” tiene como finalidad hacer una simulación del autómata bajo la herramienta de simulación de maquinas de estado finito “UPPAAL” con el propósito de detectar posibles bloqueos.

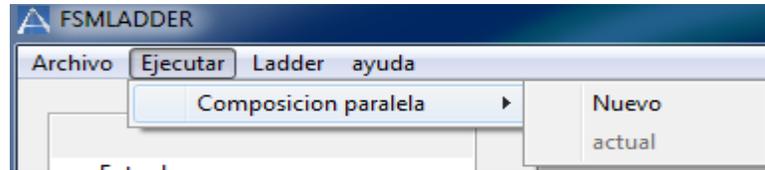


Figura 5: Opción Ejecutar.

Nuevo: Esta opción al ser utilizada despliega una ventana de selección, en ella el usuario puede seleccionar el archivo que se desea ser simulado en la herramienta UPPAAL.

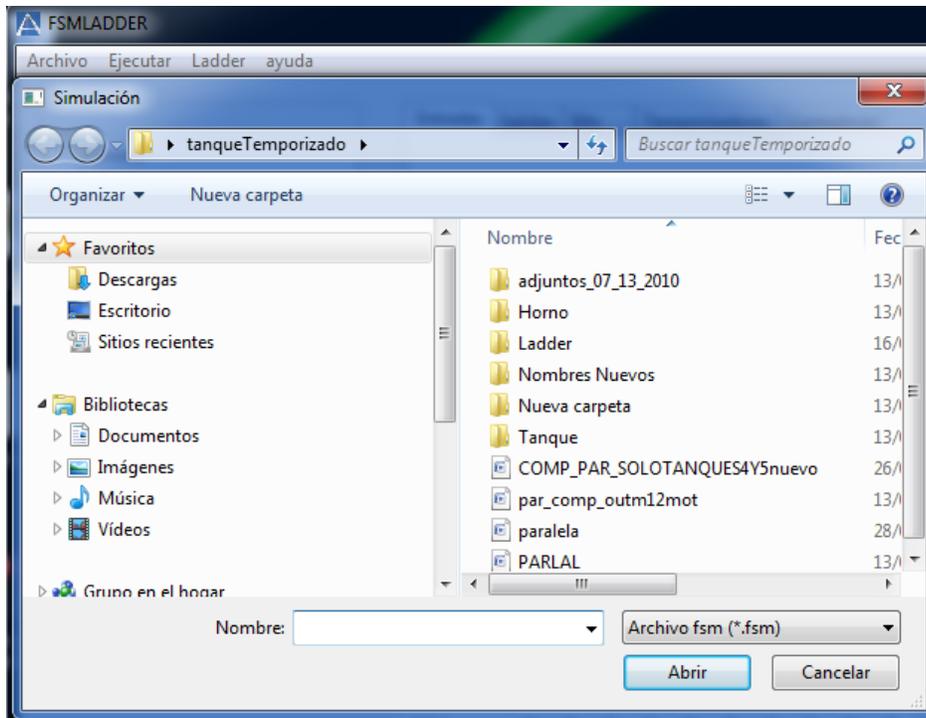


Figura 6: Ventana para abrir archivo de simulación.

Actual: Esta opción inicia la simulación del archivo producto de la composición paralela, es decir de los archivos que se hayan seleccionado en la opción “Archivo”.

Nota: La opción Actual solo estará habilitada, siempre y cuando el usuario haya hecho la composición paralela a partir de la opción “Abrir” que se encuentra en el menú de “Archivo”.

Independientemente de la opción que tome el usuario, el programa ejecutara la herramienta “Uppaal” y mostrara el archivo en el simulador de autómatas finitos.

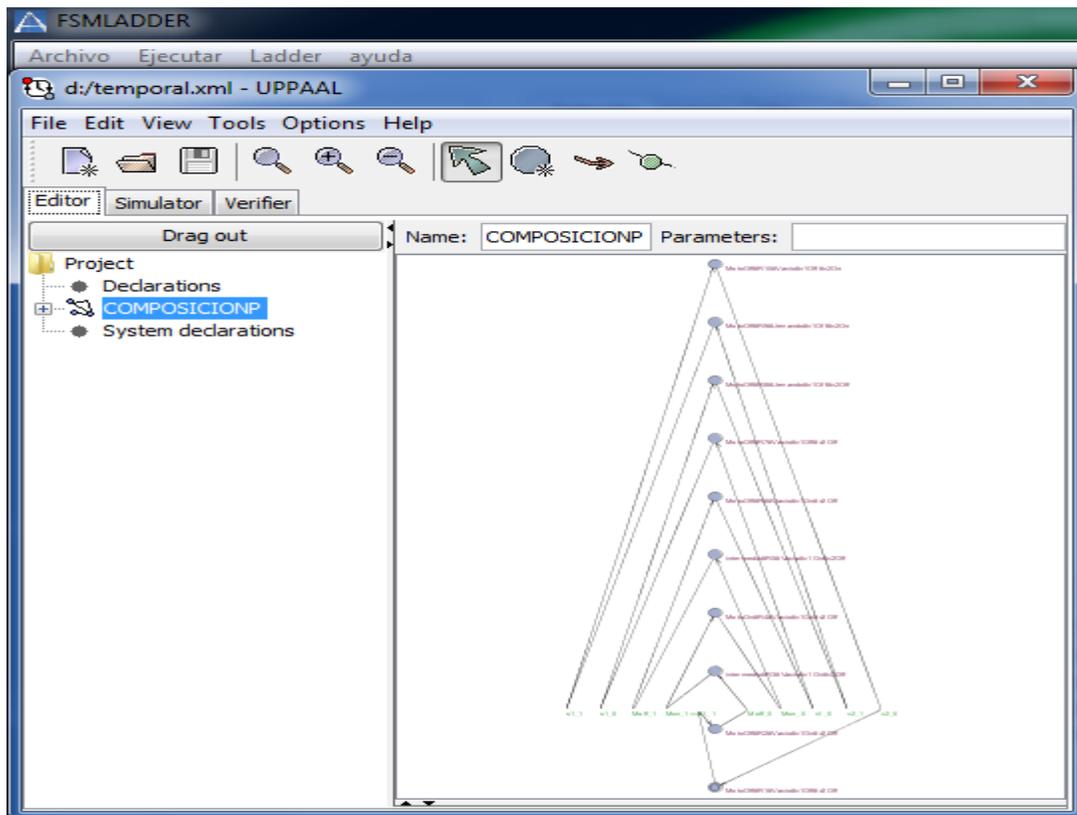


Figura 7: Simulación en Uppaal.

Nota: La herramienta FSMLADDER se encuentra deshabilitada mientras la aplicación “Uppaal” se encuentre abierta.

Nota: Al momento de cerrar la herramienta de simulación “Uppaal”, la aplicación FSMLADDER automáticamente guardara el archivo con el mismo nombre pero diferirá de la extensión (XML).

3c. Opción Ladder.

Al igual que la opción “Ejecutar” de la barra de opciones, el ítem “Ladder” tiene un submenú en el cual el usuario podrá abrir un archivo nuevo con extensión xml o uno actual producto de la simulación de la herramienta “Uppaal”.

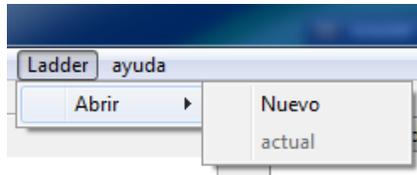


Figura 8: Opción Ladder.

Nota: La función actual solo se habilitara sin el usuario ha utilizado la opción ejecutar de la barra de herramientas.

En caso de que el ítem seleccionado sea “Nuevo”, el programa a continuación desplegara una ventana en la cual el usuario podrá seleccionar el archivo que desea convertir a la lógica ladder.

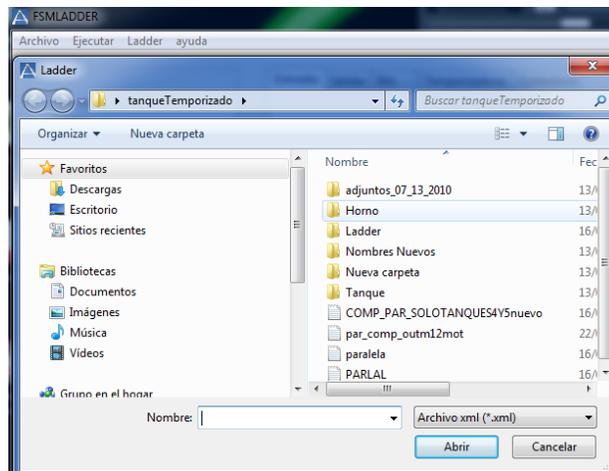


Figura 9: abrir autómata en programa UPPAAL.

3d. Opción Ayuda.

Este ítem que conforma la barra de herramientas de la herramienta FSMLADDER contiene un submenú el cual consta de 2 opciones las cuales contiene accesos directos a la guía de usuario e información básico de la herramienta FSMLADDER (versión) .

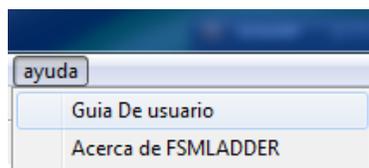


Figura 10: Opción ayuda.

Nota: Si el usuario desea buscar una palabra en particular dentro de la ayuda, solo debe presionar el comando “Ctrl+f”.

4. Árbol indicador.

Esta ventana que se encuentra en la parte derecha de la pantalla tiene como función principal mostrar las variables que contiene el autómata que haya sido abierto por medio de la opción “Ladder”, además de poder desplazarse por medio de un click por cada una de las pestañas de la ventana de asignación de variables que se encuentra en la parte izquierda de la pantalla.



Figura 11: Árbol indicador.

Como se observa en la grafica (Figura11), el árbol indicador está dividido en 5 secciones las cuales son:

- Entradas.
- Salidas.
- Bits.
- Temporizadores.
- Contadores.

En el momento que el usuario cargue un archivo por medio de la opción “Ladder”, el árbol automáticamente actualizara el contenido de cada una de las secciones de acuerdo a las variables que estén definidas en el autómata.

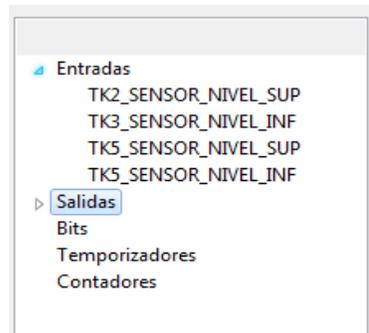


Figura 12: Árbol con contenido.

5. Ventana de asignación de variables.

Esta ventana que hace parte de la herramienta es la encargada de almacenar las direcciones ladder que el usuario asignara a cada una de las variables de entrada y salida que el autómata posea en su lógica. Esta se encuentra dividida en 5 pestañas las cuales al momento de ejecutar la aplicación se encuentran en blanco.



Figura 13: Ventana de asignación de variables.

En el momento que el usuario cargue el correspondiente archivo en la opción ladder, las diferentes pestañas de la ventana se actualizaran y mostraran el nombre de cada una de las variables del autómata, así como también a su lado una barra desplegable en donde se mostrara las direcciones en un formato compatible con la herramienta software RS-LOGIX 500.



Figura 14: Asignación de direcciones.

5a. ¿Cómo asignar y sustituir variables?

Para que el usuario asigne una dirección es necesario seleccionar una de las direcciones que se encuentra en la barra desplegable que acompaña a la variable, seguidamente tendrá que dirigir el cursor del mouse sobre el ítem y oprimir click derecho, si el proceso ha sido exitoso se observara que la barra desplegable se encontrara deshabilitada. En caso de que el usuario quiera cambiar de dirección solo tendrá que hacer doble click sobre el icono y automáticamente se mostrara una ventana de advertencia en donde se le preguntara si está seguro de que quiere hacer dicha operación, si la respuesta es afirmativa automáticamente la barra desplegable queda en blanco y se habilita nuevamente.



Figura 15: Dirección Asignada.

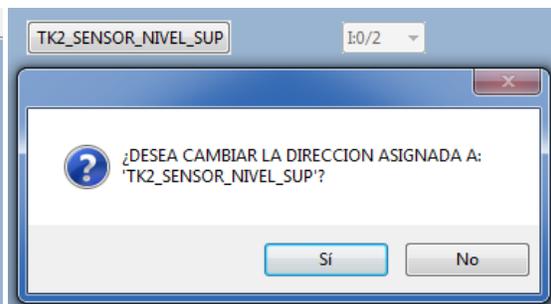


Figura 16: Cambio de dirección.

6. Indicadores de direcciones entradas y salidas

Este par de ventanas tienen como finalidad mostrar al usuario las direcciones de entrada y salida que se encuentran disponibles, en el momento que se asigne una de las direcciones de entrada o salida a una de las variables que se muestran en la ventana de asignación de variables, automáticamente se mostrara que la dirección ha sido asignada ya sea en la ventana de entradas o de salidas.

ENTRADAS	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	SALIDAS	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
I:0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	O:0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
I:0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	O:0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
I:0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	O:0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
I:0.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	O:0.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figura 17: Indicadores de direcciones de entrada y de salida

Cuando el usuario asigna una dirección se mostrara en la ventana la dirección asignada resaltando una de las casillas con un color azul (en w7).

ENTRADAS	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
I:O.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
I:O.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
I:O.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
I:O.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figura 17: Dirección Seleccionada.

7. Ventana de Consola y Problemas

Esta ventana se encuentra ubicada en la parte inferior de la ventana principal de la herramienta, en ella se encuentran las pestañas de “consola” y “problemas”.

La pestaña de consola muestra todos los archivos que el usuario ha cargado en la aplicación, así como también informa la ubicación de los archivos que se han generado a partir de cada una de las aplicaciones que se encuentran en la barra superior del entorno de trabajo. También está encargada de informar cuando ha terminado cada una de las aplicaciones que el usuario haya ejecutado en la herramienta.

La pestaña de problemas tiene con fin mostrar que variables de la ventana de asignación de variables aun falta por asignar dirección, en caso de que el usuario oprima el botón de “inicio” de conversión que se encuentra debajo de la ventana que contiene el árbol indicador.

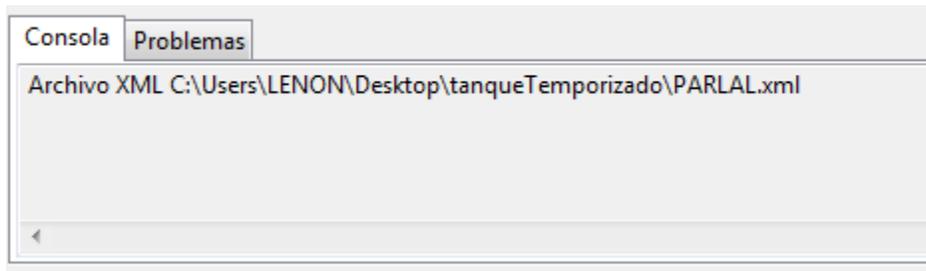


Figura 18: Ventana de Consola y problemas.

8. Botón de inicio e indicadores.

Esta parte de la herramienta se encuentra localizada en la parte derecha del área de trabajo, abajo de la ventana que contiene el árbol indicador. Esta se distingue por que aparecen tres botones dos de los cuales aparecen las palabras entradas y salidas dentro de ellos, su función consiste en informar al usuario que ya se ha

asignado a todas la variables las direcciones correspondientes, cuando esto ocurre el icono cambia de color.

NOTA: Solo se podrá iniciar la conversión siempre y cuando los iconos de entrada y de salida se encuentren iluminados.



Figura 19: Botón de inicio e indicadores.



Figura 20: Botón de inicio habilitado.

8. ¿Cómo implementar un autómata temporizado?

Para resolver este inconveniente es necesario desarrollar una sintaxis, la cual será aplicada en la herramienta de elaboración de autómatas “DESUMA”. Para ella se considera el siguiente ejemplo práctico.

Ejemplo: Se desea mantener en operación una válvula (V) por un determinado periodo de tiempo (4 segundos), los estados de autómata serán “V1encendido” y “V1apagado”.

Como primera parte se observara la sintaxis necesaria para poder generar el autómata temporizado, esta se puede apreciar en la siguiente figura

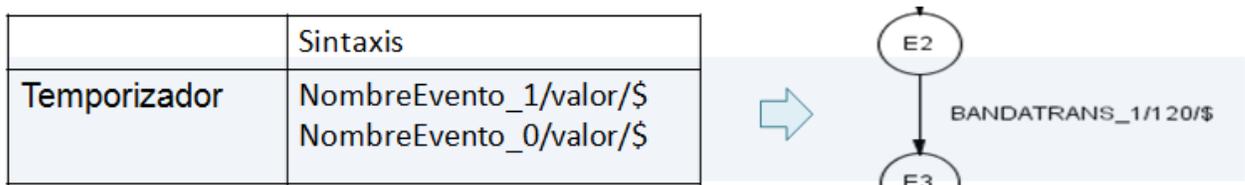


Figura 21: Sintaxis de un temporizador.

Con base a la grafica anterior se obtiene el siguiente autómata utilizando la herramienta “DESUMA”

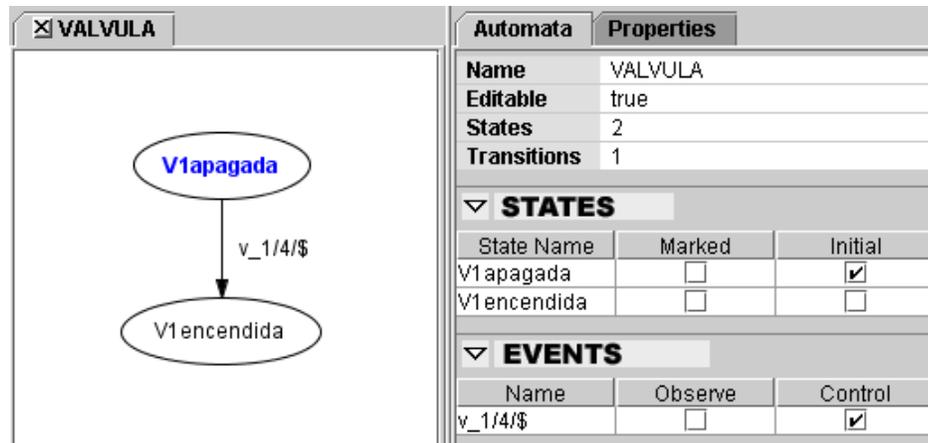
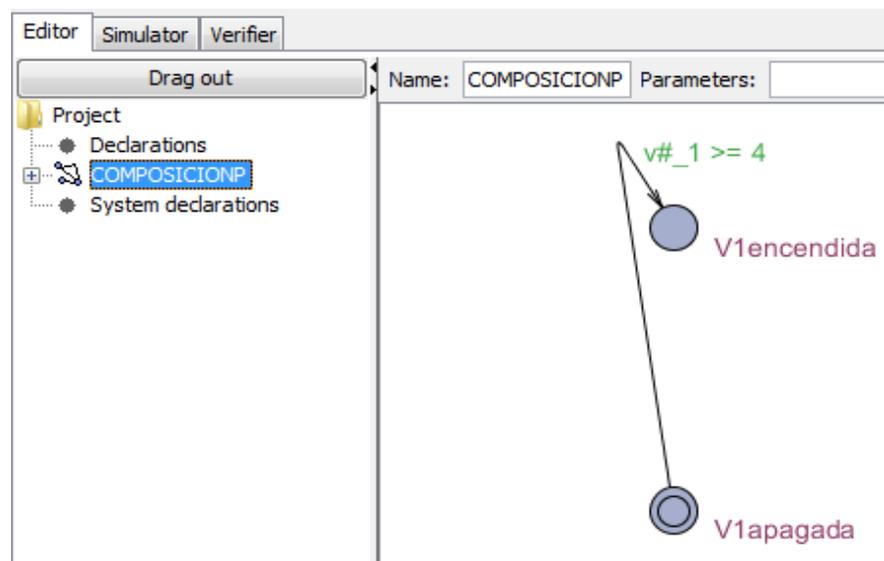


Figura 22: Autómata temporizado en “DESUMA”

Utilizando la herramienta “FSMLADDER” con la función de “composición paralela”, se hace la conversión del archivo “VALVULA.fsm” al formato de la herramienta “UPPAAL” y se obtiene el siguiente resultado:



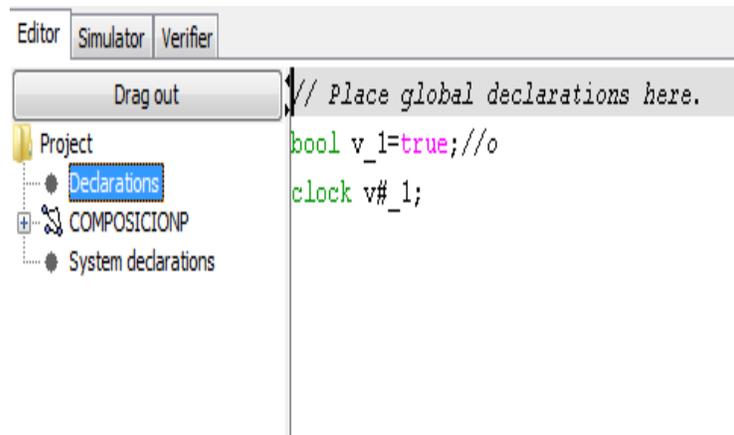


Figura 23: Autómata temporizado en “UPPAAL”

Ahora para ampliar el ejemplo anterior se agregara una sentencia tipo condición, esta consistirá en que solo se podrá acceder al siguiente estado siempre y cuando se haya cumplido un periodo de tiempo, para el ejemplo se considera 4 segundos. Para ello el usuario debe familiarizarse con la siguiente sintaxis que se mostrara en la grafica.

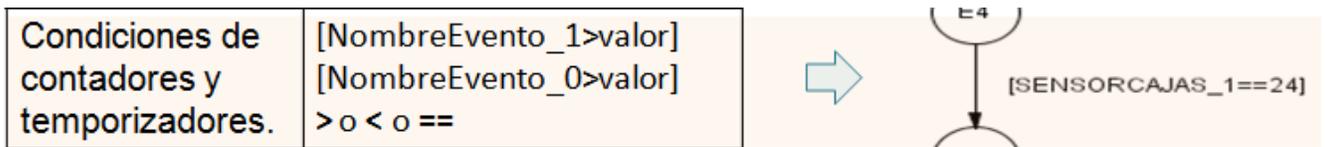


Figura 24: Sintaxis de una condición.

Tomando en cuenta la anterior sintaxis el autómata resultante es el siguiente:

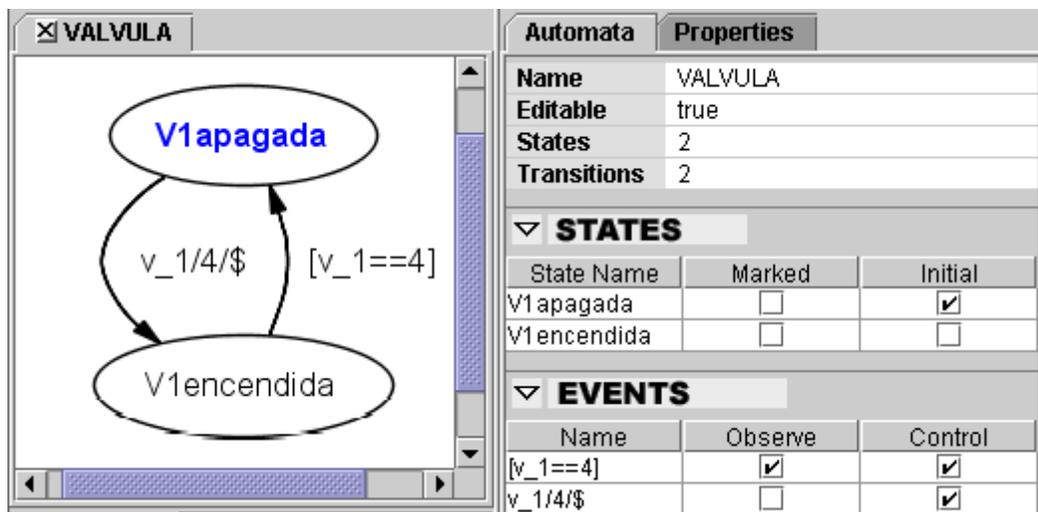


Figura 25: Autómata con condiciones en “DESUMA”

Como se puede apreciar en la figura el autómata “VALVULA” se ha agregado una sentencia tipo condición, en donde solo se puede acceder al estado “V1apagada” siempre y cuando se haya cumplido con un periodo de tiempo determinado.

Utilizando la herramienta “FSMLADDER” se obtiene el siguiente resultado:

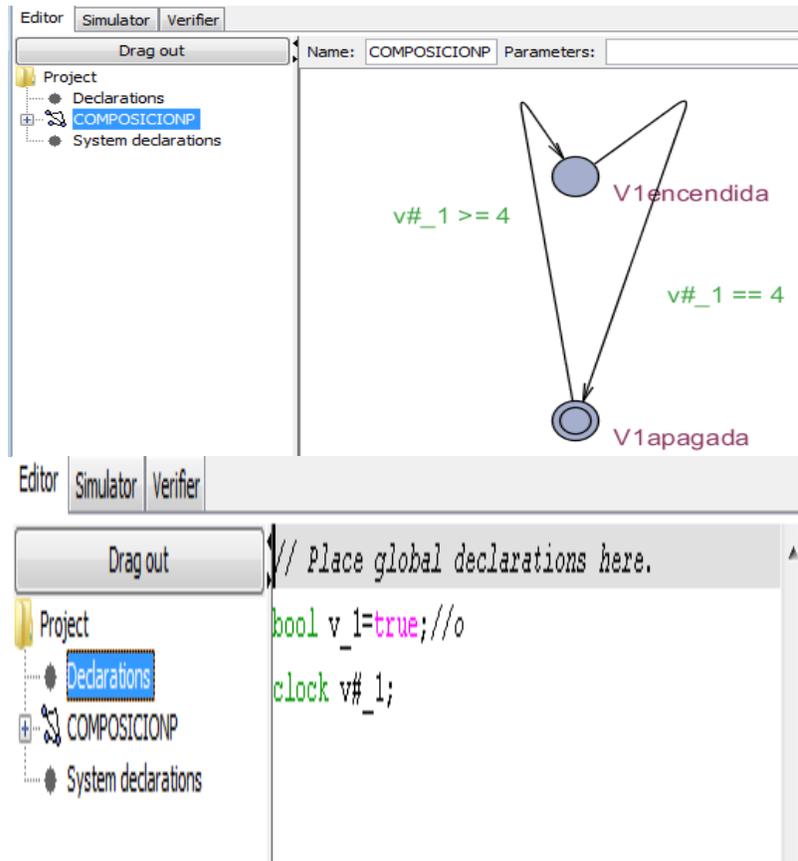


Figura 26: Autómata con condiciones en “UPPAAL”.

9. ¿Cómo implementar un contador?

Para resolver este problema se considera el siguiente ejemplo. Antes que todo, el usuario debe familiarizarse con la sintaxis que se muestra en la siguiente figura

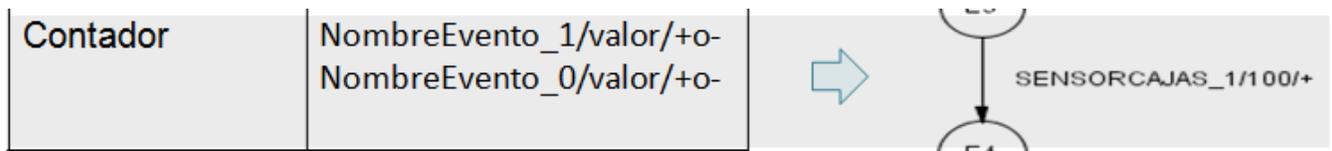


Figura 27: Sintaxis de un contador.

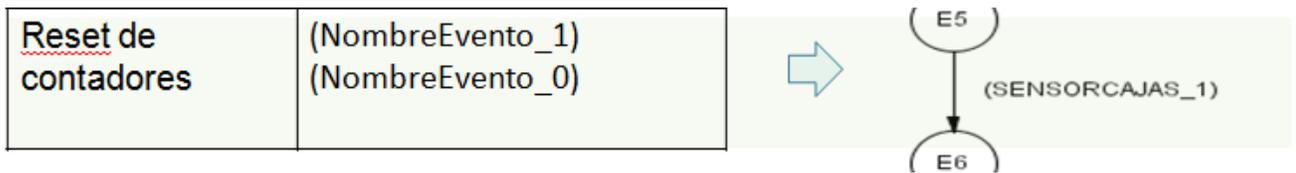


Figura 28: Sintaxis de un reset.

Ejemplo: Se desea encender una válvula “V” a partir de un número de veces que ocurra el evento “sensor”, en este caso se considera que el evento debe ocurrir 3 veces y el contador volverá a su valor inicial, es decir que el usuario deberá implementar una sentencia de reset.

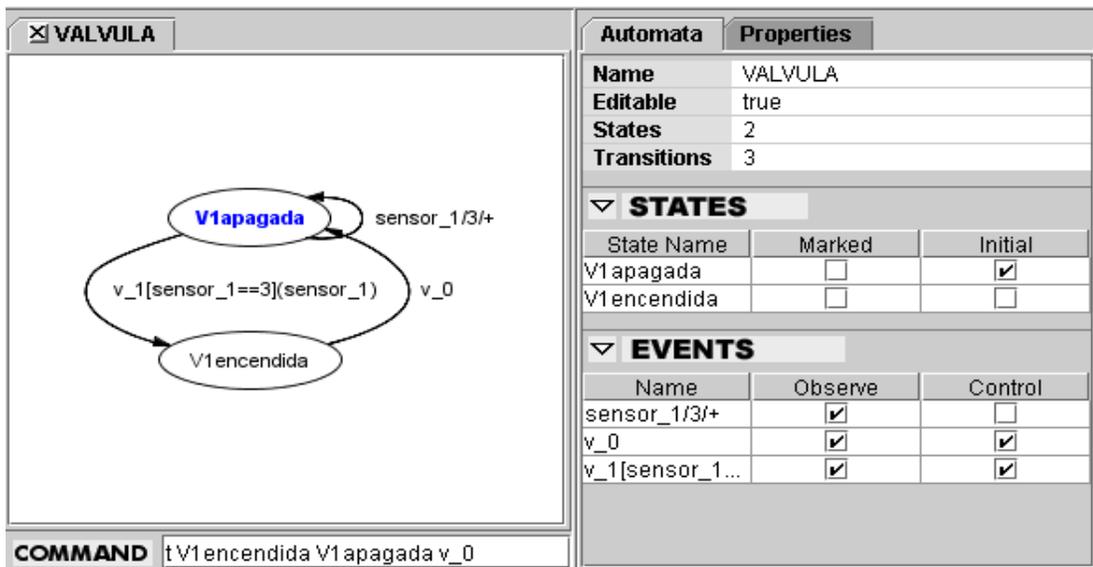


Figura 29: Autómata con contadores “DESUMA”.

Como se puede apreciar en la grafica anterior el autómata ya contiene la información solicitada en el ejemplo, ahora se procede a realizar la conversión al formato “UPPAAL” utilizando la herramienta “FSMLADDER”.

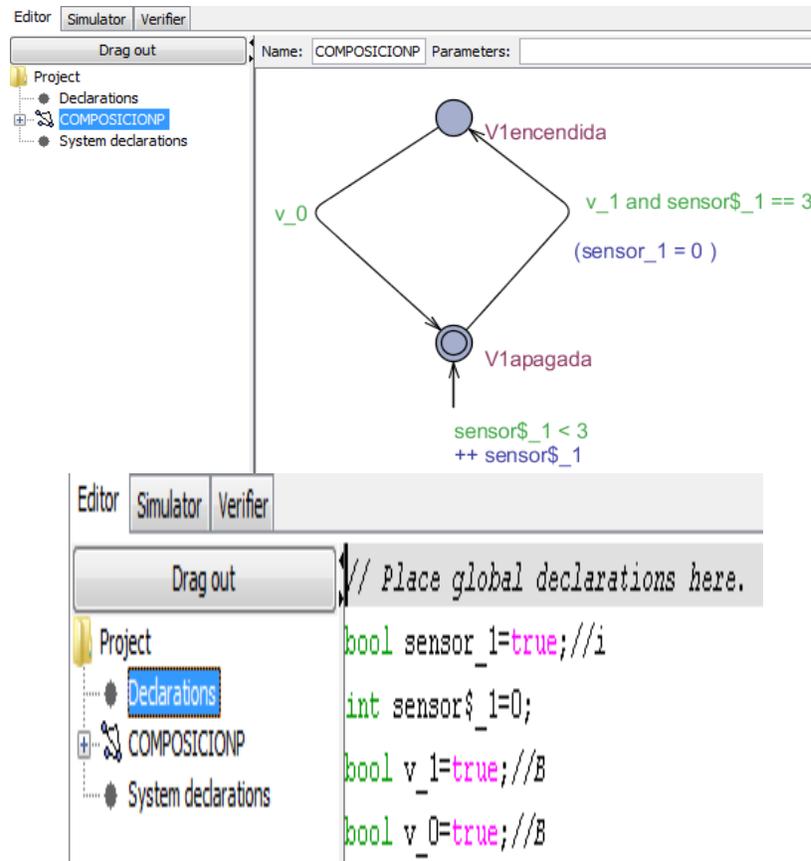


Figura 30: Autómata con contadores “UPPAAL”.

10. Ejemplo práctico.

A continuación se desarrollara un pequeño ejemplo en cual se mostrara en la siguiente grafica.

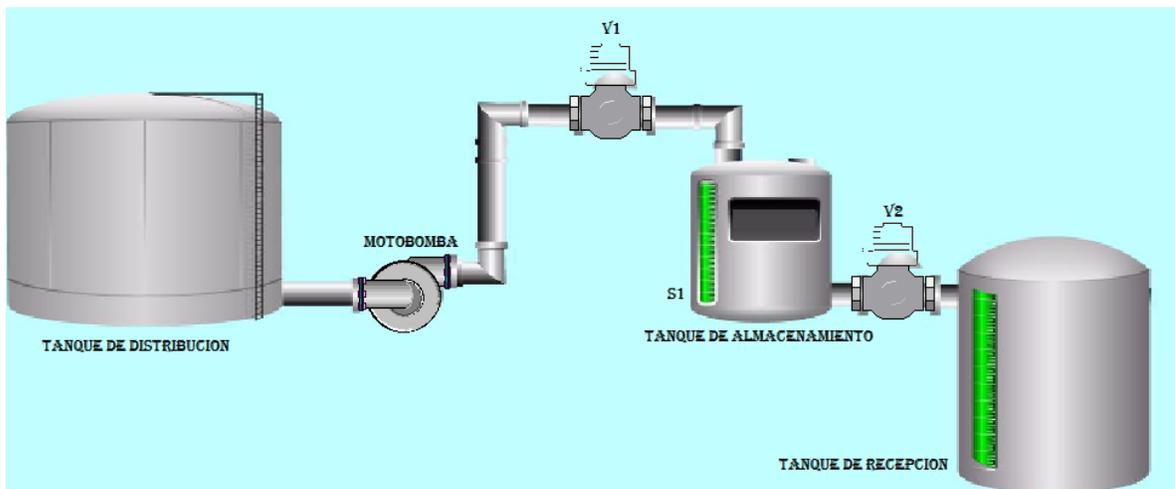


Figura 31: Ejemplo práctico.

El proceso que se muestra en la grafica está compuesto por los siguientes elementos:

- Tanque de almacenamiento.
- Tanque de distribución.
- Tanque de recepción.
- Motobomba: Es la encargada de hacer circular el liquido al tanque de almacenamiento.
- V1: Válvula que permite el paso del fluido al tanque de almacenamiento.
- S1: Sensor que indica que el nivel de almacenamiento del tanque se encuentra en cero.
- V2: Valvular que permite liberar fluido almacenado en el tanque de almacenamiento y llevarlo al tanque de recepción.

10a. Descripción del proceso.

Las condiciones iniciales del proceso son las siguientes:

La motobomba debe encontrarse apagada.

Las válvulas 1 y 2 deberán encontrarse cerradas (apagadas).

El sensor de nivel deberá encontrarse encendido, es decir que el tanque deberá estar vacío o en su capacidad mínima.

El proceso se llevará de la siguiente forma. Cuando el sensor de nivel inferior se encuentre se encuentre activado es decir el tanque de almacenamiento se encuentre vacío, la válvula de entrada **v1** y la **motobomba** deberán activarse para alimentar el tanque de distribución por un periodo de tiempo de 15 segundos, cuando este periodo se haya cumplido se procederá a interrumpir el fluido proveniente del tanque de distribución cerrando tanto la **motobomba** como la válvula de alimentación **v1**, posteriormente la válvula de salida **v2** se activará para llenar el tanque de recepción hasta que se active el sensor de nivel **s1** del tanque de almacenamiento, cuando esto ocurra se cerrará la válvula de salida **v2** y se repetirá nuevamente el proceso anteriormente mencionado.

En el proceso los actuadores son los siguientes:

- Motobomba.

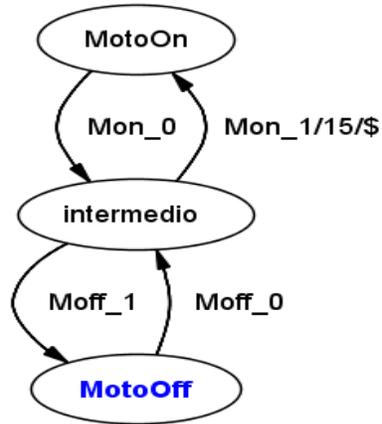


Figura 32: Autómata de la motobomba.

- V1.

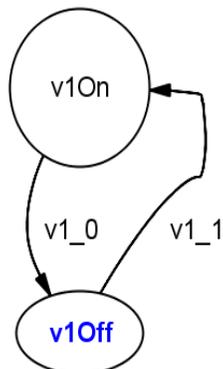


Figura 33: Autómata Válvula V1.

- V2.

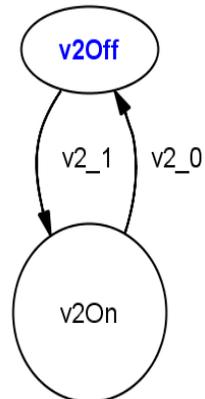


Figura 34: Autómata Válvula V2.

Autómatas de los sensores

En este caso solo hay un sensor en el proceso **S1** su autómata es el siguiente:

S1.

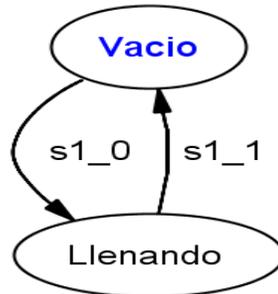


Figura 35: Autómata Sensor S1.

Cumplido este paso ahora se procede a generar el autómata que contiene las restricciones del proceso.

Autómata de las restricciones.

A continuación se mostrará el autómata que contiene las respectivas restricciones que presentan el proceso.

Restricciones.

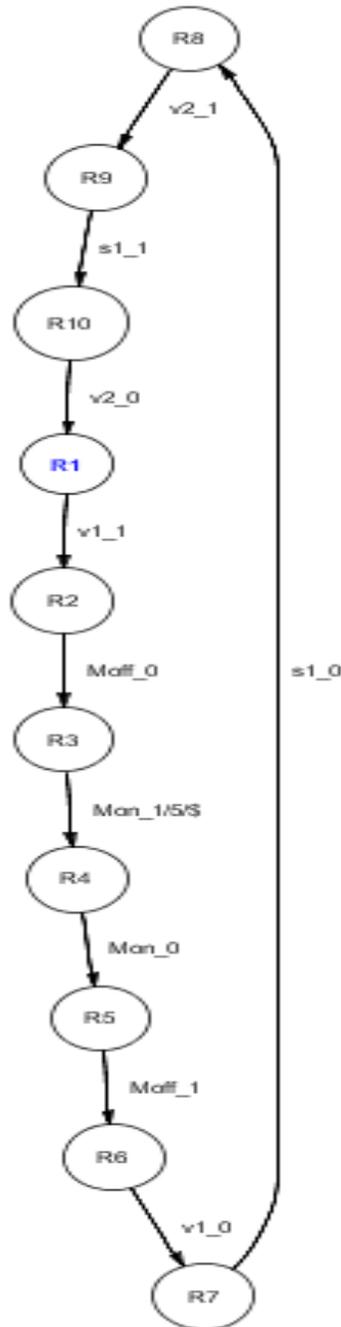


Figura 36: Autómata de restricciones

El siguiente paso a tomar consiste en iniciar la herramienta software FSMLADDER, en ella se ingresaran los autómatas que se han generado con la herramienta DESSUMA.

Cargar archivos en FSMLADDER

Antes que todo se debe seleccionar el espacio de trabajo con la opción de directorio, la cual se encuentra en el menú de archivo, en este caso se tomara el escritorio.

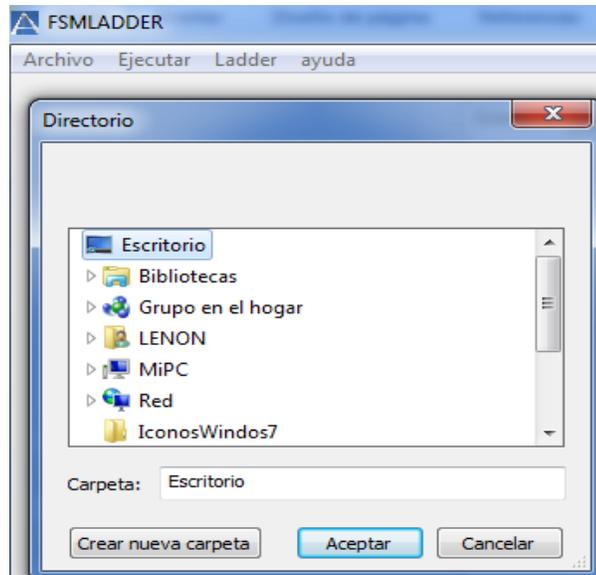


Figura 37: Asignar Directorio de trabajo.

Ahora se cargaran los archivos en FSMLADDER con la opción abrir que se encuentra en el menú archivo y se precederá a generar la composición paralela de ellos, para luego asignar el nombre de “PARALELA” al archivo que se obtuvo en el proceso.

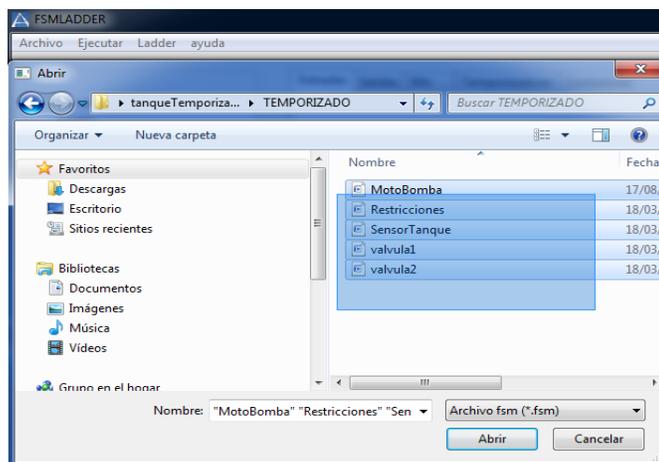


Figura 38: Cargar archivos en FSMLADDER

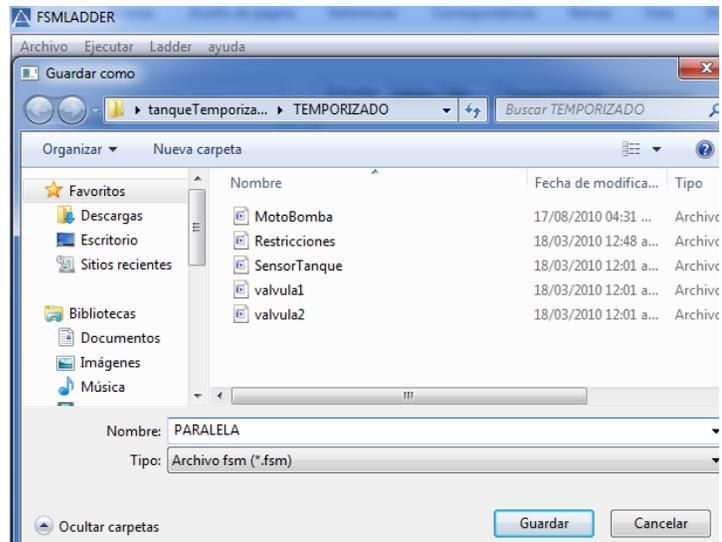


Figura 39: archivo resultado de la composición paralela.

Simulación en Uppal.

El siguiente paso a tomar consiste en hacer la simulación del archivo que se obtuvo en la composición paralela, para ello se ingresa en la opción **Ladder/Composicionparalela/actual** que se encuentra en la barra de herramientas superior.

Nota: En este caso se toma la opción actual debido a que ya se ha generado la composición paralela desde la opción **Archivo/Abrir**.

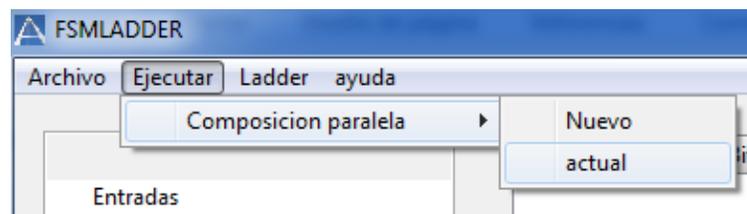


Figura 40: iniciar simulación.

En el momento que se active el icono de actual el software activara la herramienta de simulación UPPAAL y se podrá observar el siguiente diagrama de la composición paralela del proceso.

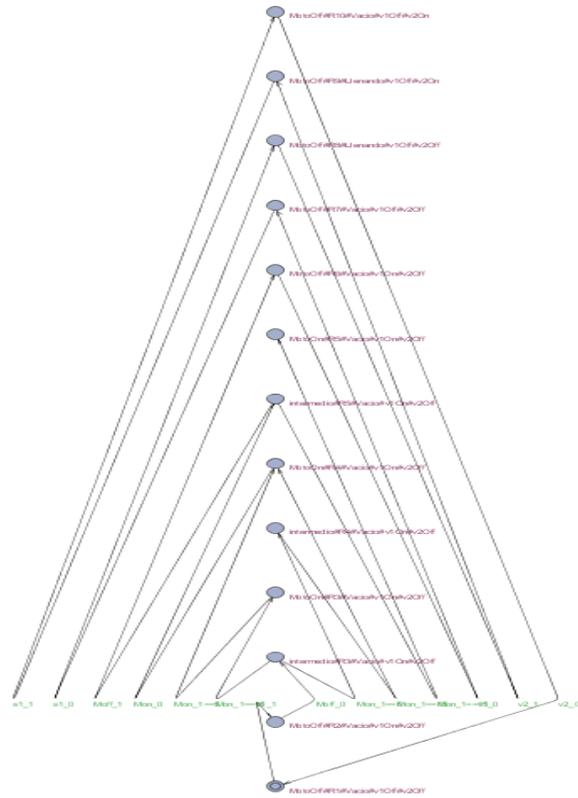


Figura 41: Composición paralela.

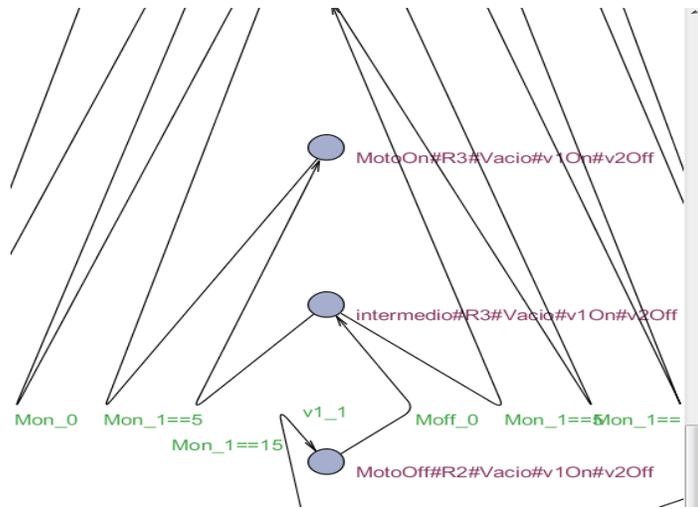


Figura 42: Estados de la composición paralela.

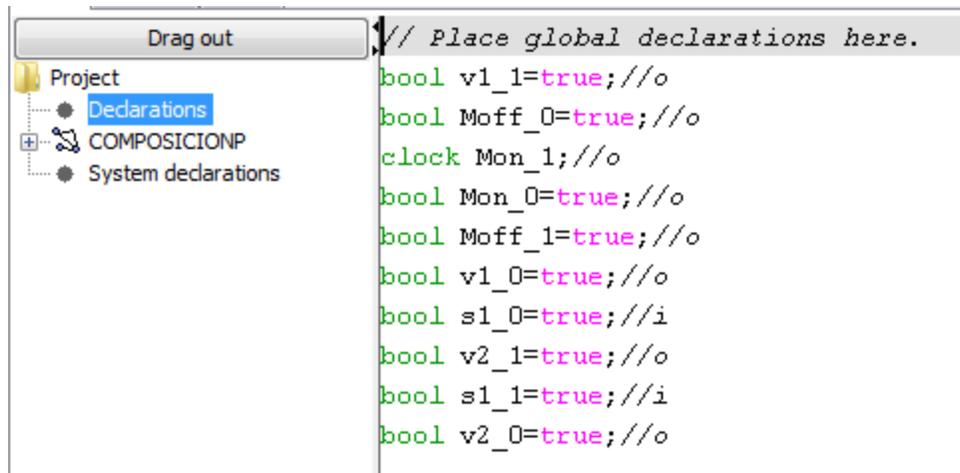


Figura 43: Variables de la composición paralela.

Nota: Si en algún caso el usuario desea cambiar una de las variables del archivo es necesario que al final de cada definición agregue los siguientes caracteres:

- Si la variable es una entrada se asignara los caracteres **//i**
Ejemplo: **bool Sensor_1=true;//i**
- Si la variables esa una salida se asignaran los caracteres **//o**
Ejemplo: **bool Encendido_1=true;//o**
- Si la variable es un bit controlable **//B** o si es un bit observable **//b**.
Ejemplo: **bool start=true;//B**
Ejemplo: **bool cambio=true;//b**

Generar código ladder.

Terminada la etapa de simulación, ahora se procede a definir las direcciones de cada una de las variables entrada y salida del autómata que contiene la composición paralela.

Primero se abre el archivo que contiene la composición paralela (PARALELA.XML) con la opción **ladder** que se encuentra en la barra de herramientas superior del entorno de trabajo.

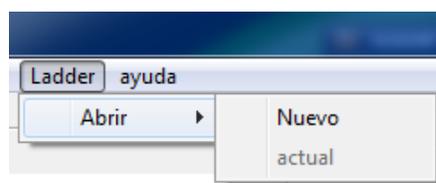


Figura 44: Iniciar opción Ladder

Se selecciona el archivo que se desea asignar las direcciones, compatibles con el formato RSLOGIX 500

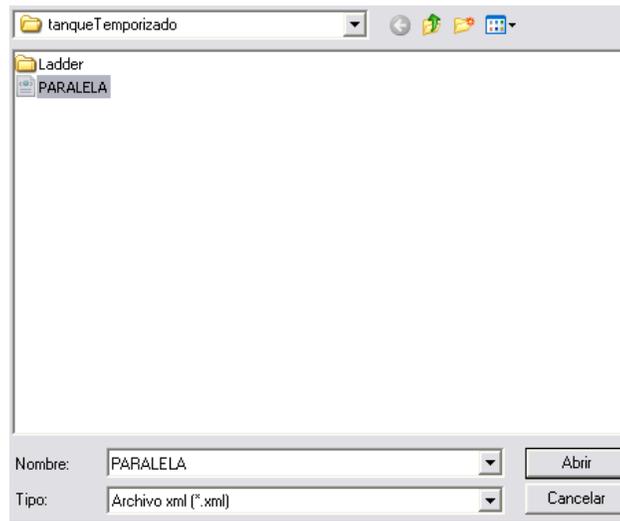


Figura 45: Seleccionar Archivo PARALELA.XML

Ahora se procede asignar las direcciones de entrada y salida del autómata.

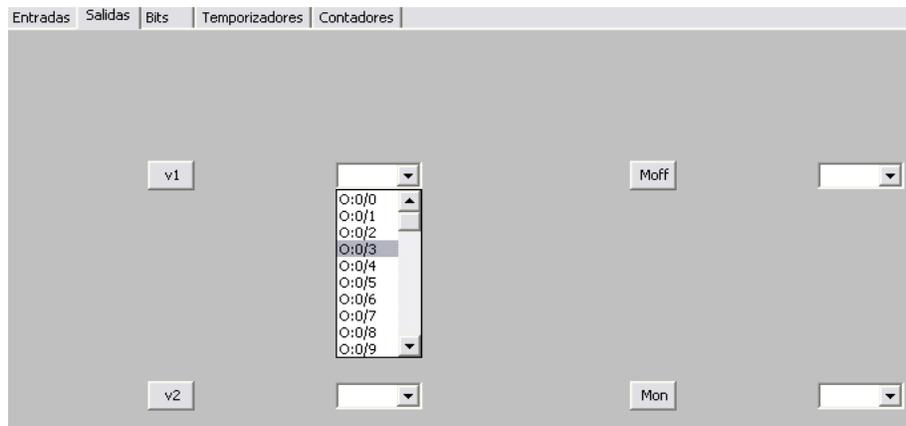


Figura 46: Asignando direcciones a las variables

Terminando este paso solo se tiene que ejecutar la conversión con el botón de **inicio** . Si todo salió de forma satisfactoria, el usuario podrá observar en la carpeta que se asignó como directorio de trabajo 2 nuevos archivos, los cuales tienen extensiones **sy6** y **slc**.

El archivo **PARALELA.sy6** se encuentra almacenado el nombre de las variables del autómata, así como también el nombre de sus estados.

El archivo **PARALELA.slc** se encuentra almacenada la lógica ladder compatible con la herramienta RSLOGIX500

NOTA: Si desea mover el archivo que contiene la lógica ladder, es recomendable mover también el archivo con extensión **sy6**.

El último paso para completar la conversión consiste en ejecutar la herramienta RSLOGIX500 y abrir el archivo que se genero a partir del autómatas (PARALELA.xml), para ello se va a la opción **File/Open** y se busca el archivo que se obtuvo en la conversión (PARALELA.slc) como se muestra en la grafica y se presiona la opción **abrir** en la ventana.

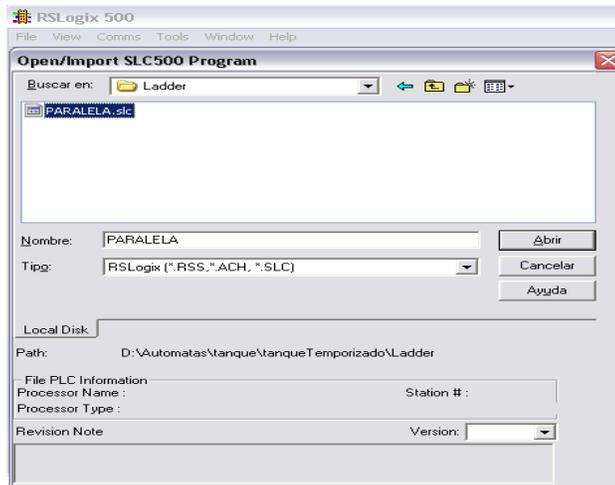


Figura 47: Archivo PARALELA.slc

En el momento que se abre el archivo la ventana de la figura anterior se cierra para dar lugar a una nueva, en donde se dejara las opciones que se encuentra por defecto y se procede a presionar **ok**.

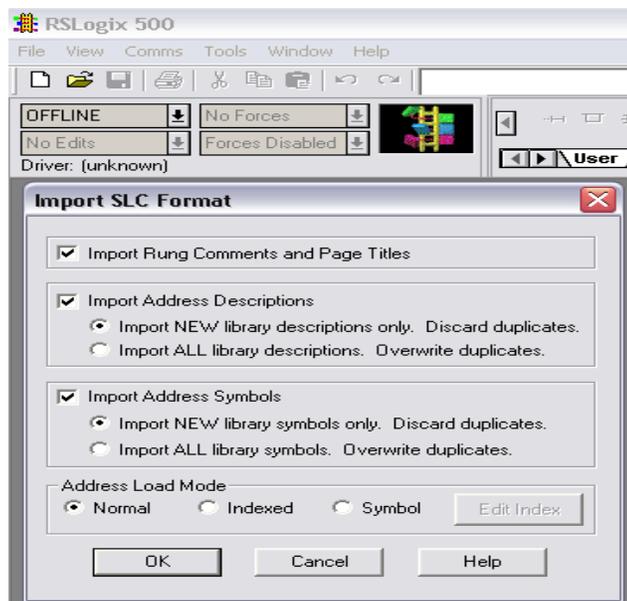


Figura 48: Tipo de formato SLC

NOTA: Si desea conocer afondo el funcionamiento de la ventana de importar, puede utilizar la opción de ayuda que suministra la herramienta RSLOGIX 500.

Para finalizar este tutorial a continuación se mostrara una parte del código resultante del autómata **PARALELA** en la herramienta RSLOGIX 500.

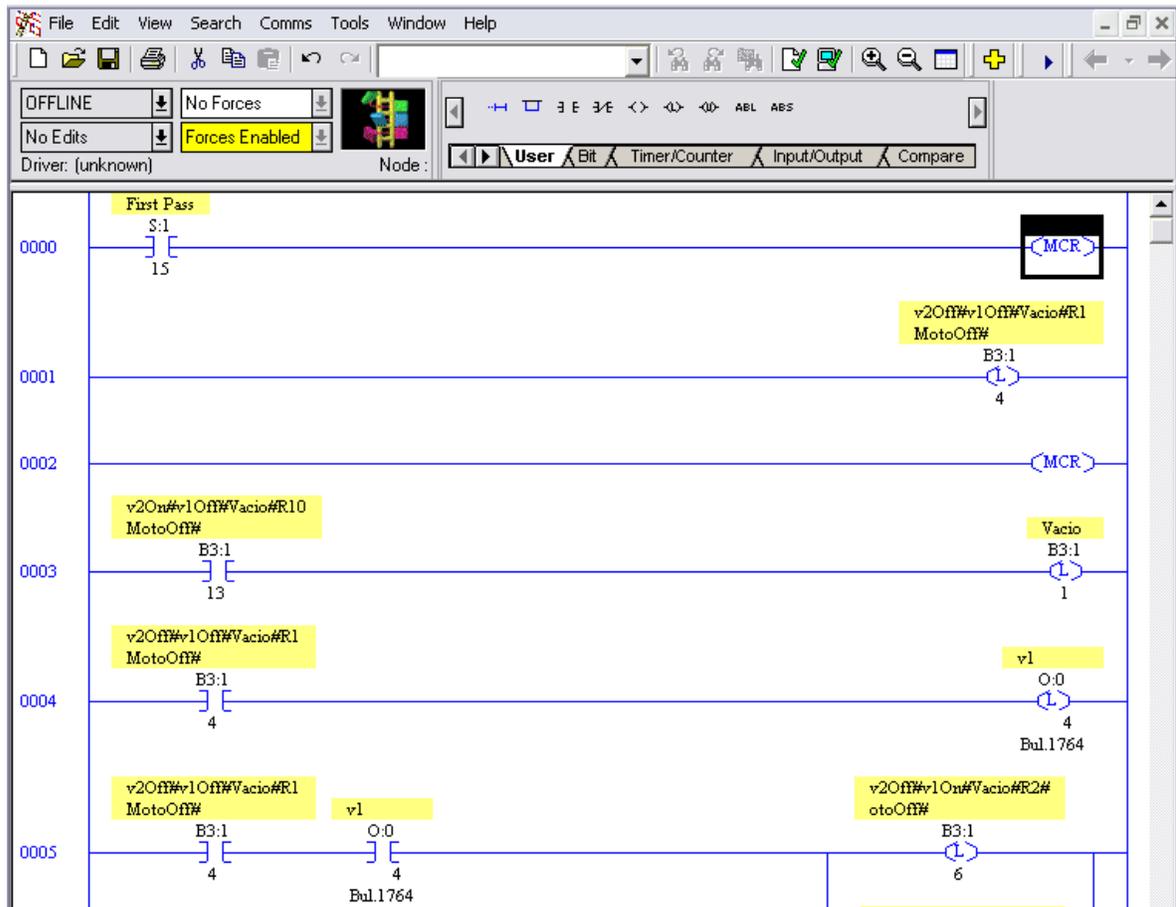


Figura 49: Código Resultante.

ANEXO A. MANUAL DE USUARIO HERRAMIENTA DESUMA

1. Crear un autómata.

De doble clic en el acceso directo de DESUMA.



Figura 50. Acceso directo.

Una vez de clic sobre el icono, aparecerá la ventana que se muestra en la figura 51. A continuación de clic en el icono new mostrado en esta misma grafica, esto le permitirá crear un espacio de trabajo (**Workspace**) para la elaboración del autómata como se observa en la figura 52.



Figura 51. Ventana principal

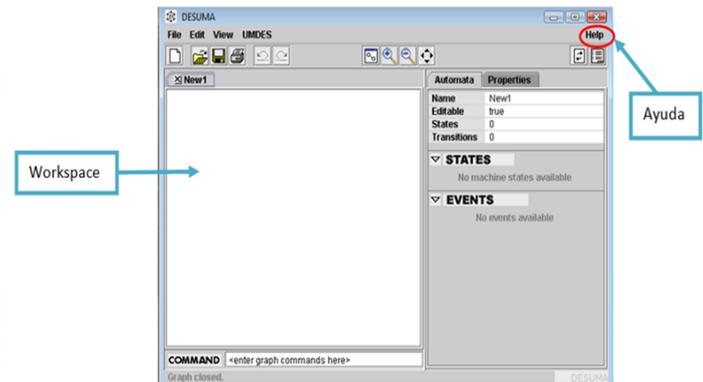


Figura 52. Entorno de trabajo

Para empezar a crear el autómata en el Workspace se deben crear los estados y los eventos, para ello se deben escribir unas sentencias en la casilla COMMAND. Como se detalla a continuación

Crear un estado: ingrese la tecla s o la palabra state dentro de la línea COMMAND seguida del nombre del estado sin dejar espacios y presione <Enter>. s *Nombre_del_estado*.

Crear varios estados: digite dentro de la línea COMMAND la palabra ms seguida por el numero_de_estados a crear y de manera opcional el nombre de los estados.

Así: *ms Numero_de_estados Nombre_de_los_estados* (opcional) y presione <Enter>.

Crear un evento: digite en la línea COMMAND la letra t seguida por el estado inicial del evento, un espacio, el estado destino, nuevamente un espacio y el nombre del evento. Así: *t estado_origen estado_destino Nombre_del_evento* y presione <Enter>.

2. Guardar un autómata creado y eliminar un evento o estado.

Para guardar el autómata se da clic sobre *File*, a continuación se desplegará una ventana de opciones, donde encontrará la opción *Save As...* y de clic sobre esta. Como se indica en la Figura 53 de la izquierda. A continuación, aparecerá una ventana donde se debe especificar el nombre del archivo y la ubicación, dar clic en *guardar* como muestra la figura 53 de la derecha.

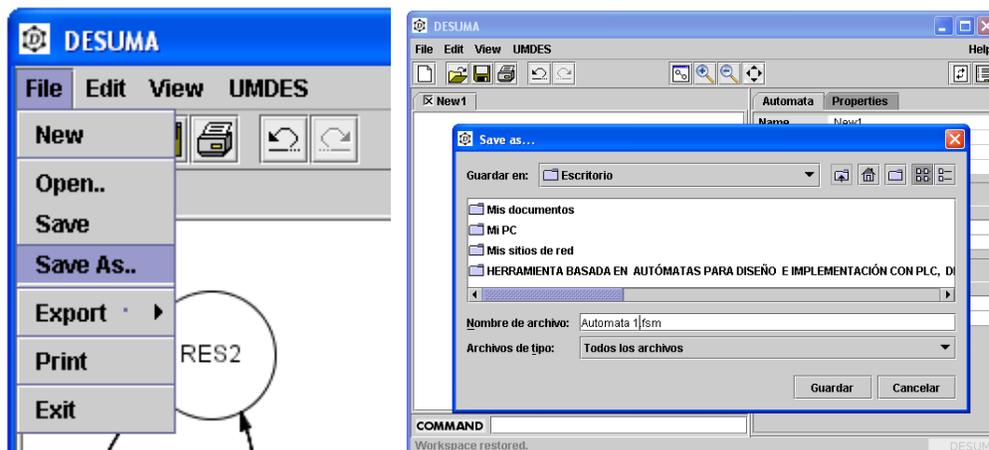


Figura 53. Guardar el Autómata.

3. Eliminar y editar un evento o un estado

3a. Eliminar una transición o evento de un autómata

Existen dos formas de eliminar una transición, la primera consiste en que dar clic en el evento o transición que se desea borrar hasta que el borde cambie de color como se muestra en la Figura 54 de la izquierda., para eliminarlo asegúrese de

que el cursor de mouse este ubicado en el espacio COMMAND, luego se presiona **supr** en el teclado del PC y se remueve la transición especificada. La segunda forma puede ser dar la siguiente instrucción a través de la ventana COMMAND: delete estado_inicial estado_origen como muestra la figura 54 de la derecha y presione **<Enter>**

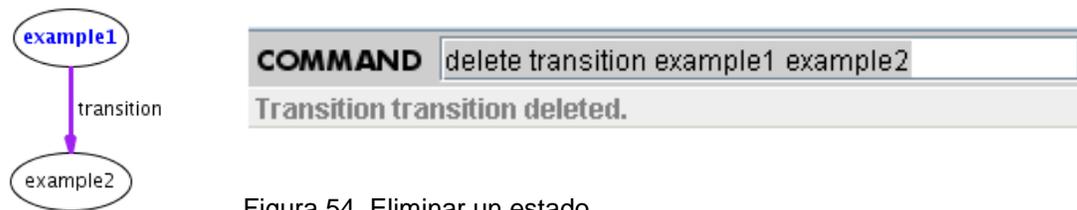


Figura 54. Eliminar un estado

3b. Eliminar un estado de un autómata.

Para eliminar un estado se puede hacer siguiendo cualquiera de los dos pasos mencionados para eliminar un evento, el primero consiste en seleccionar el estado a eliminar como se muestra en la figura 55 de la izquierda, ubicar el cursor en la casilla COMMAND y presionar la tecla suprimir ,la segunda forma consiste en utilizar un comando en la casilla COMMAND con la siguiente instrucción: delete nombre_estado



Figura 55. Eliminar un estado

3c. Editar las propiedades de los estados dentro de un autómata.

No es posible cambiar un estado desde la interface grafica de DESUMA. Por eso se debe crear un nuevo estado con el nombre apropiado y conectarlo, después de esto se debe eliminar el estado que se deseaba modificar.

Para editar las propiedades de los estados existentes ubique la tabla de estados en la tabla *States* en el *Status Panel*, bajo de la tabla *Autómata*. En esta tabla, mostrada en la figura 56, localice el estado y modifique el estado que desee, marcando o desmarcando las casillas de las propiedades. Esto solo se puede editar si el área de trabajo esta desbloqueada.

▼ STATES		
State Name	Marked	Initial
estado1	<input type="checkbox"/>	<input checked="" type="checkbox"/>
estado2	<input type="checkbox"/>	<input type="checkbox"/>
estado3	<input type="checkbox"/>	<input type="checkbox"/>

Figura 56. Editar propiedades del estado.

3d. Editar las propiedades de los eventos dentro de un autómata.

DESUMA nos permite clasificar los eventos dependiendo de su naturaleza, así, un evento podrá ser controlables (seleccionando el campo *control*) u observable (seleccionando el campo *observe*) como se muestra en la figura 57, donde el evento *evento1* se ha calificado como un evento observable, mientras que el evento *evento2* está clasificado como un evento controlable.

▼ EVENTS		
Name	Observe	Control
evento1	<input checked="" type="checkbox"/>	<input type="checkbox"/>
evento2	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Figura 57. Editar atributos de eventos

4. Operación de composición paralela de dos o más autómatas

Para realizar la operación de composición paralela entre dos o más autómatas se deben seguir los siguientes pasos.

Genere los autómatas y guardarlos, en caso de que ya los tenga creados con anticipación, ábralos en DESUMA, recuerde que para hacer la composición paralela se debe disponer de dos o más autómatas.

Ubique el cursor en una de las pestañas que contiene el nombre del autómata como muestra al lado izquierdo de la figura 58 y de clic en la pestaña UMDES, seleccione *Manipulation* y luego seleccione *Paralell Composition*, como se ilustra en la parte superior derecha la figura 58.

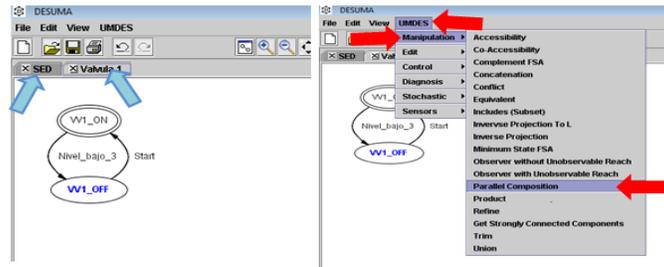


Figura 58. Selección de la operación composición paralela

Al hacer el paso 2 aparecerá una ventana con los autómatas que se encuentran abiertos de los cuales se debe seleccionar el autómata que no se encuentre visible, clic en *Select*, después de esto aparecerá una ventana igual a la anterior, en la cual se debe dar clic en *select* (Ver figura 59.) y aparecerá la composición paralela, que corresponde a un nuevo autómata, guárdelo y úselo para su propósito.

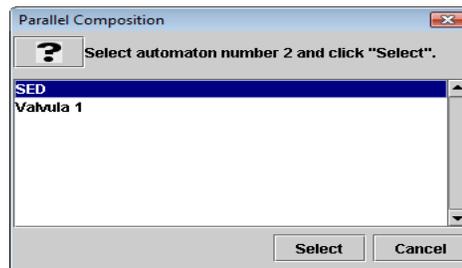


Figura 59. Selección de los autómatas a componer

ANEXO C. METODOLOGIA DE MODELADO

Ver PDF ANEXO C