

**ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE
TELECOMUNICACIÓN
UNIVERSIDAD POLITÉCNICA DE CARTAGENA**



Proyecto Fin de Carrera

Tutorial de SDL sobre el protocolo de comunicaciones Aloha Ranurado



AUTOR: Tamara Cayuela Díaz

DIRECTOR: Francesc Burrull i Mestres

Septiembre / 2014



| | |
|------------------------------|--|
| Alumno | Tamara Cayuela Díaz |
| E-mail del Autor | tamaracayueladiaz@gmail.com |
| Director | Francesc Burrull i Mestres |
| E-mail del Director | Francesc.burrull@upct.es |
| Título del PFC | Tutorial de SDL sobre el protocolo de comunicaciones Aloha Ranurado |
| Resumen | <p>Este proyecto servirá como herramienta de apoyo para la realización en SDL del protocolo de comunicación Aloha Ranurado de N nodos. A lo largo del proyecto, se presentará un tutorial SDL sobre el programa Telelogic Tau, mediante el que se realizará paso a paso la implementación, simulación y validación del sistema Aloha Ranurado.</p> |
| Titulación | Ingeniería Técnica de Telecomunicaciones, Especialidad Telemática |
| Departamento | Tecnologías de la información y las Telecomunicaciones |
| Fecha de Presentación | 09/2014 |

ÍNDICE

| | |
|--|-----------|
| 1. MOTIVACIÓN Y OJETIVOS..... | 10 |
| 2. INTRODUCCIÓN..... | 12 |
| 3. ÁMBITO DE APLICACIÓN..... | 15 |
| 4. EDITOR Y ANALIZADOR..... | 17 |
| 4.1 Iniciando SDL Suite..... | 17 |
| 4.1.1 <i>La ventana Organizador.....</i> | 17 |
| 4.2 Preferencias..... | 18 |
| 4.2.1 <i>Qué va a aprender.....</i> | 18 |
| 4.2.2 <i>¿Para qué sirven las preferencias?.....</i> | 18 |
| 4.2.3 <i>Mostrando y Cambiando Preferencias.....</i> | 19 |
| 4.2.4 <i>Preferencias de Ayuda.....</i> | 20 |
| 4.2.5 <i>Estableciendo Impresora por Defecto.....</i> | 21 |
| 4.2.6 <i>Estableciendo tamaño de Área de Dibujo.....</i> | 21 |
| 4.2.7 <i>Guardar Preferencias.....</i> | 22 |
| 4.3 Creando una estructura en SDL..... | 23 |
| 4.3.1 <i>Qué vamos a aprender.....</i> | 23 |
| 4.3.2 <i>Personalizar capítulos/secciones en Organizer.....</i> | 23 |
| 4.4 Creación de un diagrama del sistema..... | 26 |
| 4.4.1 <i>Adición de un nodo raíz.....</i> | 26 |
| 4.4.2 <i>Crear el diagrama del sistema.....</i> | 27 |
| 4.4.3 <i>Personalizando la ventana Editor SDL.....</i> | 29 |

| | |
|---|----|
| 4.4.4 Colocando Símbolos en bloque de referencia..... | 30 |
| 4.4.5 Mover y Cambiar el tamaño de los símbolos..... | 32 |
| 4.4.6 Dibujando canales entre bloques..... | 32 |
| 4.4.7 Dibujar canales hacia el entorno..... | 34 |
| 4.4.8 Dibujar canales desde el entorno..... | 34 |
| 4.4.9 Dibujar un símbolo de texto..... | 34 |
| 4.4.10 Cambiar tamaño de la ventana de Texto..... | 35 |
| 4.4.11 Otros detalles en el Diagrama del Sistema..... | 35 |
| 4.4.12 Símbolo de referencia del paquete..... | 35 |
| 4.4.13 Encabezado del Kernel..... | 35 |
| 4.4.14 Símbolo adicional de la cabecera..... | 35 |
| 4.4.15 Marco..... | 36 |
| 4.4.16 Numeración de páginas..... | 36 |
| 4.4.17 Guardar el nuevo diagrama creado..... | 36 |
| 4.4.18 Guardar estructura de diagrama..... | 40 |
| 4.4.19 El archivo de sistema..... | 40 |
| 4.4.20 Más acerca de guardar..... | 42 |
| 4.5 Imprimir Diagrama de Sistema..... | 43 |
| 4.5.1 Qué va a aprender..... | 43 |
| 4.5.2 Como imprimir..... | 43 |
| 4.6 Comprobando el diagrama de sistema..... | 45 |
| 4.6.1 Qué va a aprender..... | 45 |
| 4.6.2 Ejecución del analizador..... | 45 |
| 4.6.3 Buscando errores de análisis..... | 47 |
| 4.6.4 Corregir errores de análisis..... | 47 |
| 4.6.5 Como interpretar el mensaje de error..... | 48 |

| | |
|---|-----------|
| 4.7 Creación de un nuevo bloque en el diagrama..... | 50 |
| 4.7.1 <i>Qué va a aprender:</i> | 50 |
| 4.7.2 <i>Creación de un diagrama de bloque desde Organizer.</i> | 50 |
| 4.7.3 <i>Nombre de Procesos y Número de Instancias</i> | 53 |
| 4.7.4 <i>Rutas de señales</i> | 53 |
| 4.7.5 <i>Conectando puntos</i> | 54 |
| 4.7.6 <i>Editar Diagrama de Bloque</i> | 55 |
| 4.7.7 <i>Uso del diccionario de señal para señales individuales</i> ...55 | |
| 4.7.8 <i>Completando el diagrama</i> | 57 |
| 4.7.9 <i>Trabajar con múltiples diagramas</i> | 58 |
| 4.7.10 <i>Trabajar con múltiples ventanas</i> | 59 |
| 4.7.11 <i>Estado actual de Organizer</i> | 61 |
| 4.7.12 <i>Comprobación de la sintaxis del diagrama del bloque</i> ...61 | |
| 4.8 Crear un diagrama de proceso..... | 62 |
| 4.8.1 <i>Editar el proceso Emisor</i> | 62 |
| 4.8.1.1 <i>Qué va a aprender</i> | 65 |
| 4.8.1.2 <i>Creando el diagrama</i> | 65 |
| 4.8.2 <i>Editar Proceso Usuario</i> | 72 |
| 4.8.2.1 <i>Qué va a aprender</i> | 72 |
| 4.8.3 <i>Editar proceso Canal</i> | 76 |
| 4.8.3.1 <i>Qué va a aprender</i> | 76 |
| 4.9 Más acerca de Organizer..... | 82 |
| 4.9.1 <i>Que va a aprender</i> | 82 |
| 4.9.2 <i>Vista de árbol</i> | 83 |
| 4.9.3 <i>Expandir/Contraer</i> | 84 |
| 4.9.4 <i>Reorganizar diagramas</i> | 85 |

| | |
|---|-----|
| 4.9.5 Diagrama de páginas..... | 86 |
| 4.9.6 Impresión del Sistema..... | 87 |
| 4.10 Analizando el Sistema completo..... | 88 |
| 4.10.1 Qué va a aprender..... | 88 |
| 4.10.2 Habilidad de análisis semántico..... | 88 |
| 4.11 Uso del Visor índice..... | 90 |
| 4.11.1 Qué va a aprender..... | 90 |
| 4.11.2 Inicio del visor índice..... | 90 |
| 4.11.3 Encontrar una definición..... | 92 |
| 4.11.4 Encontrar Referencias..... | 93 |
| 4.12 Hasta ahora..... | 94 |
| 4.13 ANEXO: SISTEMA ALOHA..... | 95 |
| 5. SIMULADOR SDL..... | 105 |
| 5.1 Propósito de este tutorial..... | 105 |
| 5.2 Generar e iniciar un simulador..... | 106 |
| 5.2.1 Qué va a aprender..... | 106 |
| 5.2.2 Generando el simulador..... | 107 |
| 5.2.3 Inicio del Simulador..... | 109 |
| 5.3 Ejecutar transición por transición..... | 111 |
| 5.3.1 Qué va a aprender..... | 111 |
| 5.3.2 Ejecución de transiciones de inicio..... | 111 |
| 5.3.3 Utilización de los botones del Simulador..... | 115 |
| 5.4 Como ver el estado interno..... | 117 |
| 5.4.1 Qué va a aprender..... | 117 |
| 5.4.2 Reiniciar el simulador..... | 118 |
| 5.4.3 Viendo procesos y colas de señal..... | 118 |

| | | |
|-------|--|-----|
| 5.4.4 | <i>Visualización de variables e instancias de procesos..</i> | 120 |
| 5.4.5 | <i>Otras opciones de visualización.....</i> | 122 |
| 5.5 | Usando diferentes valores de rastreo..... | 123 |
| 5.5.1 | <i>Qué va a aprender.....</i> | 123 |
| 5.5.2 | <i>Configurando valores de rastreo.....</i> | 123 |
| 5.5.3 | <i>Ejecución símbolo a símbolo.....</i> | 125 |
| 5.5.4 | <i>Ocultando transiciones poco interesantes.....</i> | 126 |
| 5.6 | Observando el comportamiento del sistema..... | 127 |
| 5.6.1 | <i>Qué va a aprender.....</i> | 127 |
| 5.6.2 | <i>Configuración de seguimiento y registro de la señal..</i> | 127 |
| 5.6.3 | <i>Añadir botones para comandos comunes.....</i> | 129 |
| 5.6.4 | <i>Simulación del Sistema y comprobación del fichero signal.log.....</i> | 130 |
| 5.7 | Estableciendo un punto de interrupción | 132 |
| 5.7.1 | <i>Qué va a aprender.....</i> | 13 |
| 5.7.2 | <i>Configuración del sistema.....</i> | 132 |
| 5.7.3 | <i>Establecer un símbolo BreakPoint.....</i> | 133 |
| 5.7.4 | <i>Establecer una transición BreakPoint.....</i> | 134 |
| 5.8 | Cambiando el sistema..... | 139 |
| 5.8.1 | <i>Qué va a aprender.....</i> | 139 |
| 5.8.2 | <i>Algunas preparaciones.....</i> | 140 |
| 5.8.3 | <i>Creación de procesos y cambio en estado del Timer.</i> | 141 |
| 5.9 | Gráficos de Generación de MSC..... | 143 |
| 5.9.1 | <i>Qué va a aprender.....</i> | 143 |
| 5.9.2 | <i>Inicializando MSC.....</i> | 143 |
| 5.9.3 | <i>Seguimiento de la ejecución MSC.....</i> | 145 |
| 5.9.4 | <i>Volver a un diagrama SDL.....</i> | 147 |

| | |
|--|-----|
| 5.9.5 Finalizando traza MSC..... | 148 |
| 5.10 El mapa de cobertura..... | 149 |
| 5.10.1 Qué va a aprender..... | 149 |
| 5.10.2 Puesta en marcha del mapa de cobertura..... | 149 |
| 5.10.3 Usando Coverage Viewer..... | 150 |
| 5.10.4 Examinando detalles de cobertura..... | 152 |
| 5.10.5 Abandonando la interfaz de usuario Simulador..... | 153 |
| 5.11 Hasta ahora..... | 154 |
| 6. VALIDADOR SDL..... | 155 |
| 6.1 Propósito de este tutorial..... | 155 |
| 6.2 Generar e iniciar un validador..... | 156 |
| 6.2.1 Qué va a aprender..... | 156 |
| 6.2.2 Inicio rápido de un validador..... | 156 |
| 6.2.3 Conceptos básicos de un validador..... | 158 |
| 6.3 Navegando en un árbol de comportamiento..... | 160 |
| 6.3.1 Qué vamos a aprender..... | 160 |
| 6.3.2 Configuración de la exploración..... | 161 |
| 6.3.3 Usando el navegador | 162 |
| 6.4 Más posibilidades de rastreo y visualización..... | 167 |
| 6.4.1 Qué vamos a aprender..... | 167 |
| 6.4.2 Uso de los comandos de visualización..... | 167 |
| 6.4.3 Usando MSC..... | 168 |
| 6.4.4 Ir a un estado mediante los comandos de ruta..... | 169 |
| 6.5 Validando un Sistema SDL..... | 170 |
| 6.5.1 Qué va a aprender..... | 170 |
| 6.5.2 Realización de una exploración Bit State..... | 171 |

| | | |
|-------|--|-----|
| 6.5.3 | <i>Verificando Informes</i> | 173 |
| 6.5.4 | <i>Solución del error producido</i> | 175 |
| 6.5.5 | <i>Exploración de un espacio de estado más grande</i> .. | 177 |
| 6.5.6 | <i>Restringir el espacio de estado</i> | 180 |
| 6.5.7 | <i>Comprobación de la cobertura de validación</i> | 181 |
| 6.5.8 | <i>Ir a un estado mediante reglas definidas por el usuario</i> | 182 |
| 6.5.9 | <i>Realizar un paseo aleatorio</i> | 184 |
| 6.6 | Verificación de un MSC | 185 |
| 6.6.1 | <i>Qué va a aprender</i> | 185 |
| 6.6.2 | <i>Verificar un Sistema MSC</i> | 185 |
| 6.6.3 | <i>Saliendo de la interfaz de usuario Validador</i> ... | 187 |
| 6.7 | Hasta el momento | 188 |
| 7. | CÁLCULO DE VARIABLES | 189 |
| 7.1 | Qué va a aprender | 189 |
| 7.2 | Validación completa del sistema | 190 |
| 7.3 | Simulación del Sistema | 196 |
| 7.4 | Cálculo de la curva S-G | 198 |
| 7.5 | Utilización del terminal MS-DOS de Windows | 201 |
| 7.6 | Automatizar el proceso de Simulación | 204 |
| 7.7 | Simulación para todos los valores de PG | 207 |
| 7.8 | Comparativa entre variables | 209 |
| 8. | CONCLUSIONES | 216 |
| 9. | BIBLIOGRAFÍA | 217 |

1. MOTIVACIÓN Y OBJETIVOS

La finalidad de este proyecto es la de proporcionar los conocimientos necesarios en SDL para la correcta utilización de la herramienta Telelogic Tau, y así poder crear paso a paso un protocolo de comunicaciones en lenguaje SDL, en este caso, sobre el protocolo Aloha Ranurado de N nodos.

El principal objetivo para el desarrollo de este proyecto es el de proporcionar una ayuda, motivación y ofrecer una iniciación para la creación de protocolos de comunicaciones en SDL.

Para ello, se proporciona:

- Una herramienta adecuada para la creación de protocolos de comunicaciones en SDL, como es la herramienta Telelogic Tau.
- Los conocimientos básicos para la correcta utilización de dicha herramienta, así como la creación, validación y simulación de un protocolo de comunicaciones.

El proyecto se encuentra dividido en diversos apartados:

- **Apartado 4.** Facilitará los siguientes elementos:
 - Una introducción al lenguaje SDL y al protocolo de comunicaciones Aloha Ranurado.
 - Una primera aproximación a la herramienta Telelogic Tau.
 - Guía paso a paso de la creación del sistema Aloha.

Motivación y Objetivos

- **Apartado 5.** Se llevará a cabo la simulación del sistema. Para ello se hará uso de la interfaz de usuario del simulador, se observarán las distintas opciones que la herramienta ofrece, así como las posibilidades que nos brinda Telelogic Tau para comprobar la correcta ejecución de un sistema mediante su simulación.
- **Apartado 6.** . Mostrará cómo realizar una correcta validación del Sistema, así como las diferentes opciones y posibilidades que la validación ofrece
- **Apartado 7.** Basándonos en el protocolo de comunicaciones creado, (Aloha Ranurado de N nodos), se realizará una validación y simulación completa de dicho sistema. Se obtendrán una serie de variables pertenecientes al sistema, con las que posteriormente se realizarán una serie de comparativas y se comprobará el comportamiento de dicho sistema.

2. INTRODUCCIÓN

El lenguaje SDL es un lenguaje orientado a la descripción de sistemas de comunicaciones; más concretamente al diseño y la especificación de sistemas en tiempo real. El área de aplicación de SDL incluye procesamiento de llamadas, mantenimiento y tratamiento de fallos, gestión de redes, control de sistemas y protocolos de comunicación de datos.

SDL posee dos formas sintácticas para la representación de sistemas:

- SDL/GR: lenguaje gráfico que permite visualizar la estructura y flujo de control de un sistema.
- SDL/PR: lenguaje de programación, para la utilización de herramientas automatizadas.

El comportamiento estándar de un sistema SDL es el siguiente:

En SDL la entidad principal es *System* (Sistema) que está compuesto por *Blocks* (bloques). Los bloques se conectan entre sí y con el entorno mediante *Channels* (canales). Estos canales sirven como medio de transmisión de señales entre los bloques y el entorno. El bloque está formado por varios procesos, un proceso interactúa con varios procesos y con el entorno por medio de las señales.

Aloha Ranurado:

El ejemplo usado en este tutorial es el medio de transmisión broadcast Aloha Ranurado de N nodos. Para ello se hará uso del programa Telelogic Tau mediante lenguaje SDL, utilizado para el diseño y la especificación de sistemas en tiempo real. Desde un punto de vista académico, el estudio de redes aloha es muy interesante, ya que a pesar de su sencillez, permite extraer conclusiones respecto a protocolos más complejos.

Introducción

Comportamiento de Aloha Ranurado:

En una red de comunicación el medio de transmisión utilizado se puede dividir en dos categorías: los constituidos por enlaces punto a punto y los que utilizan un canal broadcast al que se conectan físicamente todos los nodos de la red. La característica más destacada de un canal broadcast consiste en la asignación eficiente del derecho a transmitir a los nodos que desean hacerlo.

El control de acceso en redes broadcast se lleva a cabo mediante técnicas denominadas de asignación dinámica del canal, en las que el nodo transmite en función de cuales desean hacerlo; o mediante la técnica de acceso aleatorio o por contienda, en la que no podemos planificar el instante en el que un nodo podrá transmitir y en un momento dado, los nodos pueden competir por el control del medio y se deben resolver esas contiendas. En los mecanismos de control de acceso por contienda no existe estación central que decida quién transmite a continuación, puesto que el control estará repartido de forma uniforme entre todos los nodos de la red.

El comportamiento de Aloha Ranurado se produce mediante control de acceso al medio por contienda.

El protocolo Aloha tiene un funcionamiento simple: cuando un nodo quiere transmitir una trama, la envía sobre el canal sin comprobar si hay otro nodo transmitiendo. En el caso de que no haya ningún nodo transmitiendo, la trama llega correctamente. El problema surge cuando dos nodos intentan enviar a la vez o cuando el canal está ocupado por una transmisión y un nodo quiere enviar una trama. A este inconveniente se le denomina “colisión”, en el cual es imposible realizar una correcta recepción de las tramas implicadas. En caso de colisión, el emisor espera un tiempo aleatorio y reintenta enviar la trama que ha sufrido colisión. Tenemos una ventana de colisión de $[t_0 - t, t + t_0]$, entendiéndose como t_0 el instante cuando se empieza a realizar una transmisión y t como el tiempo necesario para realizar la transmisión.

El Aloha Ranurado es una derivación de Aloha, en la que se implementa unos slots (ranuras temporales), consiguiendo disminuir el número de colisiones.

En el nuevo algoritmo, si un nodo quiere transmitir una trama, espera al inicio de la siguiente ranura temporal. Esta forma disminuye a la mitad la ventana de colisión y mejora el comportamiento del sistema. De esta manera, las colisiones solo se producirán

Introducción

cuando dos nodos quieran enviar en la misma ranura temporal. La ventana de colisión queda definida $[t_0 - t, t_0]$, disminuyendo la probabilidad de error. En caso de colisión, se utiliza el algoritmo de Back-Off, como método de retransmisión. Según este algoritmo, si se produce una colisión, espera $[0, K-1]$ ranuras temporales y reenvía. Si hay colisión, repite el proceso utilizando un valor mayor de K . Así sucesivamente hasta que se consigue transmitir con éxito la trama.

Para llevar a cabo el diseño SDL se va a hacer uso de la herramienta Telelogic Tau Suite SDL. Los productos de la Suite SDL se utilizan para el diseño y la especificación de sistemas en tiempo real.

SDL Suite soporta el lenguaje SDL y la especificación recomendada Z.100.

Vamos a demostrar, mediante el uso de un sistema SDL como ejemplo, la edición básica de un sistema y la funcionalidad de análisis que tiene disponible.

Se realizarán ejercicios que le ayuden a familiarizarse con el Editor SDL así como con SDL Analyzer (analizador de SDL).

Con el fin de aprender a utilizar estas herramientas, continúe leyendo este tutorial; debe realizar los ejercicios en su sistema informático tal y como se describen a continuación.

3. ÁMBITO DE APLICACIÓN

El tutorial que se va a realizar en este proyecto se puede desarrollar en dos plataformas distintas, tanto **UNIX** como **Windows**.

Este tutorial asume que la Suite SDL ha sido instalada correctamente.

› En UNIX, el directorio de instalación de Telelogic Tau se señala por la variable de entorno \$telelogic.

› En Windows, el directorio de instalación de Telelogic Tau será C:\Telelogic\SDL_TTCN_Suite4.3.

Se debe crear un nuevo directorio para el propósito de este tutorial.

En **UNIX**, siga las siguientes instrucciones:

1. Cree un nuevo subdirectorio, en su directorio personal (por ejemplo).
 - **mkdir ~/aloha**

En **Windows**, siga las siguientes instrucciones:

1. Cree un directorio local (por ejemplo).
 - **C:\Telelogic\SDL_TTCN_Suite4.3\aloha**

Iniciando Suite SDL

En **UNIX**, para iniciar el entorno Suite SDL:

1. Cambie el directorio a su directorio aloha

cd ~/aloha

2. Ahora, escriba

Sdt

Ámbito de aplicación

En **Windows**, para iniciar la Suite SDL de una manera adecuada para este tutorial, debe crear y utilizar un icono de acceso directo.

1. Localice el ejecutable SDL **sdt.exe** en
C:\Telelogic\SDL_TTCN_Suite4.3\bin\wini386
2. Cree un icono de acceso directo a este archivo en el Escritorio.
3. Desde el menú emergente del nuevo icono, seleccione *Propiedades*. En la ventana de diálogo, seleccione la ficha *Acceso directo* en la parte superior.
4. En el campo *iniciar en*, escriba la ruta de acceso del nuevo directorio
C:\Telelogic\SDL_TTCN_Suite4.3\aloha
5. Haga clic en aceptar.
6. Haga doble clic en el icono de acceso directo.

La realización del tutorial se realizará, de ahora en adelante, sobre la plataforma Windows.

4. EDITOR Y ANALIZADOR

4.1. Iniciando SDL Suite

4.1.1 La ventana Organizador

Al iniciar SDL Suite, se nos muestra la ventana *Organizer* (Organizador). *Organizer* es la herramienta principal desde la que tener acceso a todas las herramientas en el entorno de Telelogic Tau.

Organizer también proporciona la ventana *Welcome* (Bienvenido), donde se puede leer el acuerdo de licencia de Telelogic Tau. Esta ventana se encuentra siempre en la parte superior de la ventana *Organizer* y desaparece tan pronto como se realice alguna acción en la ventana *Organizer* (también pulsando el botón *Continuar*).

Ya está listo para empezar a trabajar.

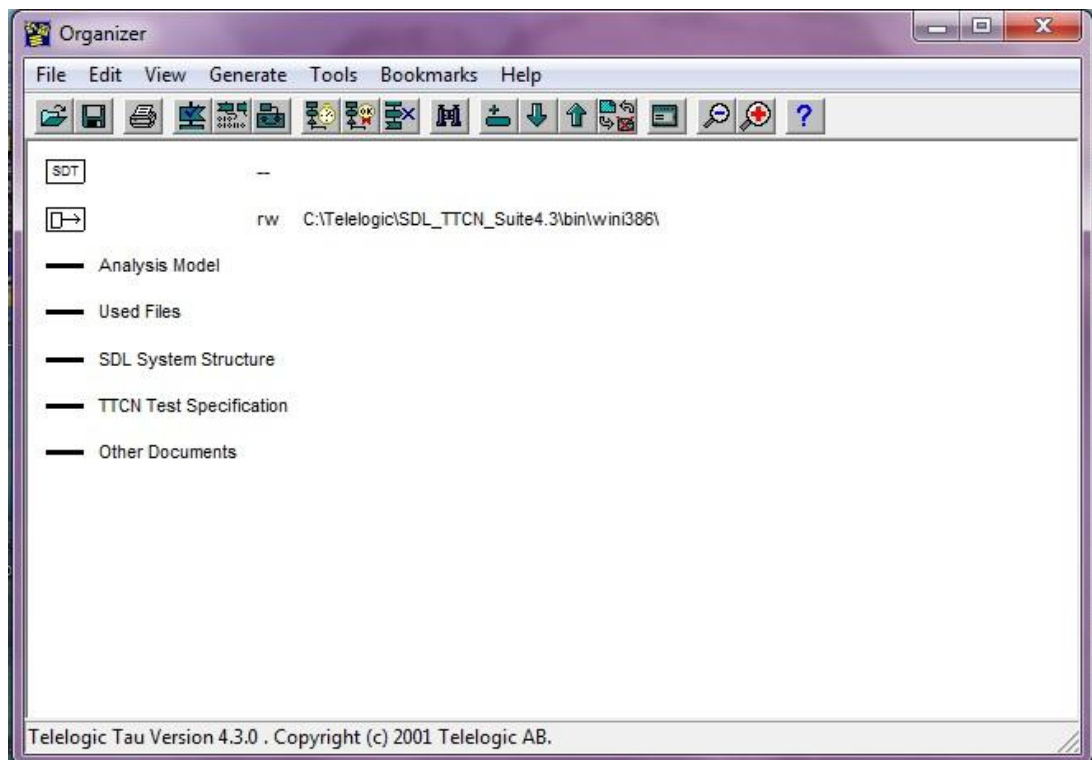


Figura 1: Ventana *Organizer*

4.2 Preferencias

4.2.1 Qué va a aprender

- A establecer e insertar preferencias.
- Los fundamentos básicos de la interfaz gráfica de SDL Suite, ayudados por ratón y teclado. Aprenderá a:
 - Usar listas gráficas
 - Usar menús desplegados
 - Usar menús emergentes
 - Usar botones de acceso rápido
 - Usar la barra de estado
 - Usar opciones de menú
 - Usar campos de texto
 - Usar barras deslizantes

4.2.2 ¿Para qué sirven las preferencias?

Antes de crear su primer diagrama en SDL, se deben establecer algunas preferencias para que coincidan con las especificaciones de tu ordenador. Estas preferencias afectan al comportamiento por defecto de las herramientas de Telelogic Tau y deben ser ajustadas con los valores correspondientes para que Telelogic Tau trabaje apropiadamente, (la mayoría de las opciones en Telelogic Tau deben establecerse como preferencias). Al instalar Telelogic Tau, los ajustes del fabricante son utilizados como los ajustes de preferencia.

Al menos se deben comprobar los siguientes ajustes.

- Preferencias de ayuda
- Preferencias de la impresora
- Tamaño del área de dibujo
- El modo de plataforma

4.2.3 Mostrando y Cambiando Preferencias

Para ver y cambiar las preferencias:

1. Desde el menú *Tools* (Herramientas) en la ventana *Organizer*, seleccione el comando *Preference Manager* (Administrador de Preferencias). Se mostrará la ventana Administrador de Preferencias.

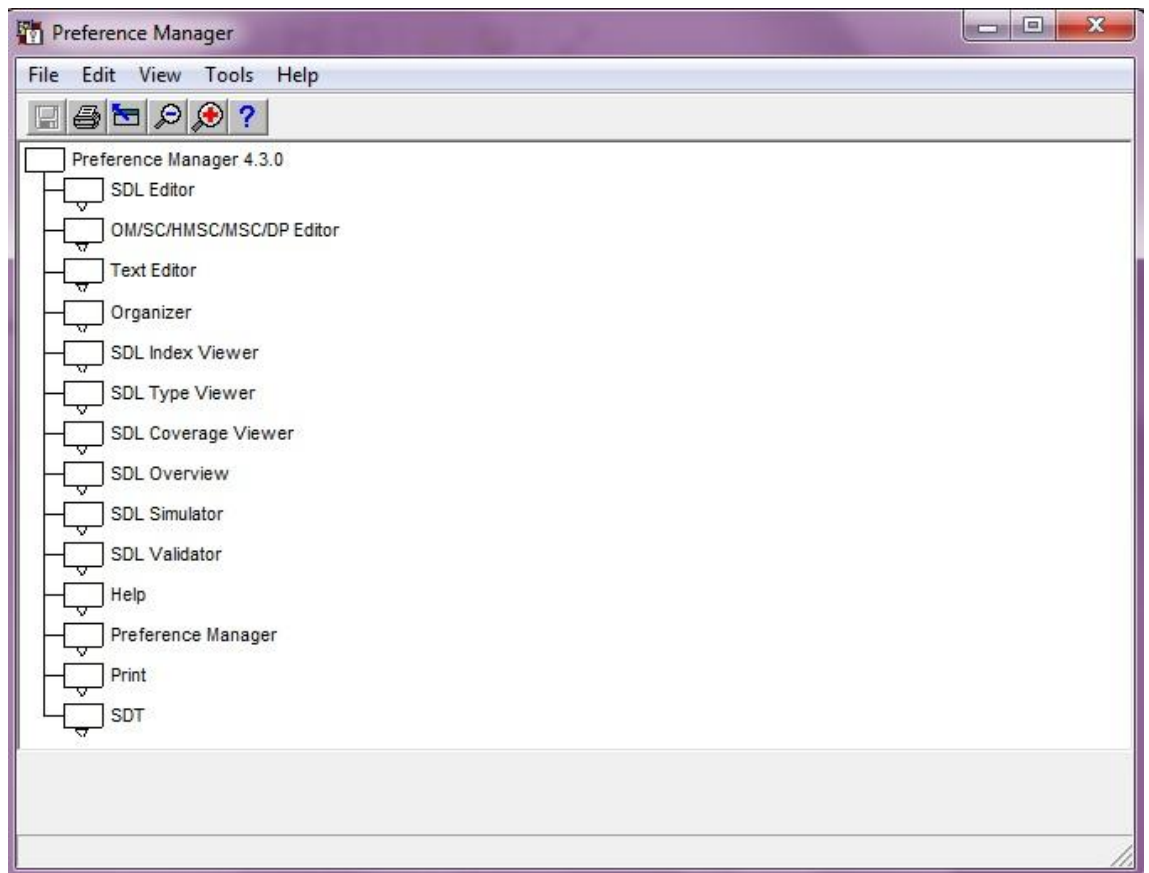


Figura 2: Ventana Administrador de Preferencias

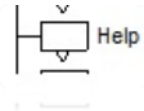
La siguiente tarea consistirá en comprobar, y si es preciso, modificar algunas preferencias.

4.2.4 Preferencias de Ayuda

Telelogic Tau facilita una ayuda online que puede ser utilizada en cualquier momento si se necesita ayuda con una ventana, un comando etc.

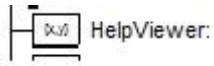
La ayuda online se encuentra en formato HTML y se puede acceder a ella desde el navegador Netscape o Internet Explorer. Se puede cambiar el navegador en los ajustes.

Para establecer las preferencias de ayuda:



1. Localice el icono titulado *Help* y haz doble clic en él.
 - También puedes pinchar con el botón derecho del ratón y seleccionar *Expand*.

Esto nos mostrará una lista con las preferencias de ayuda.



2. Localice la preferencia *HelpViewer*.

A la derecha del icono de preferencia se puede ver el valor actual, el valor guardado en la actualidad y un texto explicativo.



3. Cambie la preferencia seleccionando el icono *HelpViewer* y después seleccionando un nuevo navegador en el menú desplegado en la zona inferior de la ventana Help. El icono se volverá gris, lo que indica que la preferencia ha sido cambiada y necesita ser guardada.

4. Dependiendo de la elección realizada se debe comprobar que Telelogic Tau utilice el visor correcto cuando se abre la opción ayuda.

5. Contraiga el icono *Help* haciendo doble clic sobre él.

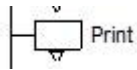
Ahora ha aprendido a trabajar con las listas gráficas en Telelogic Tau. Listas de gráficos se utilizan ampliamente en toda las herramientas; pueden mantener tantos niveles como sea necesario (el Administrador de preferencias utiliza tres niveles de sangría, como se ve en la pantalla). Algunas herramientas también soportan un árbol vertical, como una alternativa a

una lista gráfica. La funcionalidad es idéntica, sólo difiere en su presentación. Se verá uno de ellos más adelante en este tutorial.

4.2.5 Estableciendo Impresora por Defecto

En este tutorial, se aprenderá a imprimir diagramas. Antes de comenzar la impresión desde Telelogic Tau debe comprobar, si fuera necesario, las preferencias establecidas con respecto a su impresora.

Para establecer las preferencias de impresión:



Print

1. Localice el icono *Print* (Imprimir). Desplieguelo.



PrinterCommand:

2. Localice la preferencia *PrinterCommand*. Ajústelo a un valor adecuado. Puede especificar cualquier comando adecuado a su sistema operativo.

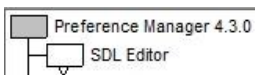


PaperFormat:

3. Localice la preferencia *PaperFormat*. Telelogic Tau soporta un número predefinido de tamaños de papel. Para definir un tamaño específico debe modificar los valores en *UserDefinedWidth* y *UserDefinedHeight*, que establecerá tanto la altura como la anchura en milímetros.
4. Si es necesario ajuste los valores de *MarginUpper*, *MarginLower*, *MarginLeft* y *MarginRight*, que establecen el espacio en milímetros de los márgenes así como de encabezado y pie de página.
5. Ajuste si es necesario la preferencia *LandScape* que especifica la orientación en horizontal o vertical.

4.2.6 Estableciendo tamaño de Área de Dibujo

Al editar diagramas en SDL, a las páginas se les asigna un tamaño predefinido. Puede especificar el tamaño por defecto para que coincida con los ajustes de tamaño y márgenes previamente establecidos.



Preference Manager 4.3.0

SDL Editor

1. Localice el icono *SDL Editor*, desplieguelo y compruebe las preferencias y *PageHeight*.
2. Cámbielas si fuera necesario.

4.2.7 Guardar Preferencias

Ahora debe guardar los ajustes realizados en las preferencias para posteriores usos de Telelogic Tau.



1. Seleccione el botón *Save* (Guardar) que se encuentra dentro del menú *File*. También puede seleccionar el botón de acceso directo situado en la parte de arriba de la ventana.
2. Se recibirá un aviso de que los ajustes realizados a las preferencias no entrarán en uso hasta que el programa haya sido restablecido. Pulsa OK. Tus preferencias han sido guardadas.
3. Cierre la ventana de Preferencias seleccionando la opción *Exit* en el menú *File*.

Los cambios en las preferencias han concluido. Por supuesto puede acceder siempre que lo desee a *Preference Manager* y ajustar otros valores.

4.3 Creando una estructura en SDL

Ya está preparado para crear su primer diagrama en SDL.

4.3.1 Qué va a aprender

- A personalizar los capítulos/secciones en la ventana *Organizer*
- A crear una estructura en SDL.
- A agregar un nodo raíz del sistema
- Crear un diagrama del sistema
- Agregar una página
- Editar un diagrama del sistema
- Guardar un diagrama en el archivo
- Guardar una estructura de diagrama en un archivo de sistema
- Trabajar con cuadros de diálogo (modal y no modal)
- Trabajar con estructuras de árbol

4.3.2 Personalizar capítulos/secciones en *Organizer*

Cuando iniciamos SDL Suite, la ventana *Organizer* muestra dos iconos que simbolizan el archivo del sistema y el directorio de origen para sus diagramas. El archivo de sistema se explicará más adelante. El directorio de origen es donde los componentes de SDL Suite buscarán los diagramas existentes y donde serán guardados los nuevos. (Este directorio puede ser cambiado por el que se desee).

Por defecto, *Organizer* muestra 5 áreas u opciones en su ventana.

- *Analysis Model* (Modelo de Análisis)
- *Used Files* (Archivos Usados)
- *SDL Sistem Structure* (Estructura del sistema SDL)
- *TTCN Test Specification* (Especificación de pruebas TTCN)
- *Other Documents* (Otros Documentos)

Estas áreas son conocidas como capítulos o secciones. Pueden ser utilizados para contener ciertos diagramas y documentos, su uso depende del usuario.

Editor y Analizador

Puesto que se va a comenzar con un diseño simple, se recomienda borrar las opciones *Analysis Model*, *Used Files* y *TTCN Test Specification*.

Para borrar estos capítulos:

1. Seleccione el capítulo *Analysis Model*.
2. Seleccione la opción *Remove* (Eliminar) del menú *Edit* (Editar).

Se abrirá una ventana de diálogo confirmando lo que se desea eliminar.

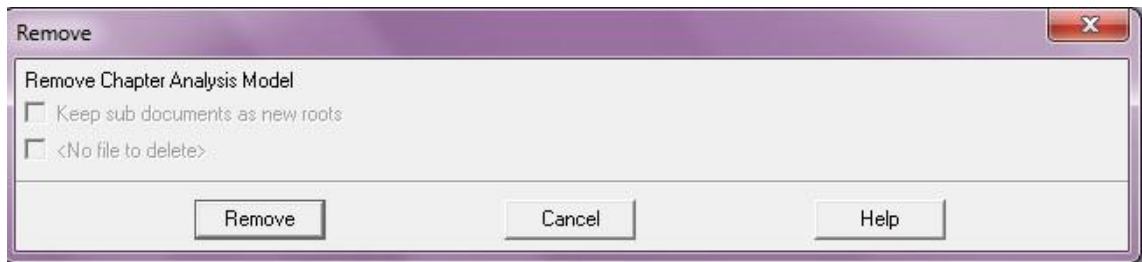


Figura 3: Confirme el capítulo/sección a eliminar

3. Repita los pasos anteriores con *Used Files* y *TTCN Test Specification*.

También puede renombrar las dos secciones restantes.

1. Seleccione *SDL System Structure*.
2. Seleccione la opción *Edit* dentro del menú *Edit* (Editar)
3. En la ventana de diálogo, debe asegurarse de que *Edit Chapter Symbol* esté marcado y pulsar el botón.

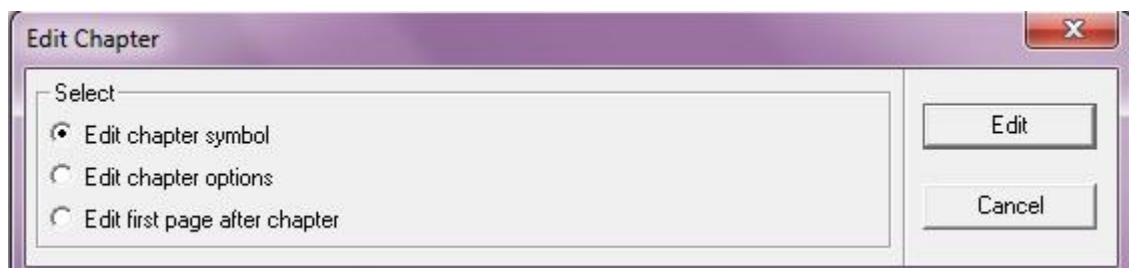


Figura 4: Editando Nombre de la sección

4. En la nueva ventana desplegada, cambie el nombre del documento; por ejemplo, a **Mi Primer Sistema SDL**. No debe cambiarse la opción *Chapter* en el apartado *Organizer*.

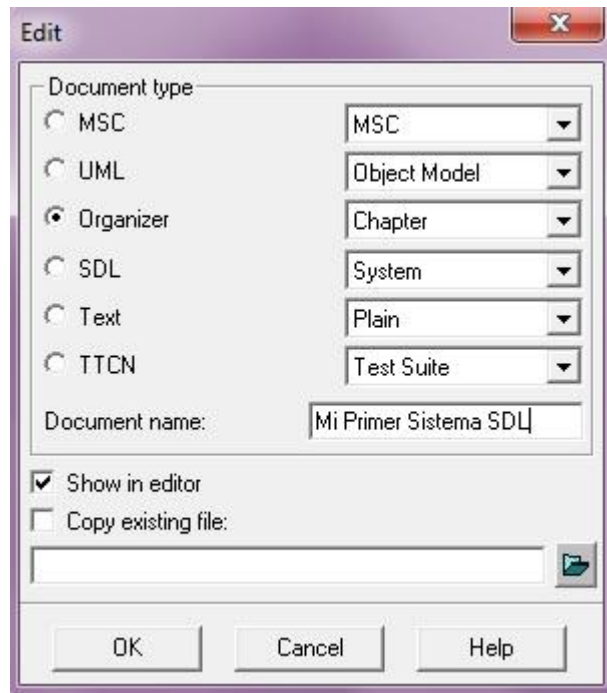


Figura 5: Nombrando la sección

5. Desmarque la opción *Show in editor* (Mostrar en el editor).
 6. Finalice pulsando el botón OK.
- Si se desea también se puede cambiar el nombre de la sección *Other Documents*, que puede ser usada para contener diagramas que no forman parte de nuestro sistema SDL pero que deben estar localizados.

4.4 Creación de un diagrama del sistema

2.4.1 Adición de un nodo raíz

Ahora va a crear un sistema SDL.

1. Asegúrese de que **Mi Primer Sistema SDL** se encuentre seleccionado.
2. Seleccione *Add New* (Añadir Nuevo) desde el menú *Edit*. Se abrirá una ventana donde se deberá especificar el nombre y el tipo de sistema a añadir.

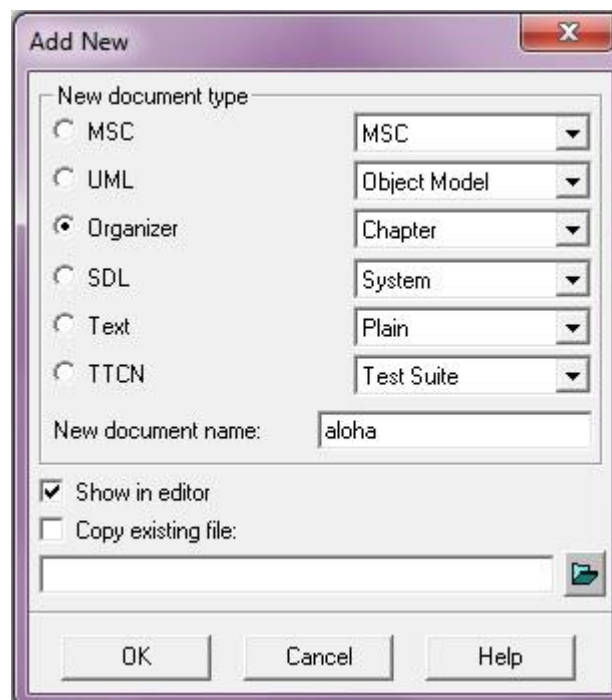


Figura 6: Añadiendo un nuevo diagrama

3. Especifique *New Document Type* como *SDL* y en *SDL* el tipo de diagrama debe ser *System*.
4. Especifique el nuevo nombre del documento como **aloha**.
5. Asegúrese de que se encuentre desmarcada la opción *Show Editor*.
6. Pulse el botón **OK**.

La ventana desaparece y en la ventana Organizer se encontrará el nodo raíz, sistema aloha. El nodo se encuentra marcado como desconectado, pues no tiene relación con ningún archivo físico.



Figura 7: El nuevo nodo raíz.

4.4.2 Crear el diagrama del sistema

Hasta ahora se ha creado una estructura del diagrama, que consiste en una referencia a un diagrama del sistema SDL. Sin embargo el diagrama mencionado todavía no existe.

La siguiente tarea consistirá en crear el diagrama del sistema.

1. Seleccione el icono del diagrama en SDL al que se ha llamado **aloha**.
2. Desde el menú *Edit* seleccione la opción *Edit*. También se puede acceder a él mediante el botón derecho del ratón y pulsando en *Edit*; o pulsando dos veces sobre el icono con el botón izquierdo del ratón.
3. Aparece una ventana llamada *Edit* que sugiere la creación de un nuevo diagrama y que sea mostrado en el editor.

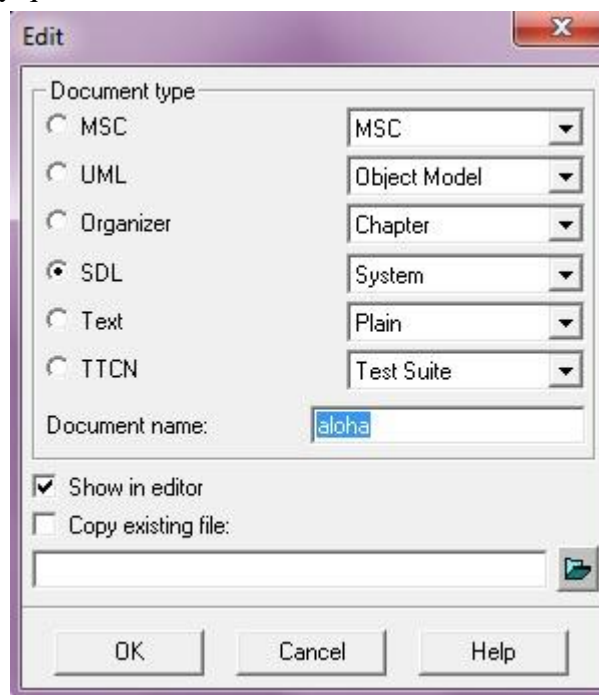


Figura 8: Creando nuevo diagrama

Editor y Analizador

SDL Suite responde mostrando la ventana SDL Editor, mostrando en la esquina superior izquierda de la página 1 el diagrama system aloha.

El Editor de SDL es la herramienta que se utiliza al editar el contenido de los diagramas.

El Editor de SDL también se utiliza para la construcción de la estructura diagrama que se muestra en la ventana *Organizer*.

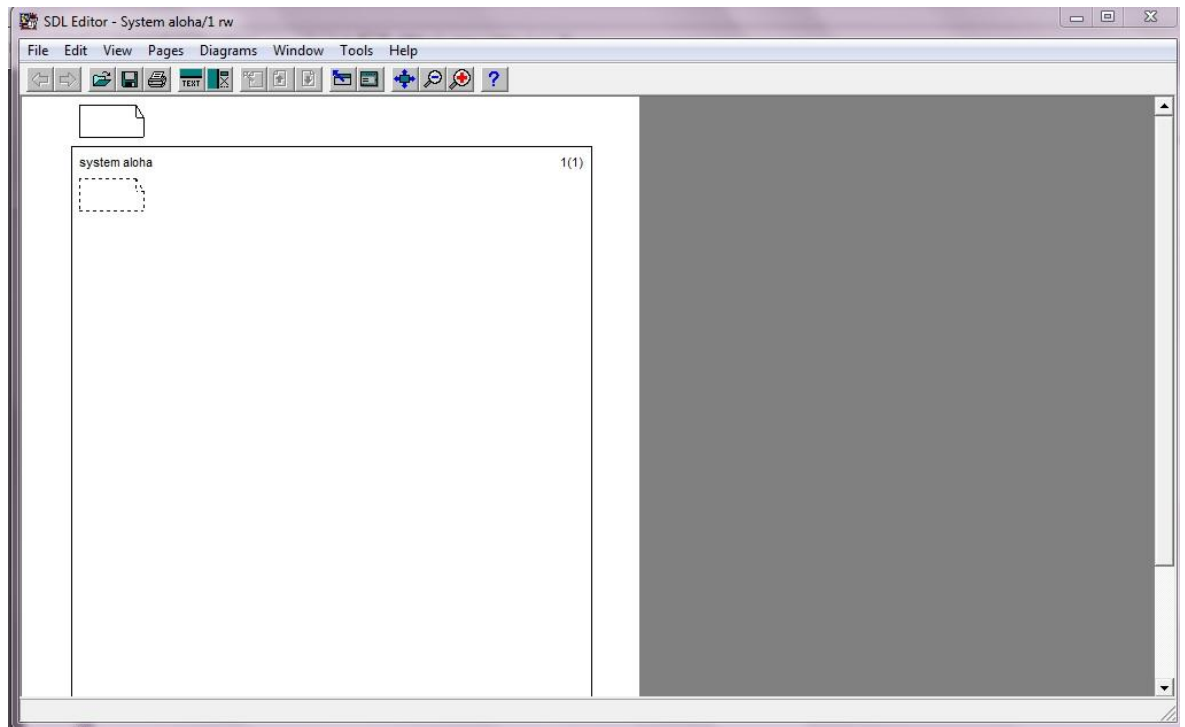


Figura 9: Ventana del editor SDL

La siguiente tarea a realizar consistirá en completar los contenidos del diagrama. La figura 10 muestra la apariencia del diagrama una vez completo. Como se puede ver el diagrama consiste en un símbolo bloque llamado aloha dentro del cual se definirá el funcionamiento del sistema.

Un bloque es un contenedor para una o más definiciones de proceso de un sistema.

También contiene símbolos de texto donde se encuentran las declaraciones de señales, así como explicaciones de lo que se está realizando.

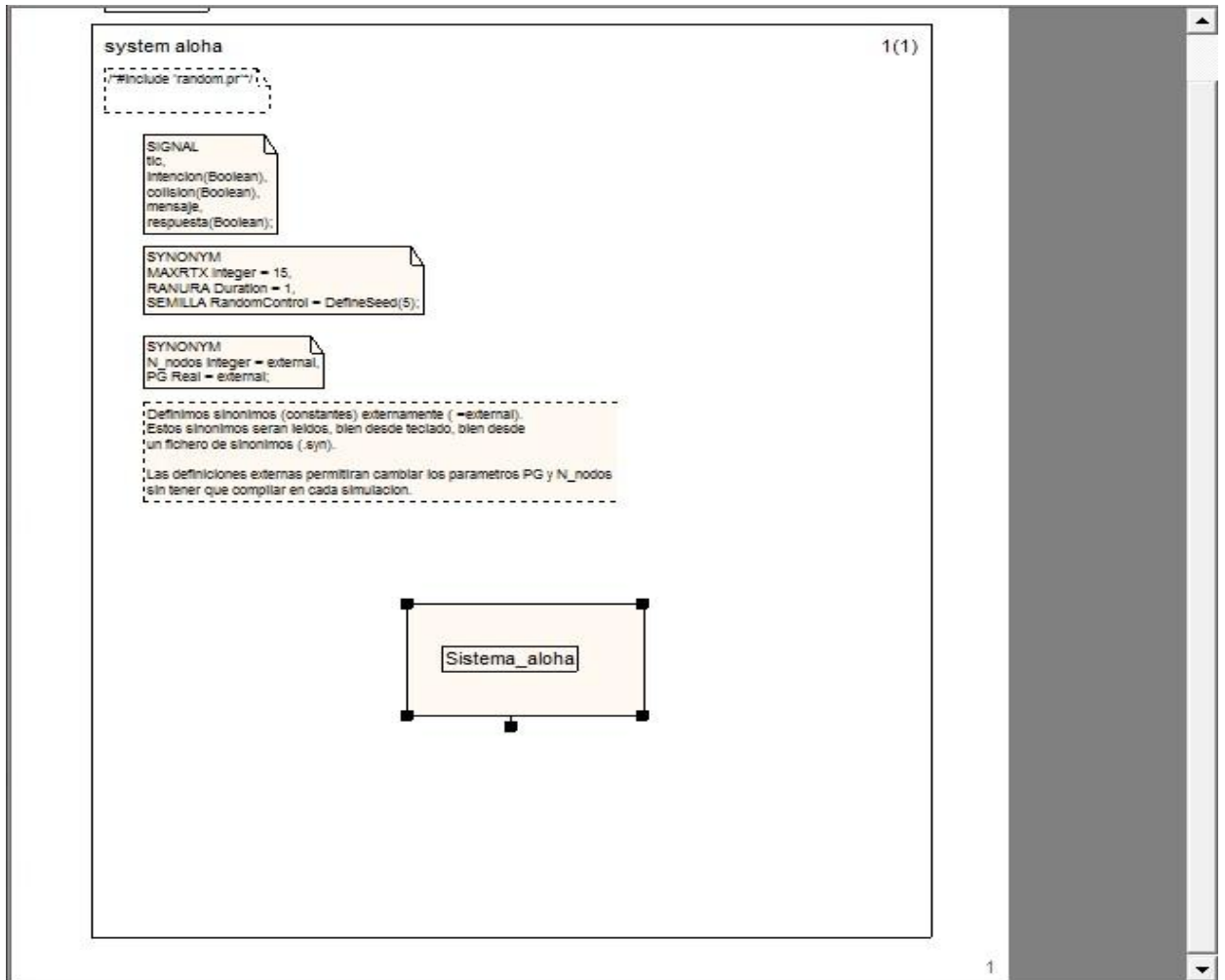


Figura 10: El diagrama del sistema

En las siguientes páginas se detallará como añadir símbolos y textos al diagrama.

4.4.3 Personalizando la ventana Editor SDL

Antes de comenzar a editar, tal vez quiera redimensionar la ventana editor. Así como ocultar o esconder varias ventanas usando el comando *Windows Option* desde el menú *View* (Vista).

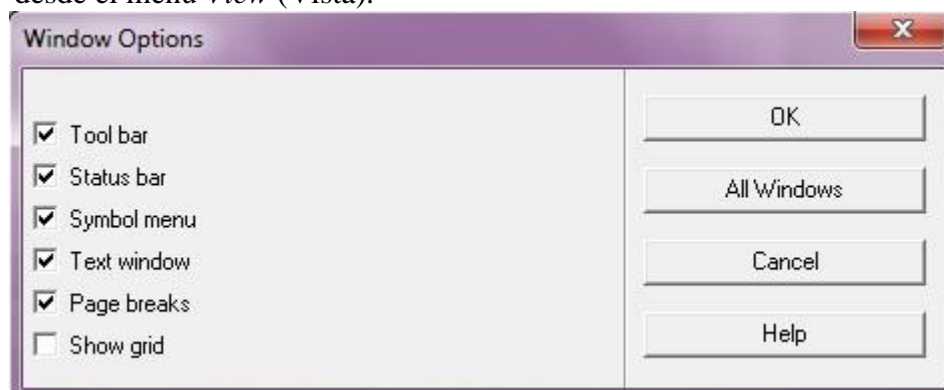


Figura 11: Ventana *Windows Options*



- Puede ocultar y mostrar la barra de herramientas de edición, la barra de estado etc.
- También puede ocultar y mostrar la ventana de texto y el menú de símbolos haciendo clic en los botones de acceso rápido proporcionados. Pues necesitará esas ventanas en breve.

4.4.4 Colocando Símbolos en bloque de referencia

1. Comience a editar el diagrama insertando un símbolo de bloque.

Para colocar el símbolo:

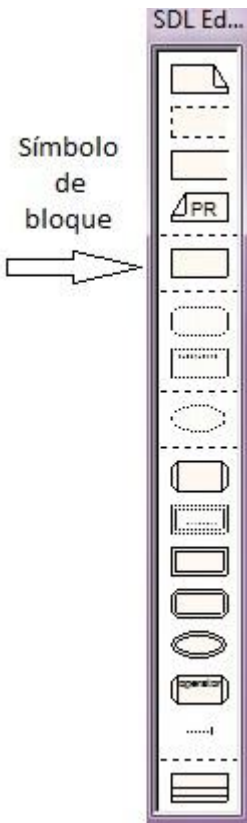
- Seleccione el símbolo de bloque en el menú de símbolos.
- Mueva el ratón en el área de dibujo. El símbolo “flota” y sigue al ratón. Pinche para colocarlo en el lugar deseado. No se permite ninguna superposición entre símbolos. (En caso de que se superpongan, suena una alerta que le indica que debe repetir la operación).

Nota: Interrumpir y deshacer

- Para abortar la inserción de un símbolo después de haberlo movido con el ratón hacia el área de dibujo, presionar <Esc>.
- Si se ha ejecutado un comando u operación que no debería haber tenido lugar, debe seleccionar de inmediato la acción de *Undo* (Deshacer) desde el menú *Edit*.

Una vez situado el símbolo, debe escribir el nombre del bloque, en este caso Sistema_aloha.

- Puede escribir texto directamente en la posición del cursor. La posición del cursor se puede ajustar haciendo clic en el texto del símbolo. Sin embargo, se puede seleccionar (resaltar) el texto directamente en el símbolo.
- Antes de que haya comenzado la edición de texto, el cursor de texto no está parpadeando. Pulsando <Delete> en esta etapa se elimina el símbolo seleccionado. Una vez que la edición de texto se ha iniciado, el cursor de texto parpadea y presionando <Delete> sólo elimina un carácter.



Editor y Analizador

- Es posible que vea un subrayado rojo que aparece en el texto si el nombre introducido tiene una sintaxis incorrecta de acuerdo con SDL. Es la manera que tiene el editor de SDL de marcar los errores textuales.
- Al editar texto, también puede hacer uso de *text window*, que le permite seleccionar el texto arrastrando.
- No importa donde se introduce el texto, el texto se muestra siempre tanto en el símbolo como en la ventana de texto.
- También puede notar que la estructura del diagrama en la ventana *Organizer* se actualiza automáticamente y añade el nuevo bloque creado.

4.4.5 Mover y Cambiar el tamaño de los símbolos

Para mover un bloque:

- Seleccione y arrastre el bloque con el ratón hasta la posición deseada.

Una vez colocado en la posición buscada, se puede cambiar el tamaño del bloque de la siguiente manera:

- Apunte a una de las esquinas del símbolo y arrastre. Debe ser bastante cerca de la esquina. Si este método no funciona, primero seleccione el símbolo con un clic y luego repita el procedimiento mientras que apunta a uno de los cuadrados de selección.

4.4.6 Dibujando canales entre bloques

La creación de canales es muy importante a la hora de crear un diagrama. Nuestro sistema aloha solo contiene un bloque, por eso aprenderá a realizarlos mediante un ejemplo sencillo que se muestra a continuación.

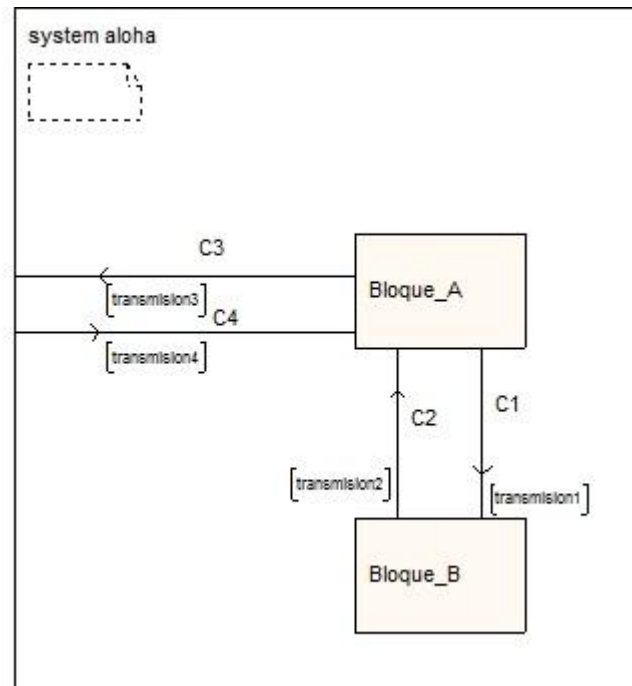


Figura 12: Ejemplo canales entre bloques

1. Selecciona el Bloque_A. Aparecerá un “asa” o “mango”.

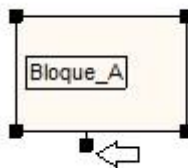


Figura 13: Mango del símbolo bloque

2. Arrastre el “mango”.
3. Tan pronto como empiece a arrastrar con el ratón aparece una línea.
4. Mueva el ratón hasta que apunte al símbolo Bloque_A. Haga clic con el ratón; el canal está conectado en ambos extremos.

Editor y Analizador

- Puede mover los extremos del canal individualmente arrastrando cada uno de ellos. Seleccione el canal en primer lugar si el punto final es difícil de seleccionar con el ratón.

Se dará cuenta de que existe una pequeña selección en la parte central de cada línea. Esto puede ser usado para crear "Puntos de ruptura" en la línea, dividiendo así la línea en diferentes segmentos de línea.

El Editor de SDL crea dos atributos de texto asociados al canal. Estos campos de texto se muestran como rectángulos al seleccionarlos, y en ellos se introduce tanto el número del canal como las señales a transmitir. Al principio, cuando los atributos de texto están vacíos, aparece un subrayado en rojo.

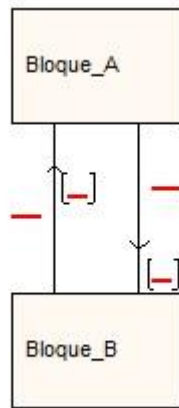


Figura 14: Atributos de texto vacíos.

Para rellenar el nombre del canal C1 y C2:

Escriba directamente, inmediatamente después de que el canal se ha dibujado. Para rellenar la señal en el campo de texto de la lista de señales:

1. Haga clic en el campo de texto delimitado con dos corchetes '[']. (Haga clic en el espacio entre los corchetes.)
2. Escriba el nombre de la señal. Tenga en cuenta que los corchetes se ajustan automáticamente al tamaño del texto.
3. Puede mover los atributos de texto a nuevos lugares, si se desea, simplemente arrastrándolos con el ratón.

4.4.7 Dibujar canales hacia el entorno

Para dibujar un canal del bloque al entorno (C3):

1. Selecciona el bloque.
2. Comience arrastrando desde el asa hasta el marco del área de dibujo.

Escriba el nombre y las señales.

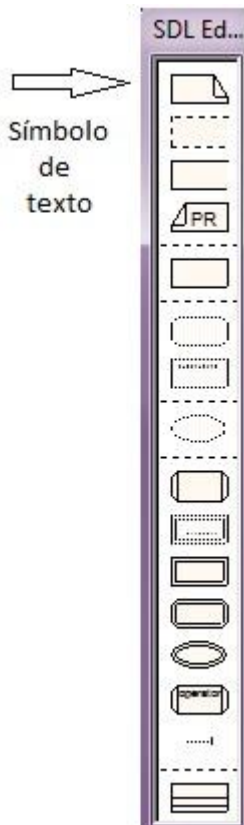
- › Tenga en cuenta que, a medida que escribe la lista de señales, se subrayará en rojo siempre que no sean sintácticamente incorrectas. Además si se escribe más de una señal, deben separarse mediante comas.
- › El Editor de SDL le permite dejar un texto que contiene errores de sintaxis.

Sin embargo, no es posible construir un sistema de SDL que contenga errores de sintaxis.

4.4.8 Dibujar canales desde el entorno

Para dibujar un canal del entorno al bloque (C4):

1. Comience por dibujar el canal del bloque al entorno, como ha hecho anteriormente
2. Seleccione el canal.
3. Seleccione *Redirect* desde el menú *Edit*. Complete el nombre del canal y las señales como se ha explicado anteriormente.



4.4.9 Dibujar un símbolo de texto

El diagrama también contiene un símbolo de texto con las declaraciones de señal requeridos.

1. Elija el símbolo de texto en el menú de símbolos (el símbolo de arriba), insértelo en el área de dibujo y continúe con el contenido. La comprobación de sintaxis incorporada es aún más evidente en este caso.

2. Cuando se cambian los contenidos del texto, el editor automáticamente cambia el tamaño del símbolo. Se puede cambiar su tamaño arrastrando la esquina inferior derecha, o alternar entre minimizar y maximizar haciendo doble clic en el símbolo.

› Los cuadros de selección en las otras tres esquinas del símbolo de texto son de color gris. Esto significa que no se puede variar el tamaño del símbolo arrastrando desde cualquiera de esas esquinas.

4.4.10 Cambiar tamaño de la ventana de Texto

Si la ventana de texto es muy pequeña para todo el texto que se debe introducir, se puede redimensionar. Se realiza de la misma forma que con cualquier otra ventana Windows.

4.4.11 Otros detalles en el Diagrama del Sistema

Aparte de los símbolos SDL, el diagrama también contiene:

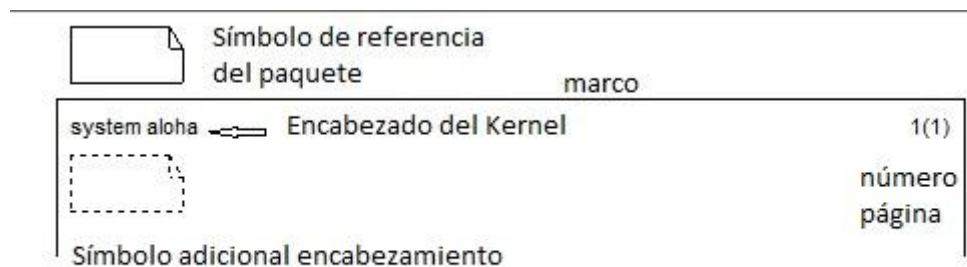


Figura 15: Otros símbolos

4.4.12 Símbolo de referencia del paquete

La referencia paquete se utiliza para referirse a paquetes SDL incluidos. Este ejemplo no incluye ningún paquete.

4.4.13 Encabezado del Kernel

El encabezado del kernel se asigna automáticamente por el editor para reflejar el tipo y el nombre del esquema que se está editando. Es editable, pero usted no va a alterar su contenido en este tutorial.

4.4.14 Símbolo adicional de la cabecera

En el Editor de SDL, se ve como un símbolo de texto discontinuo. El símbolo es editable y se puede cambiar el tamaño de la misma manera que para cambiar el tamaño de los símbolos de texto, pero no se puede mover. Su uso en el

Editor SDL es, entre otros, para definir la herencia y para especificar los parámetros formales.

4.4.15 Marco

El marco rodea los objetos contenidos en el diagrama .Es posible que desee cambiar el tamaño del marco para crear un diagrama más compacta, simplemente arrastre cualquier esquina para hacer esto.

4.4.16 Numeración de páginas

La numeración de las páginas se actualiza automáticamente, y refleja el nombre de la página y el número total de páginas. No es editable.

4.4.17 Guardar el nuevo diagrama creado

Va a aprender los comandos para almacenar diagramas en los ficheros.

1. Ahora debe tener dos ventanas en la pantalla, la ventana Organizer y la ventana de SDL Editor.



- Para encontrar la ventana de Organizer, puede seleccionar en cualquier momento el comando *Show Organizer* desde el menú *Tools* del Editor de SDL, o haga clic en el botón de acceso rápido *Show Organizer*.
2. Antes de guardar cualquier cosa, abra la opción *View Options* desde el menú *View* en la ventana *Organizer*.

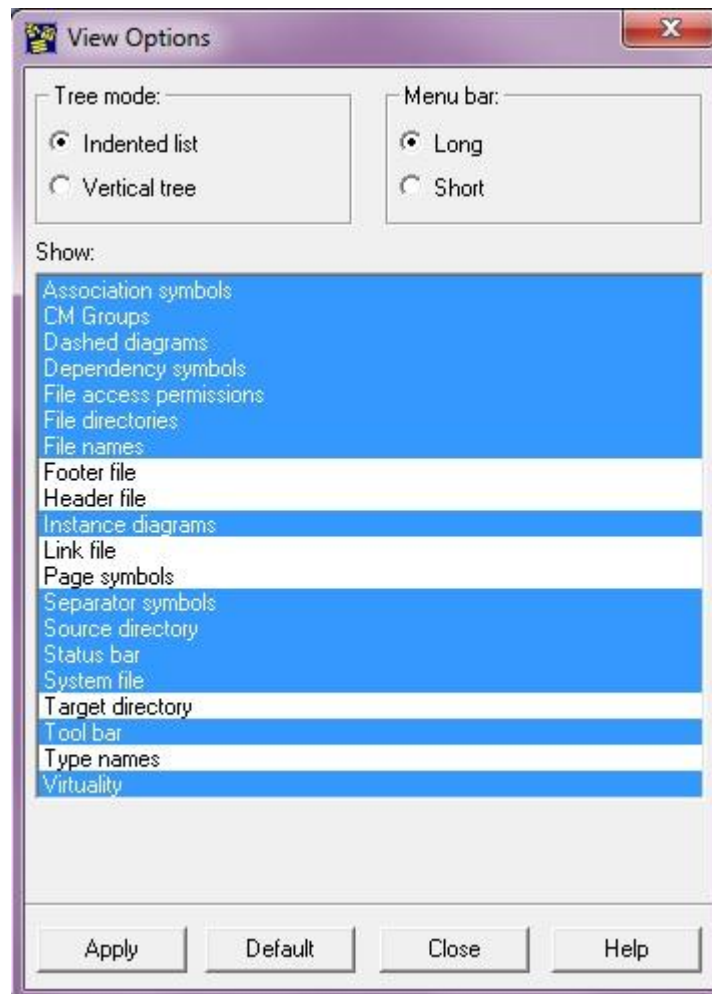


Figura 16: *View Options* en la ventana *Organizer*

- 3 Asegúrese de que las opciones se encuentran marcadas como en la figura 16 y seleccione Apply (Aplicar).

Esto hace, que entre otras cosas, los nombres del archivo y el directorio sean visibles en la ventana Organizer.

- La lista que se nos muestra en la venta View Options es una muestra de selección múltiple. Al hacer clic en un elemento de la lista (en Windows se debe mantener pulsada la tecla <Ctrl>), su estado seleccionado se activa sin afectar a ningún otro elemento. Esto hace que sea posible seleccionar cualquier número de elementos de la lista. Si por error, hace clic en un elemento de la lista sin usar la tecla <Ctrl>, puede pulsar Default para volver a la configuración predeterminada.

Editor y Analizador

- Para cerrar el cuadro de diálogo View Options, haga clic en el botón Close(Cerrar).Este tipo de ventana permanecerá abierta hasta que se decida que ya no es necesario y se cierre.
4. Observe el resultado obtenido en la ventana Organizer. El icono de diagrama del sistema es dibujado con un patrón de color gris, que muestra que el diagrama se ha modificado y no se ha guardado. El nombre del esquema (es decir **aloha**) se muestra en negrita, para indicar que el esquema está abierto en un editor. El texto a la derecha del icono muestra [unconnected] que es una convención adoptada en Telelogic Tau para mostrar que un diagrama no tiene ninguna unión a un archivo actual.

Hay otro icono de diagrama, que muestra [unconnected]. Este representa la referencia al diagrama de bloque que ha añadido al editar el diagrama del sistema.



Figura 17: Diagrama modificado, aloha.

5. Ahora, vuelva al editor SDL y guarde el diagrama SDL seleccionado la opción *Save* desde el menú *File*.
- Para localizar la ventana *Editor* cuando se encuentre en la ventana *Organizer*, haga doble clic sobre el icono del diagrama del sistema, en este caso **aloha** y la ventana *Editor* volverá a aparecer.

Editor y Analizador

6. Se muestra un cuadro de diálogo de selección de archivos. Este es un cuadro de diálogo genérico que Telelogic Tau abre cada vez que se le solicita que especifique un archivo (para abrir, guardar, etc.) El título del cuadro de diálogo muestra la naturaleza de la operación, salvo en este caso.

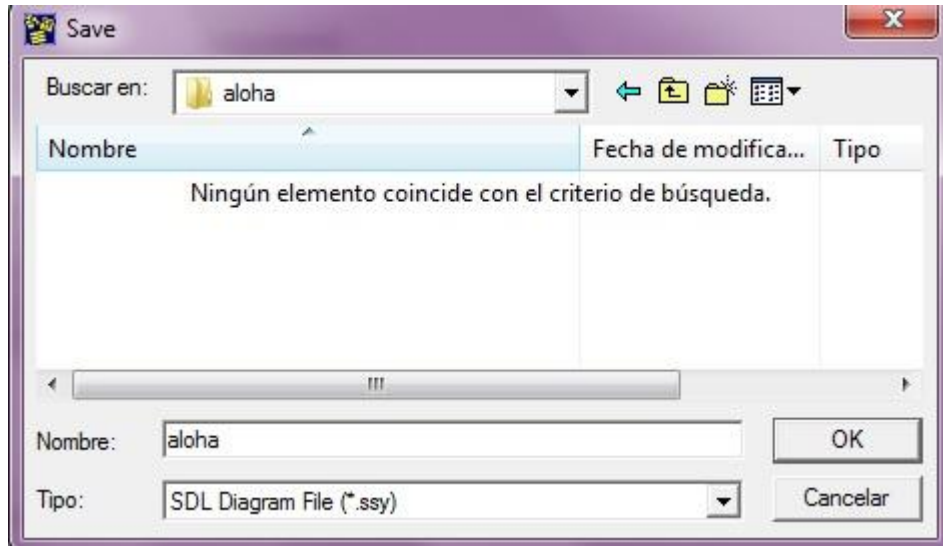


Figura 18: Guardando diagrama.

El campo tipo de archivo está predefinido *. Ssy, que es la extensión predeterminada de los archivos que contienen los diagramas de SDL.

- La lista muestra una lista de archivos que coinciden con el filtro de archivos. Debe encontrarse vacía pues no se ha creado ningún diagrama todavía.
- La lista *Buscar en* muestra la estructura de directorios desde el nodo raíz del sistema de archivos hasta el directorio actual. Puede hacer clic aquí con el fin de navegar por la estructura de directorios.

7. El Editor SDL sugiere un nombre para guardar el archivo, en este caso aloha.ssy. Se puede modificar el nombre si lo desea, pero suponemos que en este caso se dejará como está.

Simplemente pulse el botón OK y acepte el nombre del archivo. Ahora el diagrama se encuentra almacenado en el archivo. Si se fija en la ventana del Editor SDL, se puede ver que el diagrama se ha guardado en el archivo.

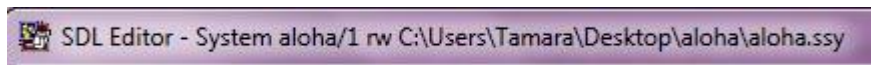


Figura 19: Título de la ventana Editor SDL.

Esta información está también disponible en la estructura del organizador, en el que se puede apreciar que el nombre del archivo ha cambiado de [unconnected] a aloha.ssy.



Figura 20: Estructura del diagrama después de haber sido guardado.

4.4.18 Guardar estructura de diagrama

Hasta el momento, ha guardado el diagrama del sistema. Pero también se debe guardar el diagrama en la ventana *Organizer* para futuras sesiones. Si se fija en el título que aparece en la ventana *Organizer*, se observa un asterisco al final. Este asterisco indica que se ha modificado la vista o la estructura del Organizador y necesita ser guardada.

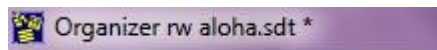


Figura 21: Título de la ventana Organizer

4.4.19 El archivo de sistema

El Organizador guarda su vista, junto con un número de opciones, en un archivo dedicado llamado el **archivo de sistema**. Los archivos de sistema se utilizan como un medio para mantener la coherencia de una estructura de SDL y proporcionar acceso inmediato a los diagramas que se definen en la estructura. El archivo de sistema está representado por su propio icono en la parte superior de la ventana *Organizer*, un rectángulo con "SDT" escrito en él. A pesar de que el archivo de sistema no ha sido guardado aún, el Organizador ha asignado un nombre de archivo para ello.

Para guardar el archivo de sistema:

1. Selecciona el comando *Save* (Guardar) desde el menú *File* (Archivo). Aparece la siguiente ventana:

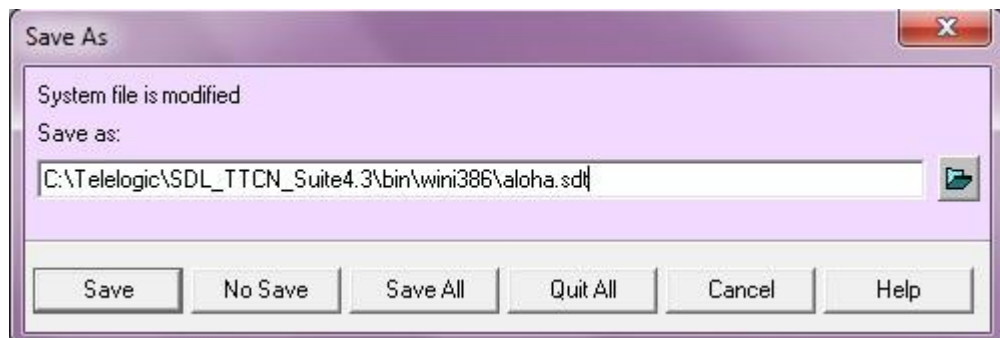


Figura 22: Ventana guardar en Organizer

2. La herramienta sugiere un nombre de archivo para almacenar la información: *aloha.sdt* (Los archivos de sistema son asignados a la extensión por defecto *.sdt*). Acepte la sugerencia y haga clic en el botón *Save*.

Una vez creado el archivo de sistema, la estructura del diagrama y las opciones de *Organizer* son guardadas para futuras sesiones. Puede abrir un sistema de archivo existente desde el menú *File* de *Organizer*.

4.4.20 Más acerca de guardar

A lo largo de este tutorial, ha aprendido a guardar diagramas individuales y también el archivo de sistema. Sin embargo existen otras maneras de guardar todo con un simple comando. Dos de esos métodos son explicados a continuación:

- Pulsando el botón *Save All* (Guardar todo) que aparece en la ventana de diálogo cuando guardamos desde Organizer. (Véase figura 22).



- Puede clicar el botón directo que se encuentra en la barra de herramientas de *Organizer*. Este botón guarda todos los diagramas, incluyendo la estructura de diagrama. (El botón *Save* situado en el editor SDL solo guarda el diagrama que contiene).

4.5 Imprimir Diagrama de Sistema

4.5.1 Qué va a aprender

- A imprimir un diagrama SDL.
- A ajustar opciones de impresión.
- A redimensionar una copia impresa

4.5.2 Como imprimir

Ya ha dibujado su primer diagrama SDL. Sería conveniente imprimir el diagrama antes de continuar con los siguientes ejercicios.

Para imprimir el diagrama:

- 1- Abra la ventana SDL Editor.
- 2- Seleccione el comando *Print* (Imprimir) desde el menú *File*. Se abrirá la ventana *Print*.



- Otra forma de hacerlo es pulsar el botón de acceso directo *Print*.

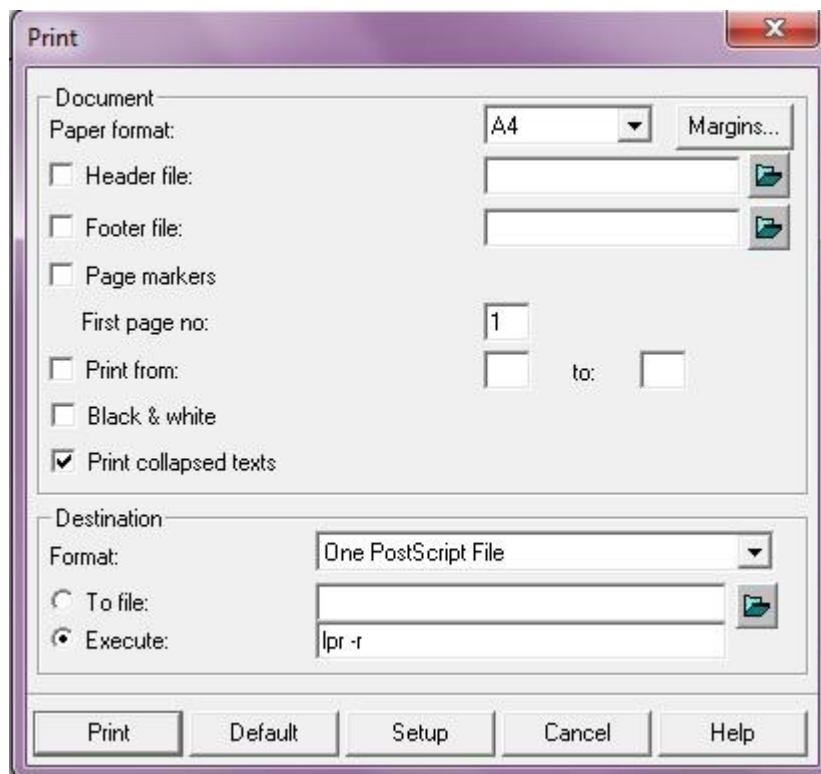


Figura 23: Ventana Imprimir

Editor y Analizador

3. Si ha configurado sus preferencias de manera adecuada, la ventana mostrará las opciones preestablecidas correctas. Si no es así, será necesario comprobar y modificar las siguientes opciones:

- La opción *Paper Format*(Formato del papel).
- *Destination Format*.
- Comando *Execute* (Ejecutar), regido por la preferencia *PrinterCommand*.

4. Una vez los ajustes sean correctos, ordene la impresión pulsando el botón imprimir. Un archivo de impresión se generará utilizando el comando *Execute* que ha especificado para imprimirlo.

Si el archivo se envía a una cola de impresión, la impresora debe responder de forma inmediata. Si no está satisfecho con el tamaño de la impresión resultante, puede redimensionarlo de la siguiente manera:

- Seleccione el botón *Setup*. Se abrirá la ventana *Print Setup*.

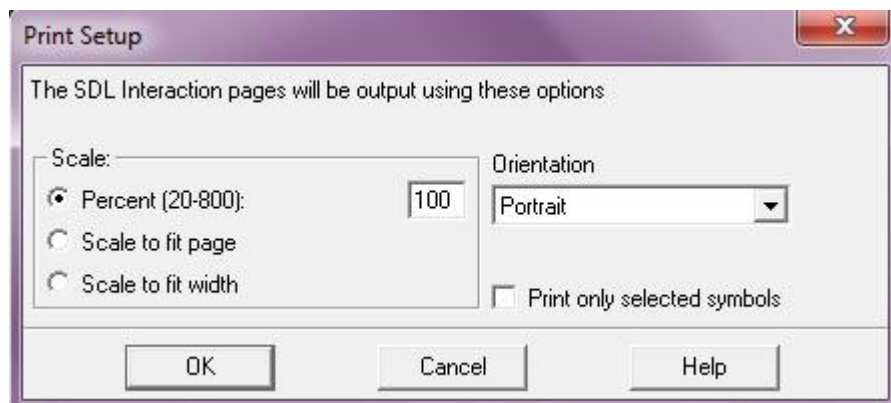


Figura 24: Ventana Print Setup

- Ajuste la escala al valor deseado y haga clic en OK.
- Ordene de nuevo la impresión pulsando otra vez en *Print*.

4.6 Comprobando el diagrama de sistema

4.6.1 Qué va a aprender

- A invocar el analizador
- Configurar opciones de análisis
- Trabajar con la ventana *Organizer Log*
- Localizar y corregir errores sintácticos

4.6.2 Ejecución del analizador

Ahora debería comprobar la sintaxis del diagrama de sistema que creó mediante la creación de los diagramas restantes. Para ello, va a utilizar la herramienta *Analyzer*, una herramienta de back-end que está totalmente integrada con el Organizador.

1. Seleccione el icono del diagrama de sistema SDL en la estructura de diagrama de *Organizer*.
Luego, desde el menú *Generate* (Generar) del Organizador, seleccione el comando *Analyze*.
2. Si ha modificado (quizás accidentalmente) cualquier diagrama, el Organizador primero le pregunta si desea guardar los diagramas modificados, en cuyo caso usted debe hacer clic en el botón *Save All* (Guardar todo) para asegurarse de que todo está bien.
3. Una vez que el cuadro de diálogo Guardar se cierra, se abre el cuadro de diálogo del analizador.

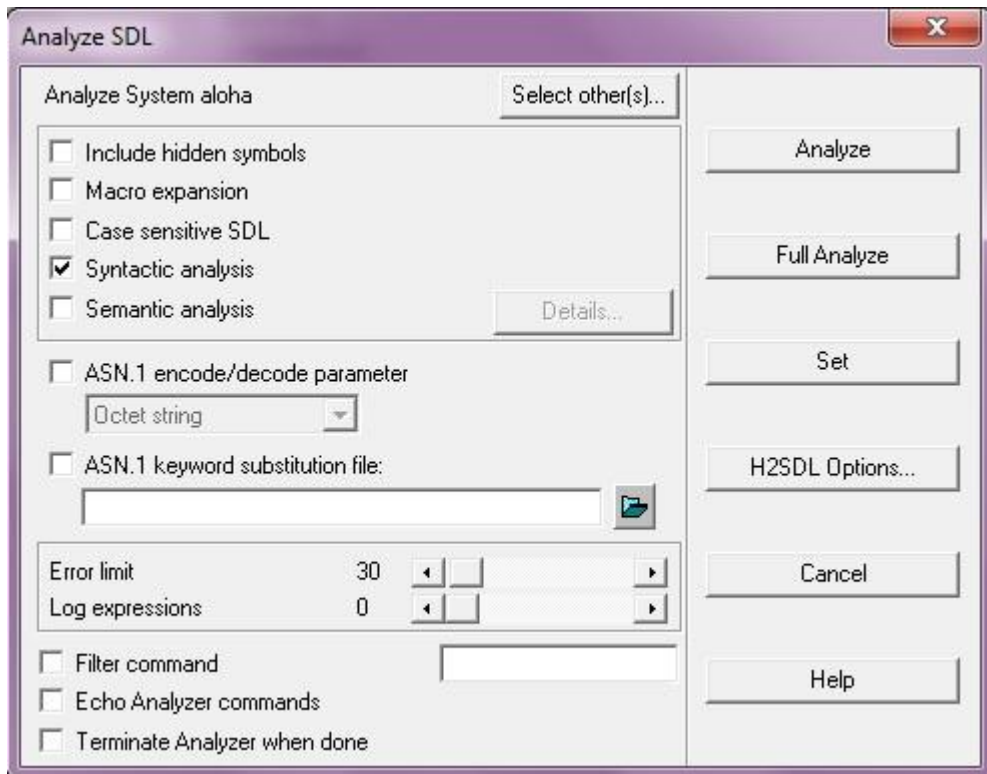


Figura 25: Ventana de diálogo *Analyzer*

3. Ajuste las opciones de la siguiente manera:
 - Macro expansión off
 - Syntactic analysis on
 - Semantic analysis off
 - Ajustar, si es necesario, la barra deslizante *Error Limit* a un valor razonable. Este parámetro define el número de errores y advertencias de las que se avisará antes de abortar el análisis.

4. Haga clic en el botón analizar. *Analyzer* comienza a procesar con las opciones marcadas anteriormente en el cuadro de diálogo. Cuando termine, la barra de estado del organizador debería mostrar algo como “*Analyzer done*”, posiblemente acompañado de información extra.

Editor y Analizador



5. *Organizer* proporciona una ventana de texto donde la información importante se registra. Por defecto, la ventana *Organizer Log* aparece en cuánto se registra un warning o error. Si la ventana no aparece automáticamente una vez finalizado el análisis, puede acceder a ella manualmente mediante el comando *Organizer Log* en el menú *Tools* o el botón de acceso directo.

4.6.3 Buscando errores de análisis

El diagnóstico ofrecido por el analizador se muestra en la ventana *Organizer Log* junto a otros mensajes importantes. Si se mira el final de análisis, ofrece un aspecto como el de la siguiente figura.

```
Syntactic analysis completed
+ Analysis completed
|
```

El texto "Número de advertencias" o "número de errores" muestra el número de advertencias o errores que se detectaron en el diagrama (si no se encontraron advertencias o errores, entonces estas líneas desaparecen por completo).

1. Para realizar este ejercicio, es posible que deba introducir un error sintáctico en el diagrama. Por ejemplo puede quitar uno de las comas que separan en la declaración de señales.

Un error de tipo sintáctico se detectó ya en el Editor de SDL, y ha sido marcado con un subrayado rojo, pero vamos a mostrar cómo el analizador informa del error.

2. Guarde todo y repita el análisis.

4.6.4 Corregir errores de análisis

La ventana *Organizer Log* muestra un error de la siguiente forma:

```
#SDTREF(SDL,C:\Telelogic\SDL_TTCN_Suite4.3\bin\wini386\aloha.ssy(1),209(35,45),4,1)
3: ERROR 312 Syntax error in rule SIGNALDEF, symbol Name found but one of the following expected:
, ; comment refinement
collision(Boolean),
?
```

4.6.5 Como interpretar el mensaje de error

- La primera parte (#SDTREF...) es una referencia al diagrama fuente, página, símbolo, número de línea y, finalmente, una posición dentro de una línea de texto donde se encontró el error. Todas las referencias producidos por el analizador se adhieren a este formato en su totalidad o parcialmente; la referencia no podrá en algunas circunstancias ser menos precisa que en el ejemplo anterior, dependiendo de la capacidad del analizador para localizar la fuente exacta del error.
- La segunda parte (ERROR 312 ...) contiene el número de error y un texto explicativo, que le dice, en este caso, que se esperaba una coma.
- La última parte colision (Boolean), junto con el carácter '?' especifica donde se produjo el error, en este caso que debería haberse insertado una coma detrás de (boolean).

Para mostrar el diagrama y símbolo donde se encontró el error, es posible utilizar una herramienta muy útil:

1. Seleccione, arrastrando el ratón, las líneas de texto que contienen el mensaje

```
#SDTREF(SDL,C:\Telelogic\SDL_TTCN_Suite4.3\bin\wini386\aloha.ssy(1),209(35,45),4,1)
3: ERROR 312 Syntax error in rule SIGNALDEF, symbol Name found but one of the following expected:
, ; comment refinement
colision(Boolean),
?
```

2. Seleccione el comando *Show Error* (Mostrar errores) desde el menú *Tools* (Herramientas).



- Como alternativa, puede hacer clic en el botón de acceso rápido *Show Error* (Mostrar mensaje de error) .

3. El símbolo que se encontró el error es seleccionado inmediatamente en el editor de SDL. Cuanta más información tiene la referencia, más precisa la selección . Corrija el error (inserte la coma) .

Editor y Analizador

4 . Para corregir el siguiente error, simplemente haga clic en el botón *Show Error* (Mostrar error) de nuevo.

5 . Guarde el diagrama y repita el análisis hasta que el analizador no informe de ningún error. Si usted se siente inseguro acerca de cómo interpretar y corregir los errores, mire el diagrama impreso.



- Usted puede borrar la ventana *Organizador Log* en cualquier momento, por ejemplo, entre los pasos subsiguientes para hacer más fácil la lectura del contenido del registro. Utilice el comando *Clear Log* (Borrar registro) desde el menú *Edit*, o pulse el botón de acceso rápido.



- Para repetir el análisis utilizando las mismas opciones, puede utilizar el botón de acceso rápido *Analyze* en la ventana *Organizer*.

Ahora ha diseñado su primer diagrama SDL utilizando la suite SDL. También ha comprobado que el diagrama es sintácticamente correcto de acuerdo a la Recomendación Z.100. ¡Felicidades!.

4.7 Creación de un nuevo bloque en el diagrama

4.7.1 Qué va a aprender

- A crear y dibujar un bloque
- A solicitar el apoyo del diccionario de señales
- Trabajar con varios diagramas utilizando el editor de SDL.
- Abrir nuevas ventanas en una página.
- Trabajar con múltiples ventanas de SDL Editor
- Llevar a cabo la comprobación de sintaxis en el bloque.

4.7.2 Creación de un diagrama de bloque desde Organizer

En este ejercicio, se va a crear un diagrama de bloque, empezando desde *Organizer*.

1. Localice la ventana *Organizer* y haga doble clic en el icono denominado Sistema_aloha. En una manera similar a la hora de crear el diagrama del sistema, obtendrá el cuadro de diálogo *Editar*. Asegúrese de que la opción *Show in editor* (Mostrar en editor) está activada y haga clic en el botón Aceptar.

A continuación, se abre el cuadro de diálogo *Add Page* (Agregar página). El cuadro de diálogo se utiliza para especificar el tipo de página (proceso o la interacción de bloque). El nombre de la página también puede ser especificado. Decida si desea que las páginas sean autonumeradas (1, 2, ... N). Esta funcionalidad está habilitada por defecto.

Editor y Analizador

2. Asegúrese de que *Process Interaction Page* se encuentre marcado y pulse OK.

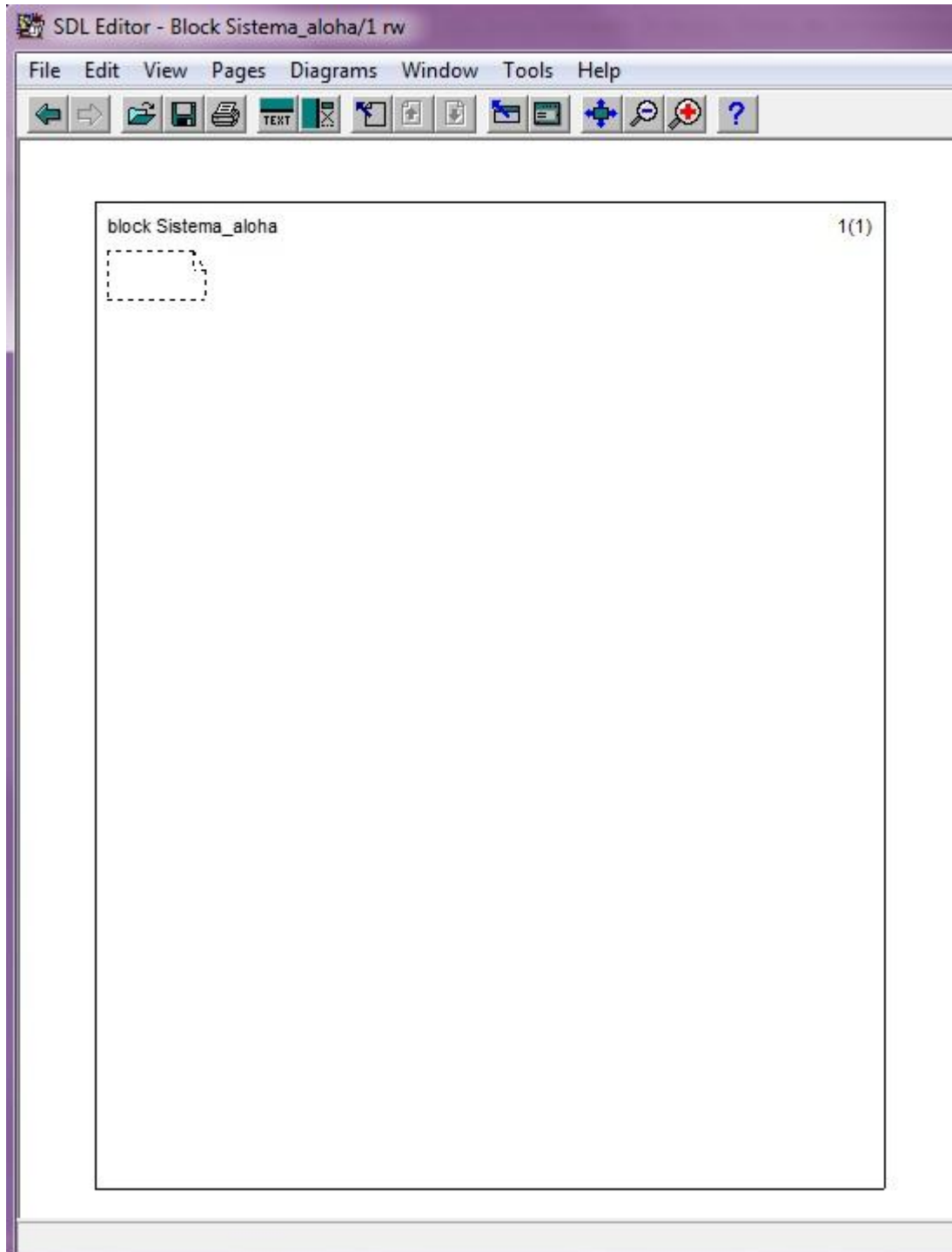


Figura 26: Process Interaction Page

Editor y Analizador

El Editor de SDL abre una ventana en la página 1 nuevo diagrama del bloque creado.

La ventana de editor de diagrama de bloques es similar a la ventana de diagrama de sistemas; sólo el menú de símbolo difiere.

La figura 37 muestra la apariencia del bloque Sistema_aloha terminado. Como usted puede ver, el diagrama contiene referencia a tres símbolos de procesos, cinco rutas de señales, y un cuadro de texto con aclaraciones del funcionamiento.

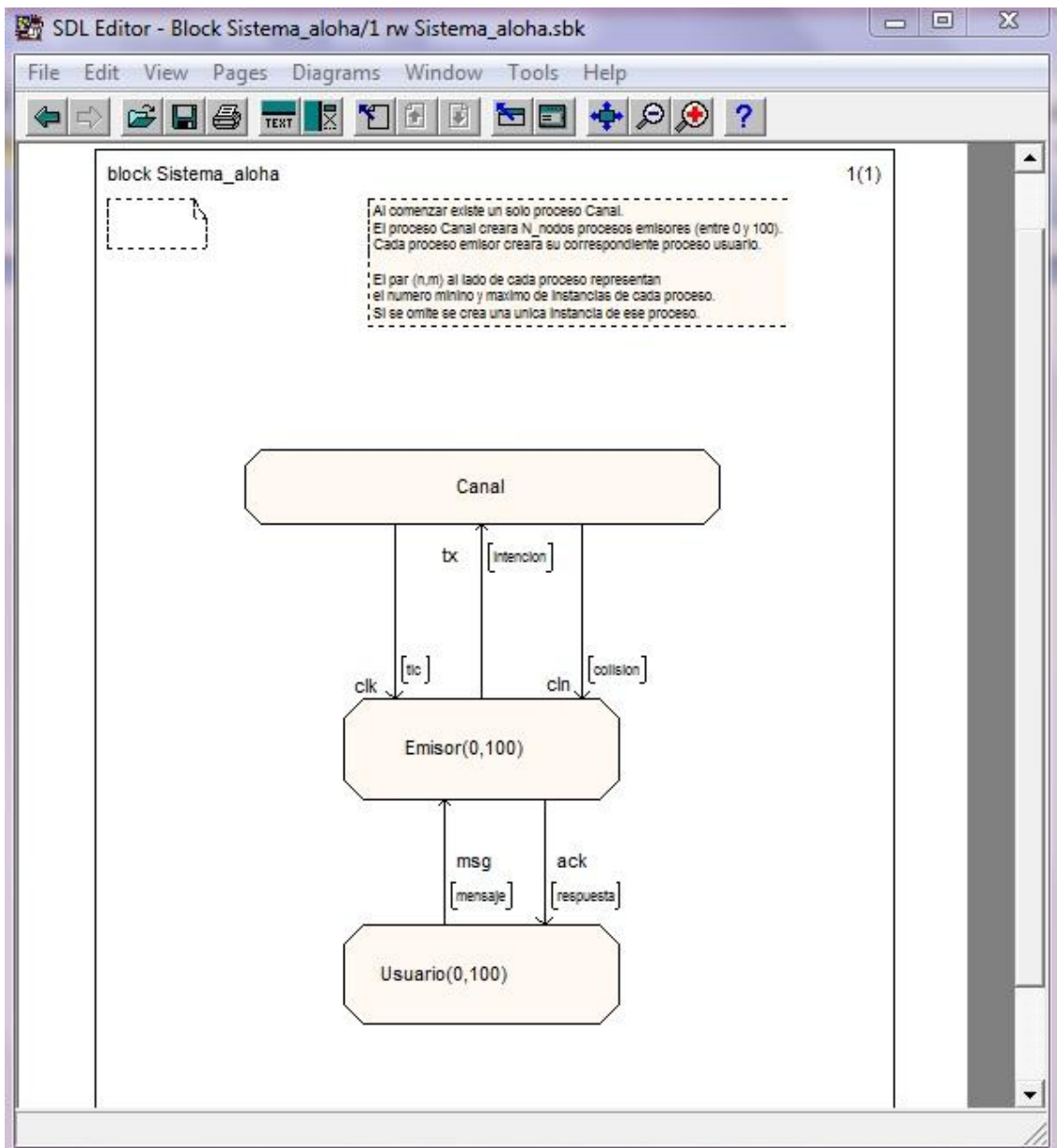


Figura 27: Bloque Sistema_aloha

Ahora debe dibujar el diagrama de bloques como se muestra en la Figura 27. Debe añadir los símbolos y líneas de una manera similar a la edición de diagrama del sistema. Por favor, tómese un minuto a leer los siguientes tres subapartados antes de iniciar el diagrama de dibujo, con el fin de familiarizarse con los nuevos conceptos que se introducen y cómo manejarlos.

4.7.3 Nombre de Procesos y Número de Instancias

Cuando se agrega un símbolo de referencia de proceso, debe especificar el número de instancias anexando el texto inmediatamente después del nombre del símbolo de referencia de proceso. El número de casos es el texto entre paréntesis, '()'.
()

4.7.4 Rutas de señales

Las rutas de señal se dibujan de una manera similar a los canales. Cuando se selecciona un símbolo de referencia de proceso, se muestran dos "asas".

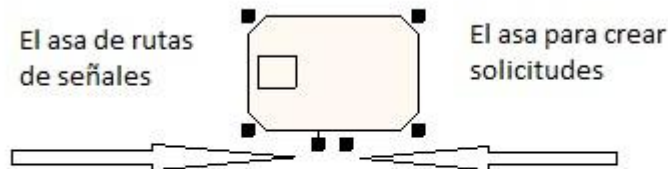


Figura 29: Las dos "asas" del símbolo de referencia de proceso

- El mango o asa de la izquierda se utiliza para la elaboración de rutas de señales, de una manera similar a la de los canales.
- El asa derecha se utiliza para dibujar la creación de peticiones. No se utiliza en este tutorial.

4.7.5 Conectando puntos

Al dibujar rutas de señales a / desde el símbolo de marco, debe no sólo rellenar la lista de nombres y la señal, sino también aprovechar los puntos de conexión gráfica para establecer conexiones entre las rutas de señal y el diagrama del sistema, es decir, la conexión de las rutas de señales a los canales. Al dibujar una ruta de señal, un objeto de texto se crea cerca del símbolo del marco. En este objeto de texto, se introduce el nombre del canal correspondiente. Esta situación no se produce en nuestro programa, pero puesto que es importante conocerla se ha creado un ejemplo. Vea la Figura 30.

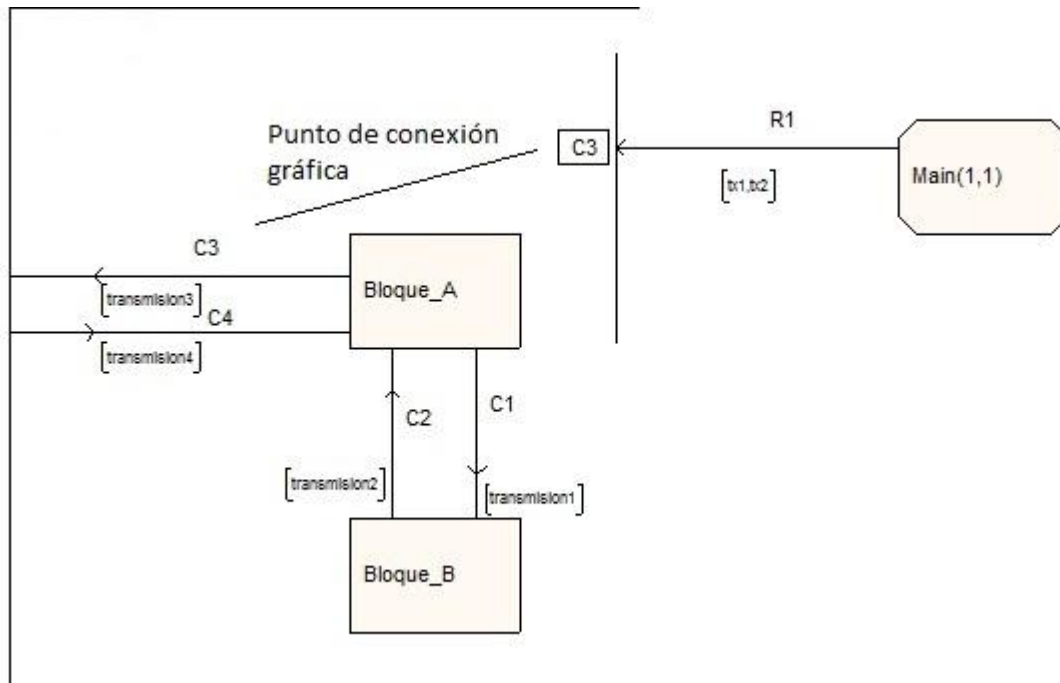


Figura 30: Punto de conexión gráfica

Para editar un punto de conexión:

- Sólo tiene que seleccionar e ingresar su contenido textual.

4.7.6 Editar Diagrama de Bloque

Ahora continúe con el ejercicio, dibuje el diagrama tal como se describe en los siguientes pasos:

1. Comience agregando el símbolo de referencia de proceso "Emisor (0,100)" y Canal.

A medida que vaya elaborando el diagrama del sistema, todo el texto que se introduce es sujeto a un análisis de sintaxis inmediata. Los errores se indican mediante subrayado en rojo, y desaparece tan pronto como se haya introducido el texto completo de acuerdo con la sintaxis SDL.

Recuerde que antes de que haya comenzado la edición de texto, el cursor de texto se encuentra sin parpadear. Pulsando <Delete> en esta etapa se elimina la totalidad símbolo seleccionado, en lugar de sólo un carácter escrito.

2. Dibuje una ruta de señal desde Canal hacia Usuario. Introduzca el nombre de la ruta de señal: clk.

4.7.7 Uso del diccionario de señal para señales individuales

Ahora debe especificar el nombre de las señales. El Editor de SDL tiene la capacidad de ayudarle en la reutilización de las señales que ya están definidos en la estructura de SDL (es decir, definidos en el diagrama del sistema), mediante una instalación conocida como el diccionario de la señal.

1. Seleccione el campo de texto de la lista de señales.
2. Seleccione el comando *Signal Directory* (Diccionario de señal) desde el menú *Window*.
Se muestra la ventana Diccionario de la señal. Si es necesario, muévelo para que se puede ver la lista de señales en la ventana Editor.

Nota:

Para que funcione correctamente, la utilidad Diccionario de señal requiere que las entradas en el diagrama SDL sean sintácticamente correctas. Si no, tiene que repetir el ejercicio anterior (Corregir errores de análisis) y ejecutar el Analizador con el fin de corregir cualquier error.

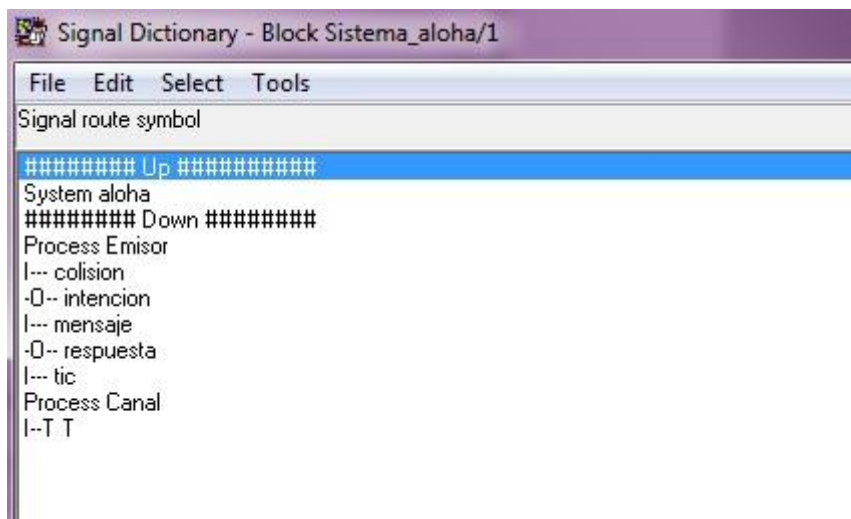


Figura 31: Diccionario de señales

3. Mire la lista de la izquierda en la ventana Diccionario de señales. En la primera sección se aprecia el separador *Up*. Esta sección incluye las señales que están disponibles al buscar en un nivel superior en la estructura del diagrama (es decir, en el diagrama matriz, aloha).

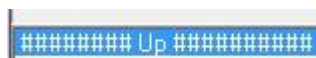


Figura 32: Separador *Up*

4. En esta sección, el primer elemento identifica el sistema aloha, como es de esperar:

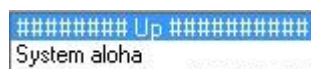


Figura 33: Elemento que simboliza el sistema aloha

5. Seleccione de la lista la señal “intención”.



Figura 34: Señal intención

6. Desde el menú *Edit*, pulse el comando *Insert*. La lista de señales se actualiza automáticamente en el editor SDL.
 - También puede hacer doble clic sobre la señal de la lista.

4.7.8 Completando el diagrama

1. Complete las siguientes señales entre Emisor y Canal tal como se muestra en la figura 27. Añada también el proceso Usuario y sus respectivas señales.
2. Una vez haya completado el bloque Sistema_aloha, guarde todo desde el botón de acceso rápido *Save* en la ventana *Organizer*. Acepte el nombre por defecto que le sugiere la ventana *Save*.

4.7.9 Trabajar con múltiples diagramas

Ahora ha creado dos diagramas SDL. Ambos diagramas se encuentran abiertos por el editor SDL; sin embargo sólo el diagrama que está siendo editado actualmente es mostrado en una ventana.

El Editor SDL ofrece un menú llamado *Diagrams* (Diagramas) donde se enumeran todos los diagramas y páginas actualmente abiertas por el editor.

1. Haga clic en el menú *Diagrams*, deben aparecer dos diagramas.

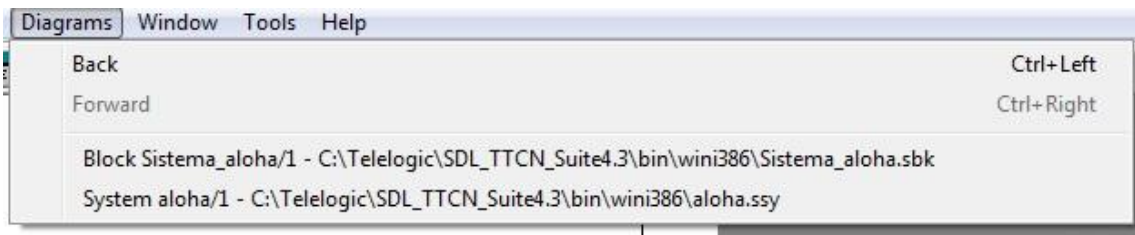


Figura 35: Menú *Diagrams*

Cada una de estas opciones de menú corresponde a un diagrama y página abierta actualmente por el editor SDL. El archivo donde se almacena el diagrama se muestra a la derecha del nombre de diagrama.

Ahora coloque el diagrama del sistema aloha a la vista:

2. Seleccione la opción System aloha/1.

El diagrama del sistema se muestra automáticamente. (El diagrama del bloque se encuentra ahora oculto).



- Las opciones de menú Back (Atrás) y Forward (Adelante) para navegar entre diagramas y páginas. El Editor de SDL realiza un seguimiento de qué páginas ha editado y se puede ir hacia atrás y hacia adelante en esta lista, de la misma forma que en un navegador al visitar páginas Web. También hay dos botones de acceso rápido para esto.

4.7.10 Trabajar con múltiples ventanas

Hasta el momento, sólo ha trabajado con una sola ventana en una página. El Editor SDL le permite abrir nuevas ventanas en el mismo diagrama, lo que permite trabajar en múltiples puntos de vista sobre una página. A esto también se le llama “instancia” de una ventana.

Para abrir una nueva ventana en una página:

1. Asegúrese de que la página de la que se va a crear una instancia es la página actualmente abierta en el Editor de SDL. Si no es así, utilice el menú Diagramas.
2. En el menú *Window*, seleccione el comando *New Window*(Nueva Ventana). Una nueva ventana que muestra la página actual se muestra al instante (ver Figura 36).
3. Ahora puede usar cualquier ventana para trabajar en la página. Cualquier cambio hace que las dos ventanas se actualicen de forma simultánea. Pruebe esto, por ejemplo, moviendo un símbolo!

Editor y Analizador

4. Es probable que tenga una ventana sólo para este tutorial, ya que los diagramas que está trabajando en este tutorial no son muy grandes.
Cierre cualquiera de las dos ventanas con el comando *Close Window* (Cerrar Ventana) en el menú *Window*.

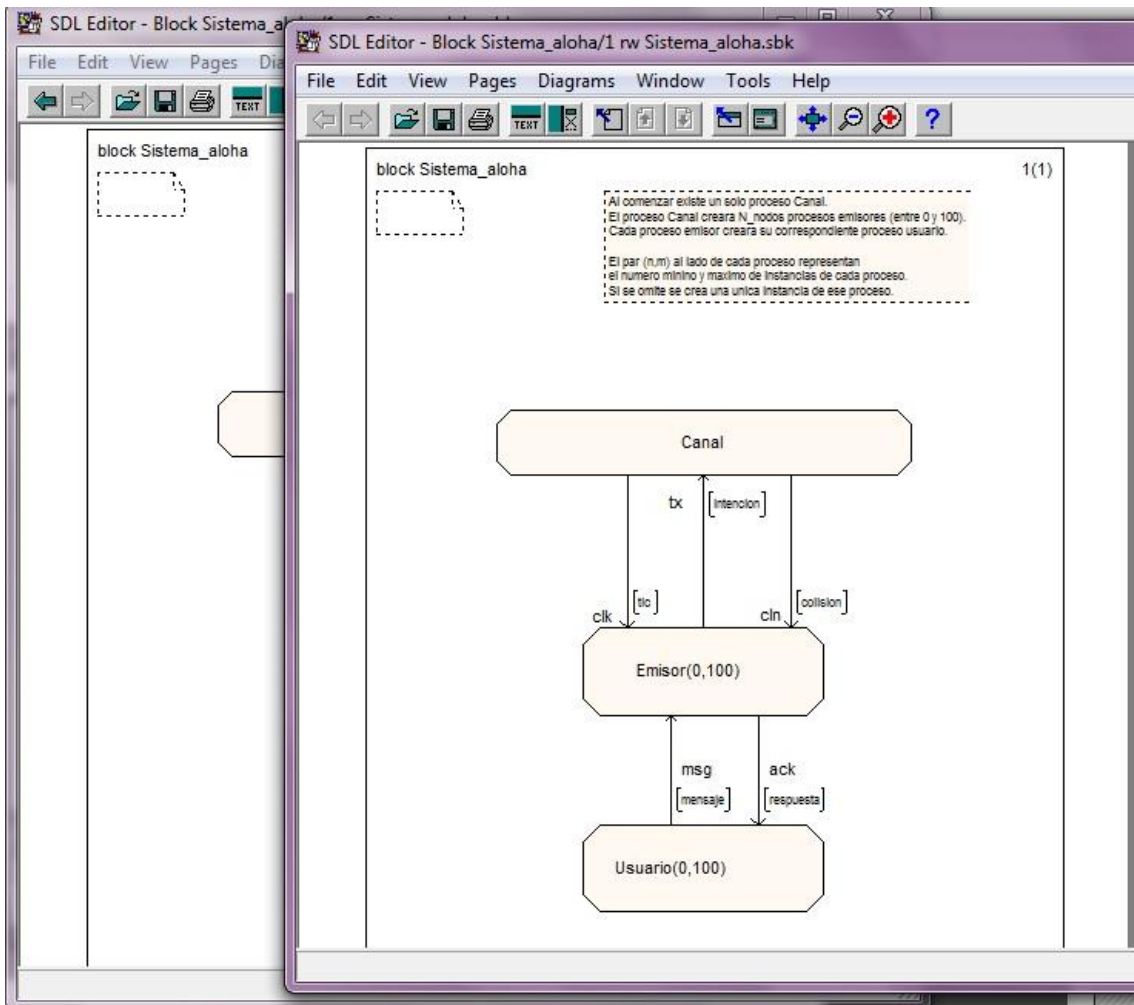


Figura 36: Dos ventanas de la misma página

4.7.11 Estado actual de *Organizer*

Guarde todo. La ventana *Organizer* debe tener el siguiente aspecto:



Figura 37: Aspecto actual de la ventana *Organizer*

4.7.12 Comprobación de la sintaxis del diagrama del bloque

Ahora es posible que desee utilizar el analizador para comprobar la sintaxis del diagrama del bloque que acaba de crear.

Para analizar el bloque *Sistema_aloha* haga lo siguiente:

1. Seleccione el icono del bloque *Sistema_aloha* para especificar el bloque como entrada al analizador.
2. Seleccione el comando *Analyze* (Analizar) y analice el diagrama de bloque.

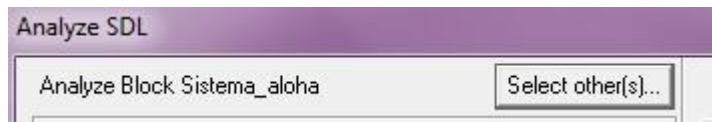


Figura 38: Especificando entrada a analizar

- Tenga en cuenta que el cuadro de diálogo *Analyze* le permite seleccionar que parte del sistema quiere analizar. Asegúrese de que el texto de la parte superior del cuadro de diálogo dice *Analyze Block Sistema_aloha*.
3. Proceda como en el diagrama del sistema:
 - Busque los errores de sintaxis desde Organizer Log.
 - Corrija los errores.
 - Repita el procedimiento si fuera necesario.

4.8 Crear un diagrama de proceso

Ha creado los elementos estructurales de su sistema SDL. Esta estructura debe completarse con la implementación de los diagramas de flujo de proceso que describen el comportamiento del sistema.

En los ejercicios anteriores, ha aprendido cómo crear nuevos diagramas, ya sea por el Organizador o desde el Editor de SDL, así que no se debe centrar más en esos detalles. Siéntase libre para hacer doble clic en los iconos del Organizador o en el Editor de SDL, o utilizar el comando Editar Organizador, dependiendo de su preferencia.

En el próximo ejercicio, aprenderá como utilizar el Editor de SDL para la elaboración de diagramas de proceso. Comience con el proceso Emisor.

4.8.1 Editar el proceso Emisor

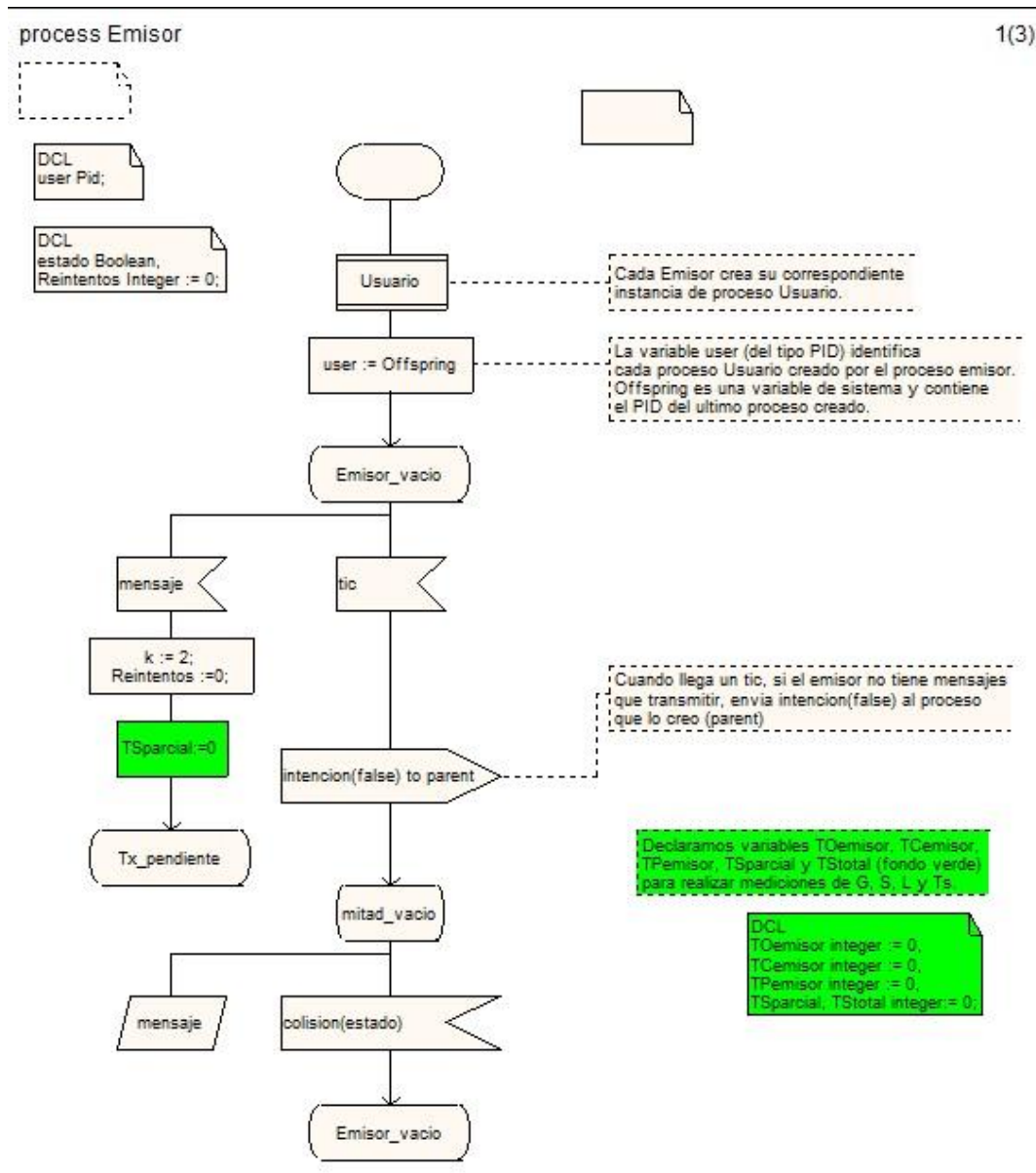


Figura 39: Proceso Emisor página 1.

process Emisor

2(3)

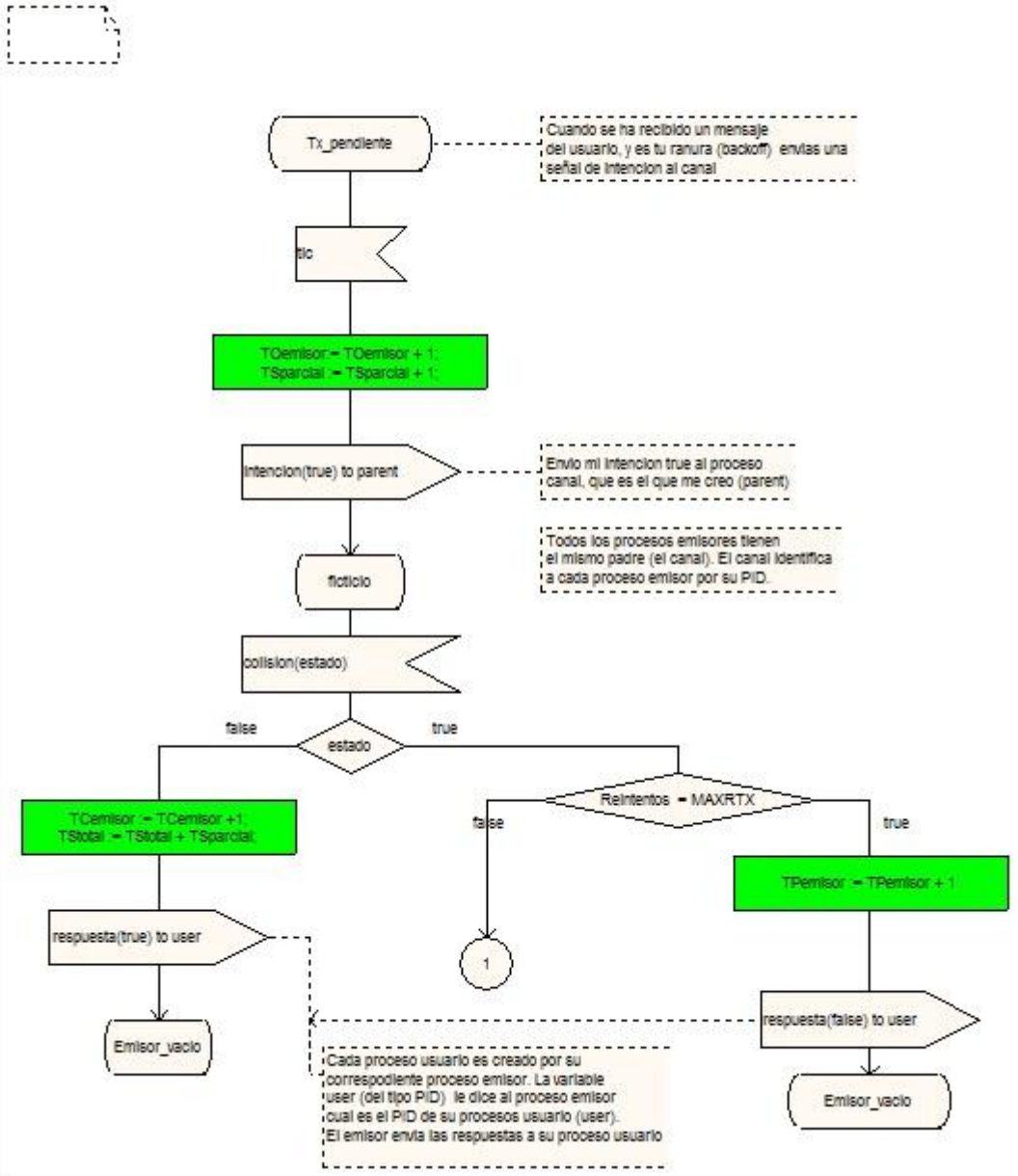


Figura 40: Proceso Emisor página 2.

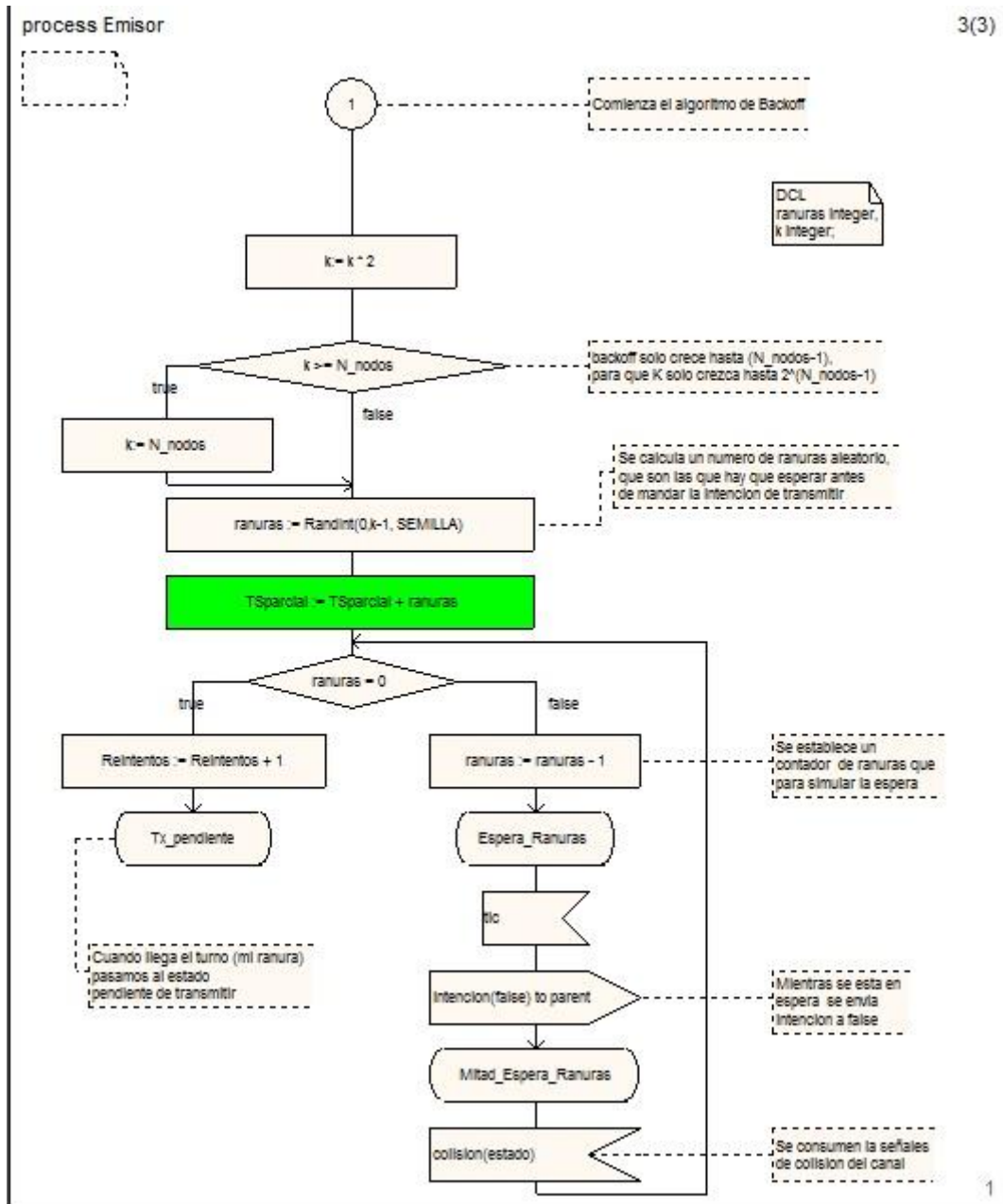


Figura 41: Proceso Emisor página 3.

Editor y Analizador

El proceso Emisor abarca la creación de usuarios, la gestión de transmisión y retransmisión de mensajes, y también del back-off.

En las siguientes páginas, encontrará sugerencias sobre cómo utilizar el editor SDL para dibujar el diagrama.

4.8.1.1 Qué va a aprender

- A añadir símbolos mediante doble clic.
- A trabajar con las funciones del portapapeles.
- Insertar símbolos en un diagrama de flujo

4.8.1.2 Creando el diagrama

1. Edite el diagrama Emisor. Cuando se le pregunte si desea agregar una página, asegúrese de que especifica una página con el tipo establecido *Graph Page*.

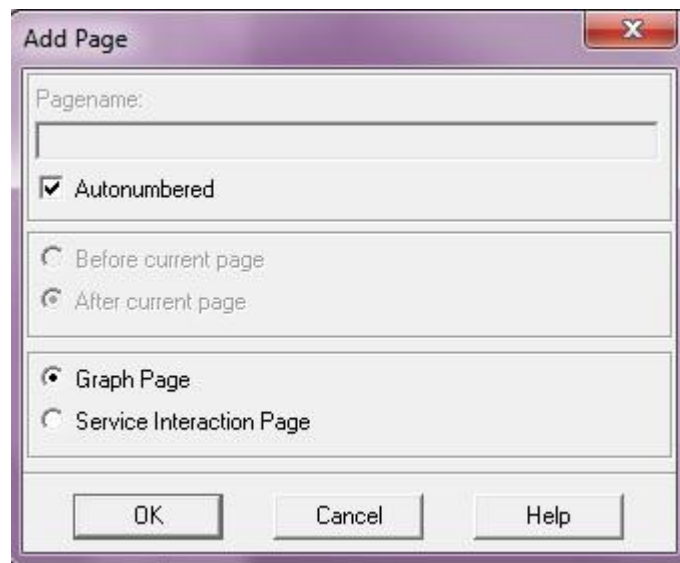


Figura 41: Especificar tipo de página como “página gráfica”

Editor y Analizador

2. Cuando el Editor de SDL responde mostrando el diagrama vacío, observe que el aspecto del menú de símbolos es diferente; ahora contiene los símbolos utilizados en un diagrama de flujo (tales como símbolos de entrada y de salida). Los símbolos más importantes serán explicados durante la realización del ejercicio.

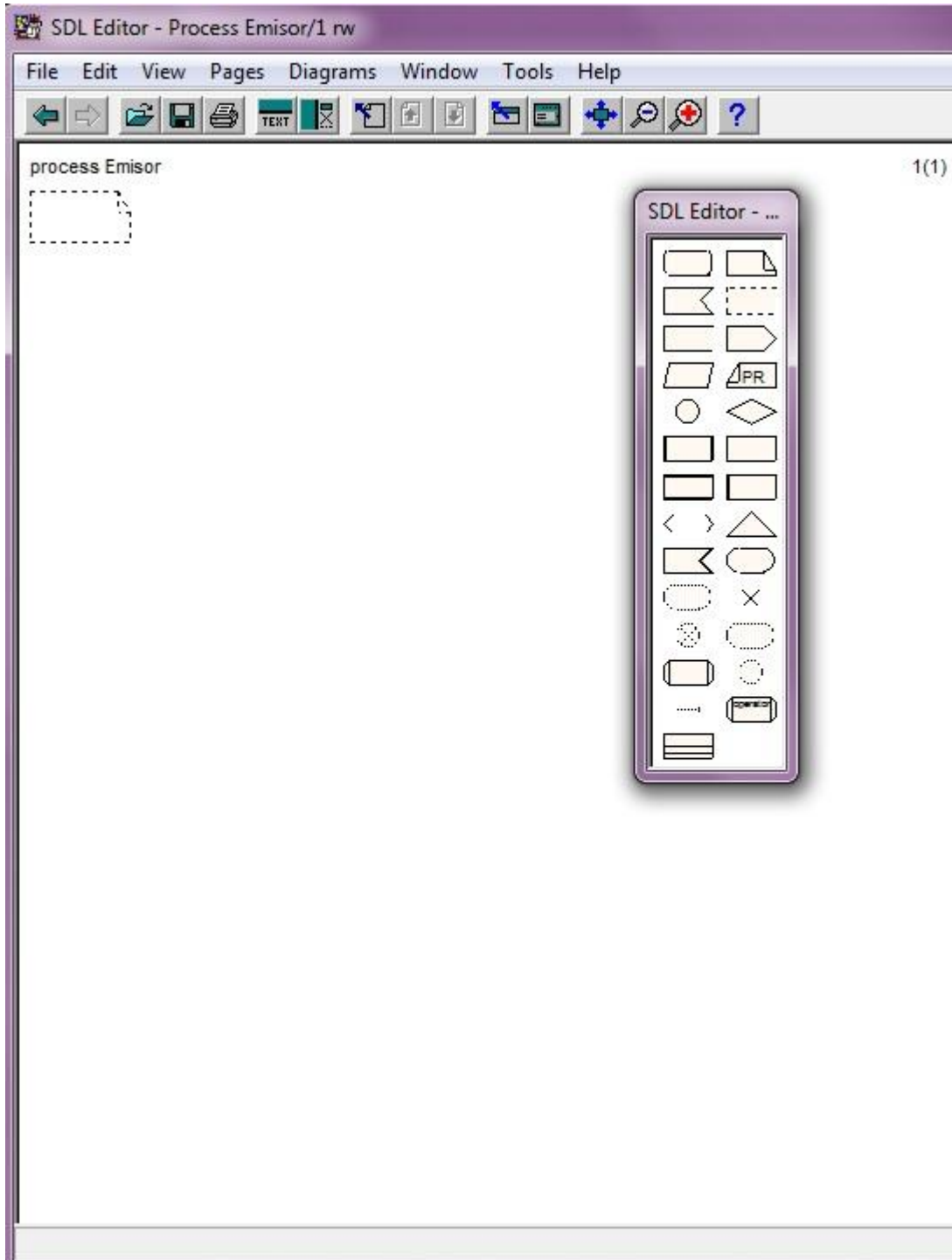


Figura 42: Ventana SDL para diagrama de flujo

Editor y Analizador

3. Puede seleccionar para introducir el texto en cada símbolo una vez el símbolo ha sido insertado, o insertar todos los símbolos del diagrama y luego editar el texto; o una mezcla de ambos.

En esta primera página, se crea al Usuario y se queda a la espera de una de las dos señales principales, **tic** (procedente del Canal) y **mensaje** (procedente del Usuario).

Dibujando página 1



4. Comience la realización del diagrama con el símbolo *Start*, que indica el comienzo del proceso.



5. Añada el símbolo de llamada a un proceso existente. En este caso el proceso **Usuario**.



6. Continúe con el símbolo de estado del sistema **Emisor_vacio**.

7. Ahora el diagrama se divide en dos ramas.

Dibujando rama de la izquierda

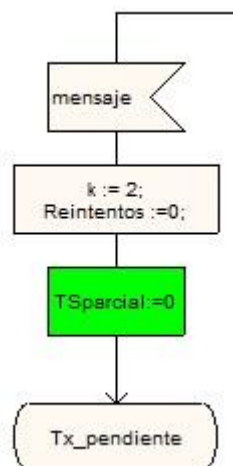


1. Dibuje el símbolo de señal de entrada **mensaje**. Estos símbolos deben encontrarse inmediatamente después de un símbolo de estado. Si la señal que hemos recibido es consumida se produce una transición de estado, poniendo la información transportada por la señal a disposición del proceso.



2. A continuación introduzca dos símbolos de tareas definiendo las siguientes variables $k:=2$; $\text{Reintentos}:=0$; y $\text{TsParcial}:=0$. Estos símbolos son usados para asignación de variables u operaciones con temporizadores. Pueden contener una lista de asignaciones separadas por comas.

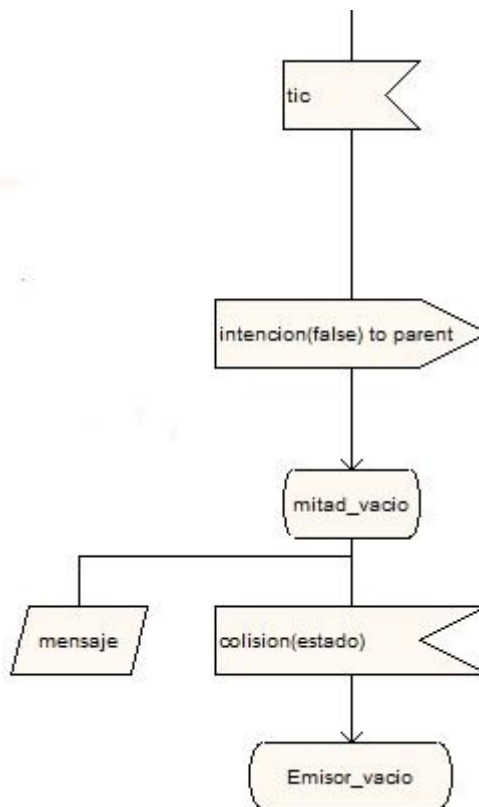
3. Para finalizar añada el estado **Tx_Pendiente**.



Editor y Analizador

Dibujando rama de la derecha

1. Añada el símbolo de señal de entrada **tic**.
2. Añada el símbolo de señal de salida **intención(false) to parent**. Cuando llega un tic, si el emisor no tiene mensajes que transmitir, envía intención (false) al proceso que lo creo (parent).
3. A continuación, añade el símbolo de estado **mitad_vacio**.
4. Del estado anterior surgen dos sub-ramas, la de la izquierda contiene la señal **mensaje**.
5. La rama de la derecha consiste en una señal de entrada **colision (estado)** y el estado **Emisor_vacio**.



6. Finalmente añade dos símbolos de texto con las siguientes declaraciones: **user Pid: estado Boolean; Reintentos integer:=0;**. Los símbolos de texto contienen las declaraciones de estructuras, variables, temporizadores y señales. Para la declaración de variables se debe utilizar la palabra DCL seguida del nombre y el tipo de variable que se quiera declarar.

Editor y Analizador



7. Cabe destacar el símbolo para añadir comentarios de texto.
8. Ha finalizado la edición de la primera página que contiene el proceso Emisor. A continuación proceda con las siguientes.

Dibujando página 2

En esta segunda página, se establece cuando se recibe la señal **mensaje** del Usuario, es decir cuando se quiere transmitir.

1. Comience añadiendo el símbolo de estado **Tx_pendiente**.
2. A continuación el símbolo de entrada **tic**.

Defina las variables

TOemisor:= TOemisor + 1;

TSparcial:=TSparcial + 1;

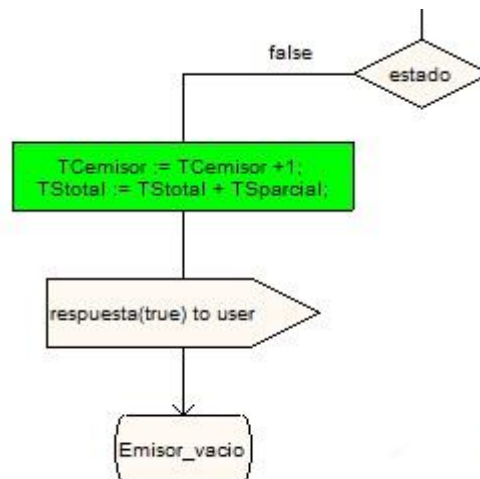
3. Añada la señal de salida **intencion(true) to parent** (envío intención true al proceso Canal, que es el que me creó); así como el estado **ficticio** y la señal de entrada **colision(estado)**.
4. Añada el símbolo de decisión **estado**. El símbolo de decisión es usado para escoger entre dos rutas alternativas según el resultado de una condición.
5. Si la señal es false, indica al Usuario que todo ha ido bien. **Elabore la rama de la izquierda** mediante la definición de las variables :



TCemisor:= TCemisor +1;

TStotal:= TStotal + TSparcial;

Añada la señal de salida **respuesta(true) to user** y el estado **Emisor_vacio**.

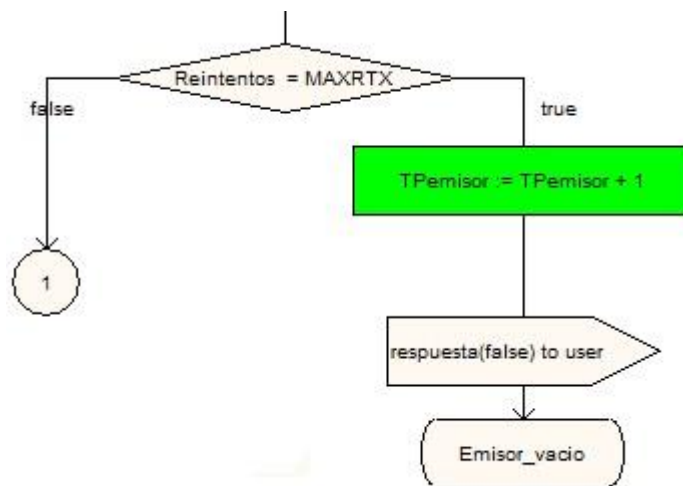


Editor y Analizador

6. Elabore la rama de la derecha mediante la adición de el símbolo de decisión **Reintentos = MAXRTX**. Si la señal es true > colisión, se reintentará transmitir pasando por back-off un número de veces MAXRTX.
7. La rama de la izquierda nos conduce a un conector, que simboliza la continuación con otro conector que posea la misma etiqueta, y es utilizado para realizar saltos incondicionales.
8. La rama de la derecha consta de la declaración :

TP emisor:=TPemisor +1;

Así como de la señal de salida **respuesta(false) to user** y el estado **Emisor_vacio**.



Dibujando página 3.

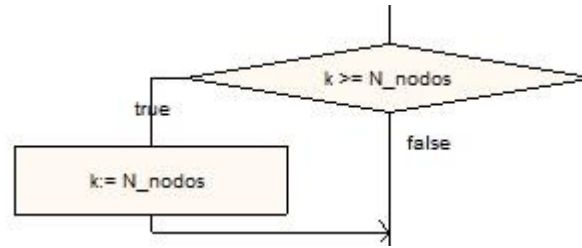
En esta tercera página, se cuenta el tiempo de espera hasta el próximo reintento. (tiempo back-off).

Se genera un número aleatorio, que es el número de ranuras que se va a esperar.

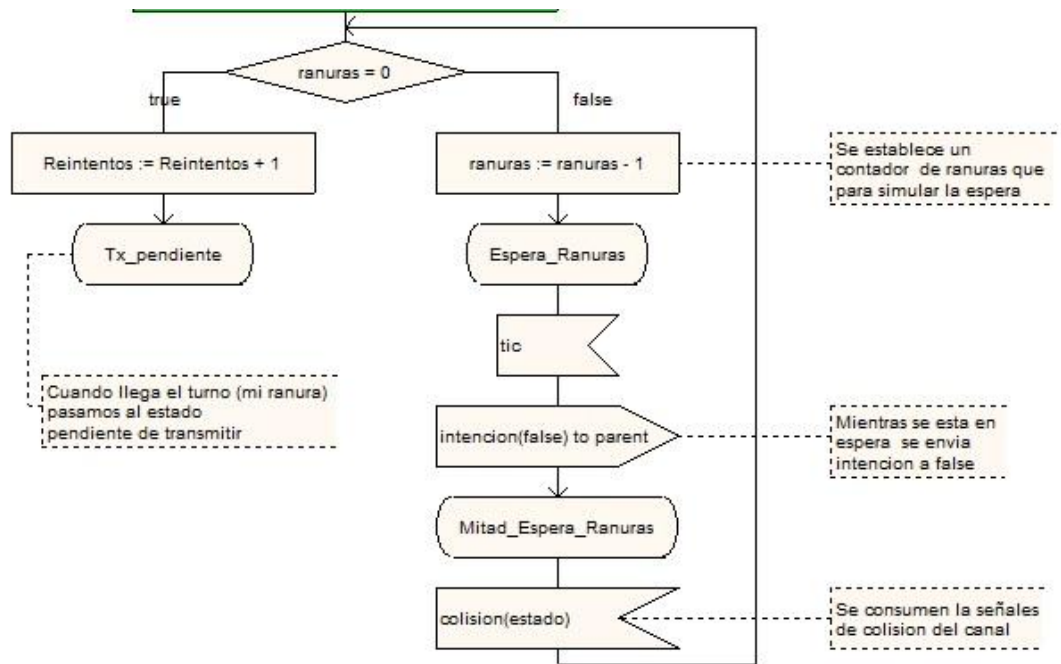
1. La página 3 consiste en la realización del algoritmo de Back Off del proceso.
2. Comience con la adición del símbolo conector previamente añadido.
3. Declare en el símbolo de tarea **k:= k*2;**

Editor y Analizador

- Añada el símbolo de decisión $k \geq N_nodos$. Back Off sólo crece hasta $N_nodos - 1$, para que k sólo crezca hasta $2^{(N_nodos - 1)}$.



- Se calcula un número de ranuras aleatorio mediante la declaración $ranuras := RandInt(0, k-1, SEMILLA)$. Y añada el símbolo tarea con la declaración $TSparcial := TSparcial + ranuras$.
- Finalice el diagrama como se muestra a continuación.



- Ha finalizado la edición del proceso Emisor.

4.8.2 Editar Proceso Usuario

Envía de forma aleatoria, y según una probabilidad PG, un mensaje a Emisor para indicar que tiene algo que transmitir.

Primero, cree el diagrama del proceso Usuario de la forma habitual.

4.8.2.1 Qué va a aprender

- A solicitar ayuda gramática.

Proceda a la realización del diagrama de flujo de la misma forma que en el apartado anterior.

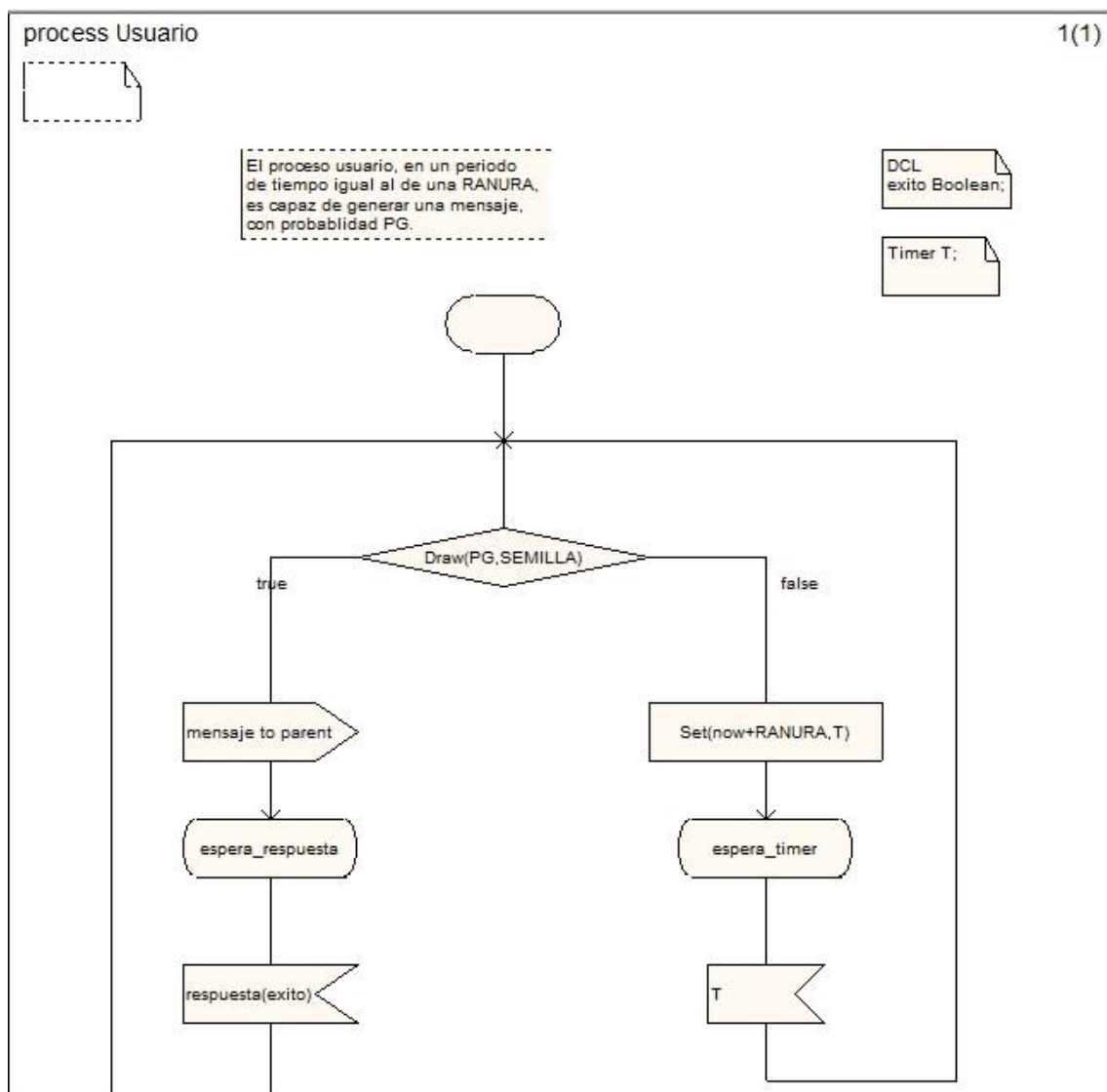


Figura 43: Proceso Usuario.

Editor y Analizador

1. Comience con el símbolo **Start**.
2. Añada el símbolo de precisión **Draw(PG, SEMILLA)**.
3. Si se ha generado el mensaje, se procede por la rama de la izquierda, consistente en el envío de la señal **mensaje to parent**, el estado **espera_respuesta** y la recepción de la señal **respuesta(éxito)**.
4. Si no se genera mensaje, haga clic en el símbolo de tarea. En esta tarea se establece el contador de tiempo, por lo que se va a suponer que para este ejercicio no tiene la gramática de la función **Set** en mente.
5. El editor de SDL proporciona una ventana llamada *Grammar Help*, que le ayuda a introducir expresiones de SDL correctamente. Va a utilizarla a continuación.
6. Seleccione el comando *Grammar Help* desde el menú *Window* en el Editor SDL.

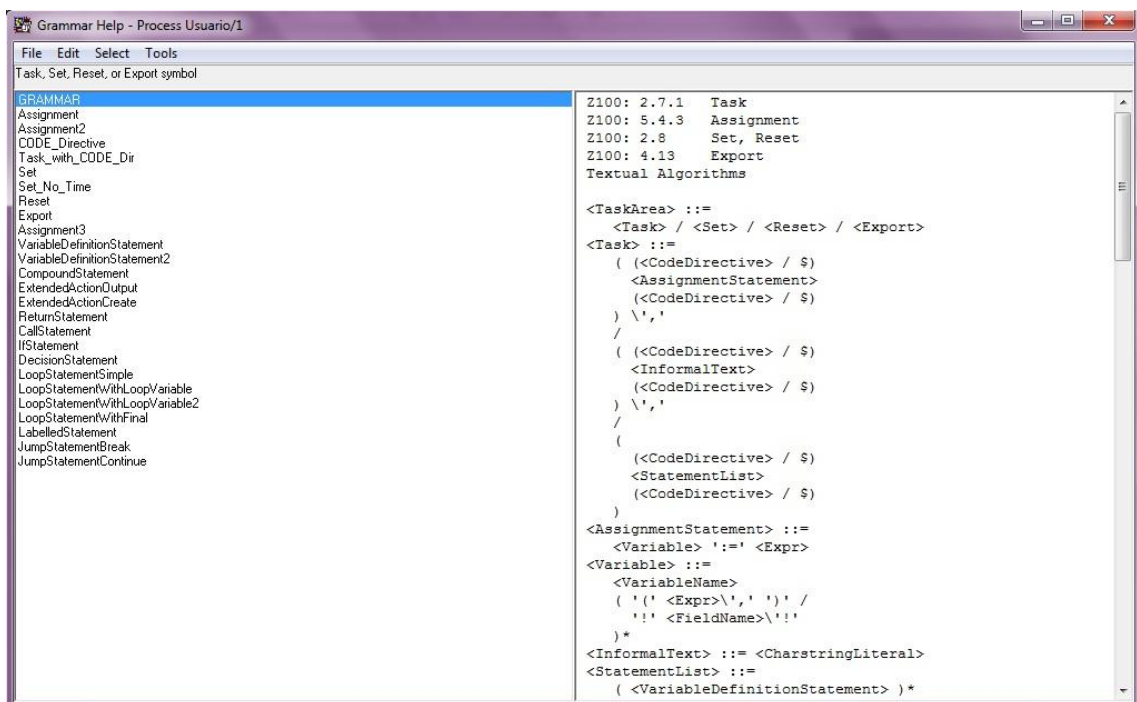


Figura 44: Ventana *Grammar Help*

La lista de la izquierda muestra una serie de “casos de uso”, cada uno identificado con su nombre. La primera opción es la gramática del objeto seleccionado.

La parte superior de la lista de la derecha muestra una serie de referencias a la definición Z100.

Editor y Analizador

Debajo de las referencias Z.100 se enumeran los elementos textuales formales (SDL / PR), expresiones que son legales para agregar al símbolo. (Las expresiones formales deben por supuesto ser reemplazadas por los valores actuales que se utilizan en su contexto).

El caso de uso que va a utilizar es el de un temporizador, por lo que debe localizar el elemento titulado Set en la lista de la izquierda y seleccionarlo.

La lista de la derecha se actualiza para reflejar la gramática formal de la expresión: "SET (Now + Expr, TimerName)".

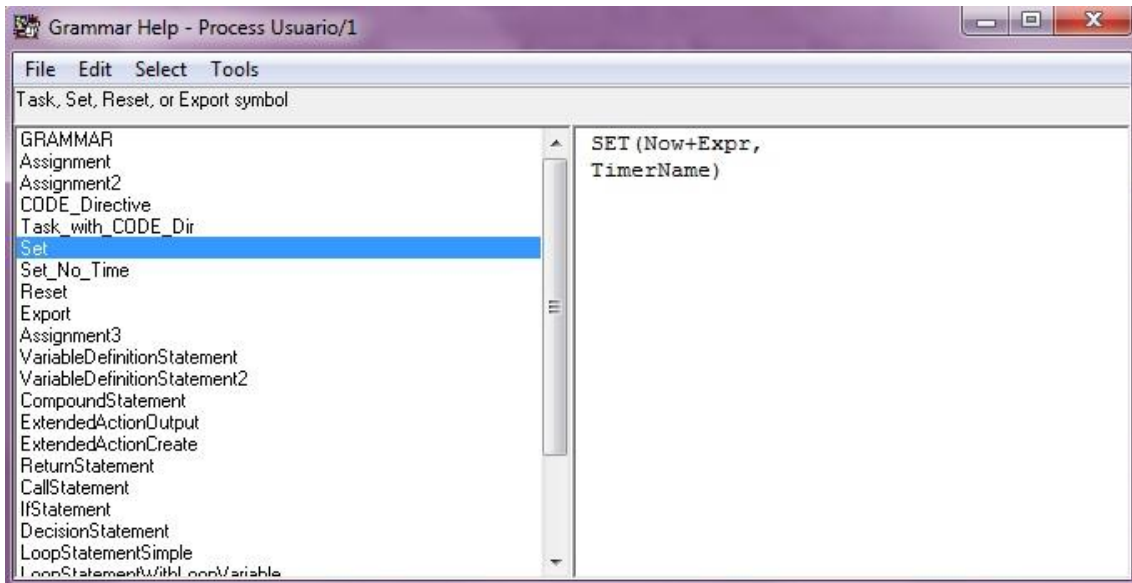


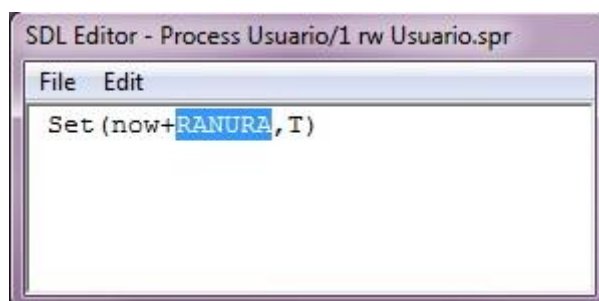
Figura 45: Gramática de la función Set

Introduzca el texto en el símbolo tarea seleccionando la opción *Insert* desde el menú *Edit*. El símbolo de tareas se actualiza inmediatamente.

- También puede hacer doble clic en el elemento situado en la lista.

Ahora, cambie los nombres genéricos *Expr* y *TimerName* a sus valores actuales (RANURA y T, respectivamente).

Utilice la ventana de texto del Editor de SDL para realizar los cambios.



Editor y Analizador

7. Añada el estado **espera_timer** y la señal de entrada **T**.
8. Por último es importante que se encuentren los símbolos de texto con las declaraciones éxito **Boolean** y **Timer T**.
9. A finalizado la edición del proceso Usuario.

4.8.3 Editar proceso Canal

Se encarga de generar los N emisores. En cada tic, pregunta si se quiere transmitir, ve quien transmite y comprueba si hay colisión de los mensajes enviados al Canal.

4.8.3.1 Qué va a aprender

- Declarar un procedimiento al que se llamará en el proceso actual.

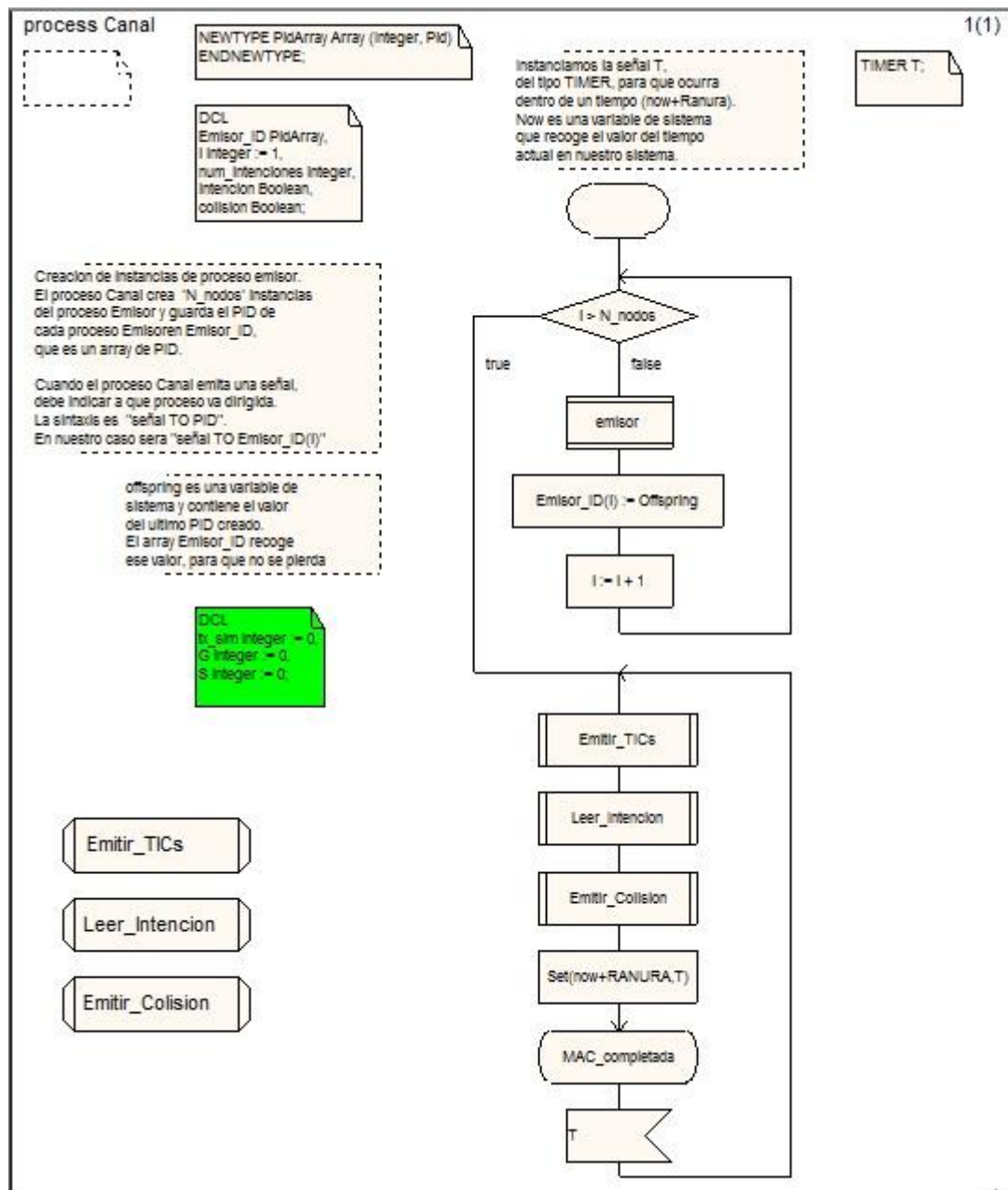


Figura 46: Proceso Canal

procedure Emitir_TICs

1(1)

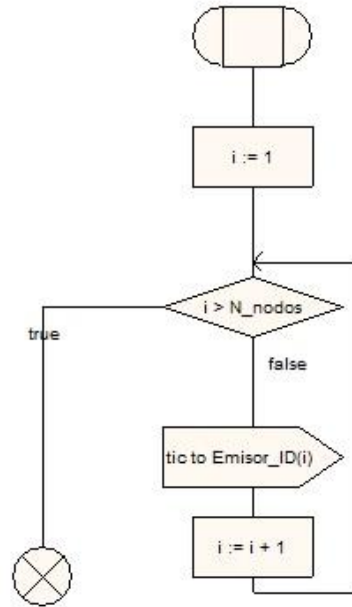


Figura 47: Procedimiento Emitir_TICs

procedure Leer_Intencion

1(1)

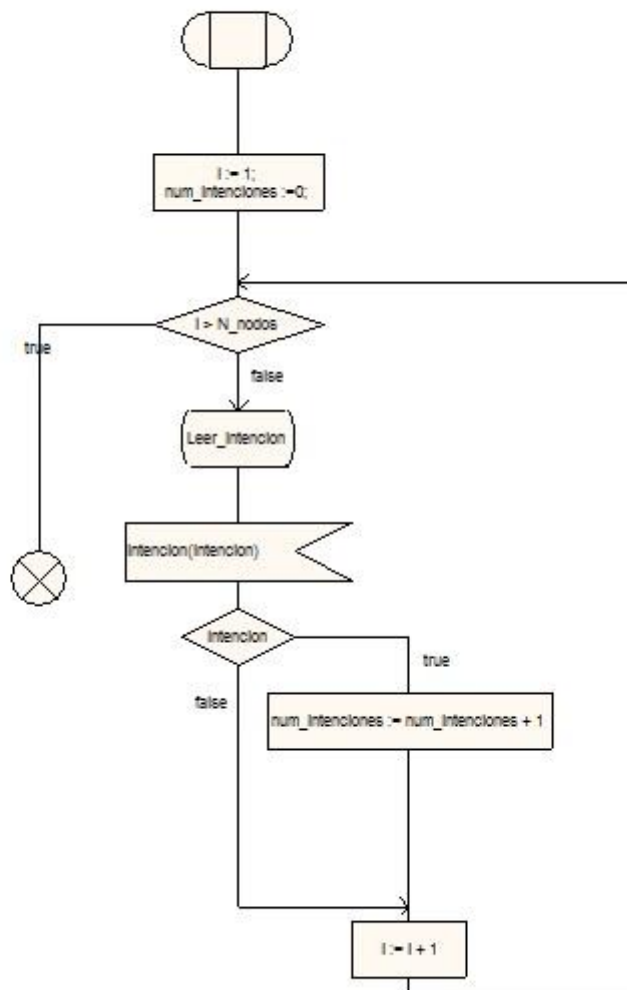


Figura 48: Procedimiento Leer_intencion

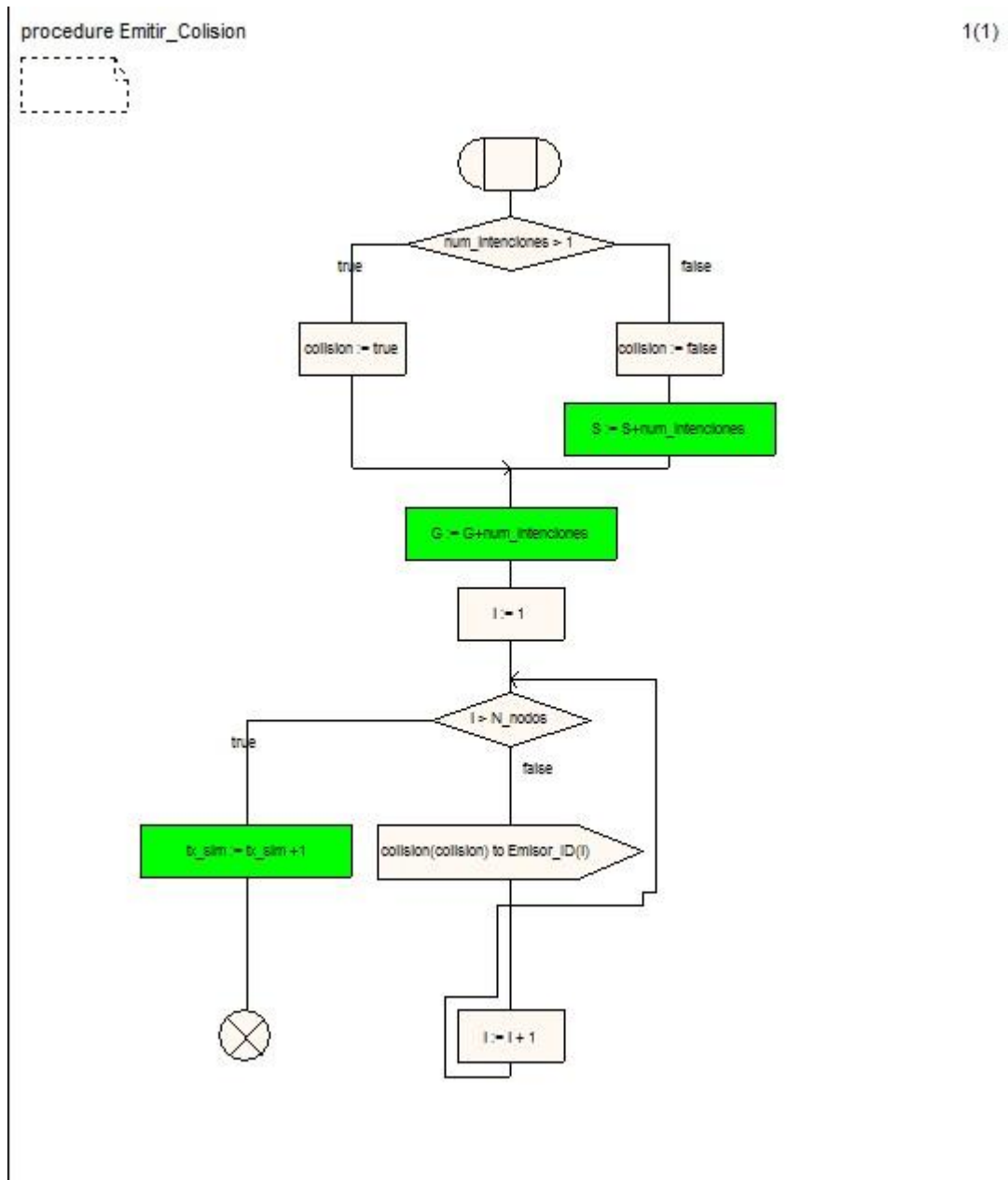
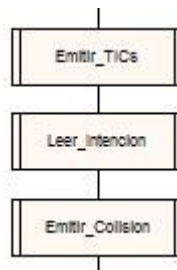


Figura 49: Procedimiento Emitir_colision

Editor y Analizador

Puesto que la realización del proceso Canal se realiza de la misma forma vista anteriormente, se va a explicar la llamada a procedimientos dentro del mismo proceso.



Como se puede ver se realizan llamadas a tres procedimientos que se van a crear dentro del propio proceso Canal, y que vienen representados en el diagrama de flujo de la siguiente manera.



Emitir_TICs : envía una señal tic a cada Emisor.

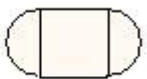
Leer_Intencion: Una vez preguntado el Emisor si desea transmitir, se comprueba cuántos de ellos quieren hacerlo, en caso de que sean 2 o más se producirá colisión, pues sólo uno puede ocupar la ranura.

Emitir_Colision: Sabiendo quien quiere transmitir, se conoce si se produce colisión o no. Se informará a todos los Emisores para saber si ha habido éxito o se ha producido colisión.

Comience la creación del procedimiento **Emitir_TICs** haciendo doble clic sobre el símbolo del procedimiento.



Figura 50: Creando el procedimiento Emitir_TICs



1. Comience añadiendo el símbolo de inicio de procedimiento.

2. Añada el símbolo de tarea **i:= 1**.

3. A continuación el símbolo de decisión **i>N_nodos**.

4. Si la condición no se produce, añada la señal de salida **tic to Emisor_ID(i)** y el símbolo de tarea **i:= i +1**.



5. Si se produce la condición, finalice el procedimiento con el símbolo retorno de procedimiento, que vuelve al momento en el diagrama en el que fue llamado.

6. Proceda a realizar los dos procedimientos restantes, **Leer_intencion** y **Emitir_colision** de la misma manera.

4.9 Más acerca de Organizer

Cuando el diagrama esté listo, guárdelo todo. La estructura del diagrama en la ventana *Organizer* debería tener el siguiente aspecto:



Figura 51: Estado actual de la estructura del diagrama.

En este tutorial, sólo ha podido utilizar una pequeña parte de la funcionalidad disponible. Por ejemplo, puede personalizar la ventana *Organizer* para mostrar la información a través de diferentes opciones de visualización.

4.9.1 Que va a aprender

- A trabajar con árboles verticales.
- A expandir y contraer la estructura del diagrama.
- A reorganizar diagramas en una estructura organizador.
- Mostrar directorios y páginas.
- Imprimir todo el sistema.

4.9.2 Vista de árbol

1. Abra el cuadro de diálogo *View Options*, pulse la opción *Vertical Tree* y haga clic en aplicar. La ventana *Organizer* cambia su modo de presentación.

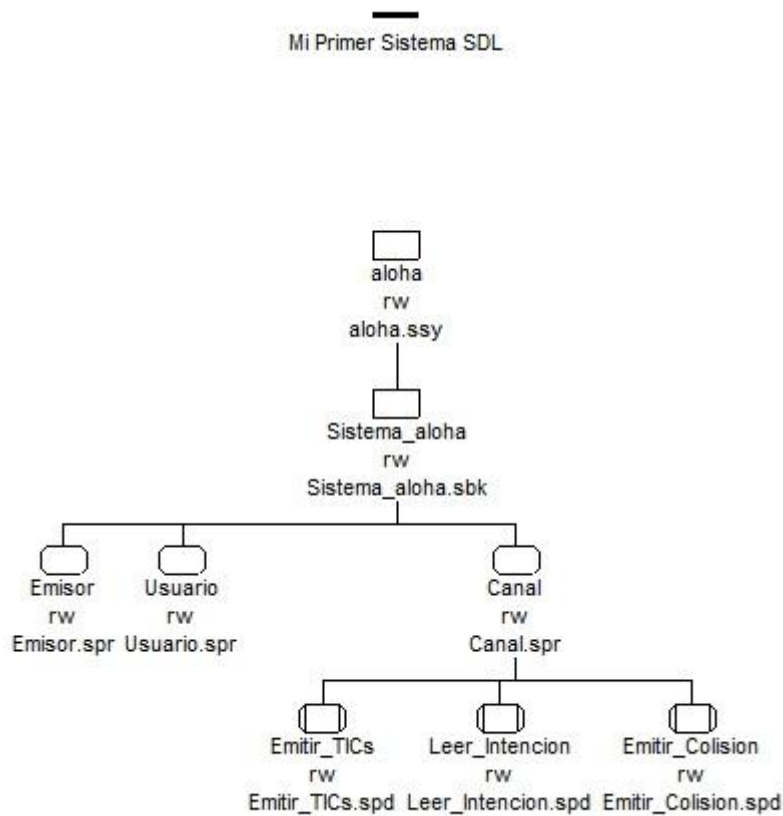


Figura 52: Estructura árbol vertical.

2. Aplique la opción *intended list* de nuevo.

4.9.3 Expandir/Contraer

Puede hacer que partes de la estructura del diagrama sean invisibles (y visibles de nuevo) mediante los comandos *Expand* (Expandir) y *Collapse* (Contraer) en el menú *View*.

Expandir está disponible solamente en los nodos contraídos (indicados por un pequeño triángulo), mientras que Contraer está disponible en nodos expandidos que tienen una subestructura.

1. Contraiga el proceso Canal. Como puede ver se añade un pequeño triángulo debajo del proceso.



Figura 53: Proceso contraído.

2. Expáñdalo de nuevo pulsando *Expand Substructure*.

4.9.4 Reorganizar diagramas

La ventana *Organizer* le permite organizar el orden de aparición de los símbolos.

Puede hacerlo ya sea con las flechas de tecla (<Arriba>, <Abajo>, <Derecha>, <Izquierda>) o con los botones de acceso rápido *Move Up* y *Move Down*.

Digamos que desea cambiar el orden de aparición de los procesos Emisor y Usuario.

1. Seleccione el proceso Emisor.
2. Pulse el botón *Move Down*, también puede hacerlo pulsando la tecla <Shift> y <Abajo>.

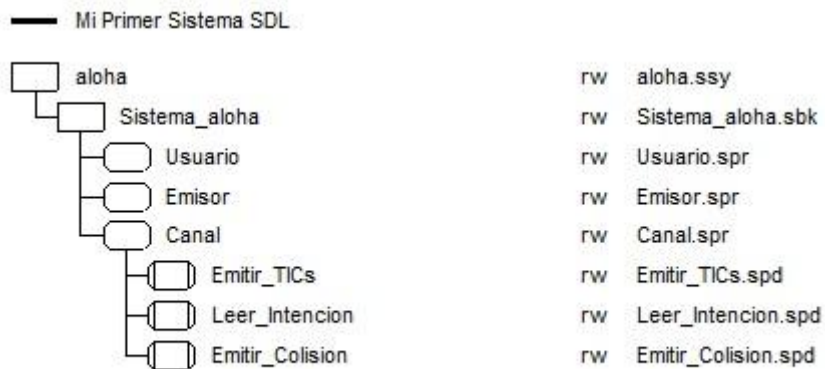


Figura 54: Reorganización de Emisor y Usuario.

3. Vuelva a cambiar el diagrama a su forma original.

4.9.5 Diagrama de páginas

1. En el cuadro de diálogo *View Options* marque la opción *Page Symbols*.
2. Aplique las opciones, el resultado es una lista en la que las páginas de SDL se hacen visibles.

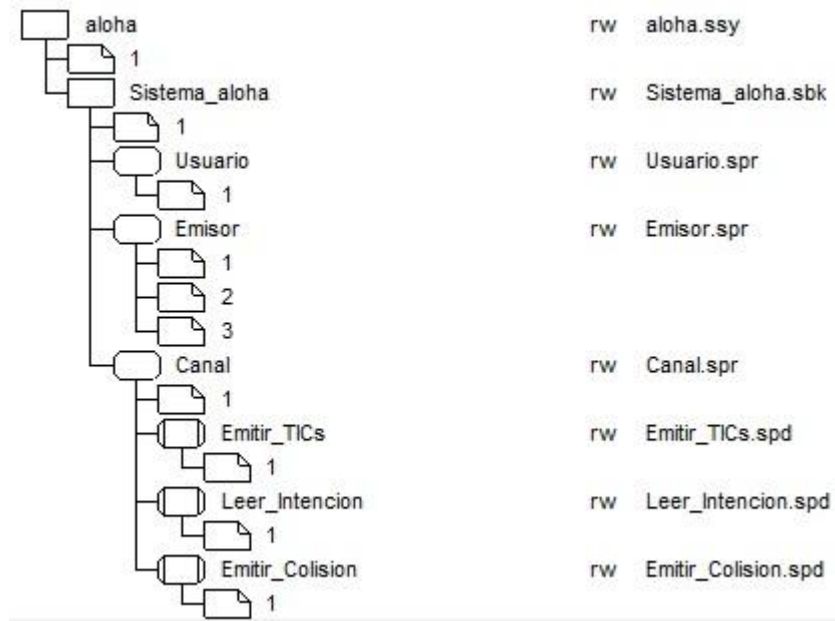


Figura 55: Páginas del diagrama.

4.9.6 Impresión del Sistema

Desde *Organizer* puede imprimir todos los diagramas que incluye el sistema con la utilización de un solo comando. También puede incluir una tabla de contenido.

1. Seleccione el diagrama de Sistema.



2. Pulse el botón de acceso rápido para impresión. Esto abre la ventana de impresión del organizador.

3. Pulse la opción *Table of contents* y pulse el botón *Print* para imprimir una impresión global de todos los diagramas SDL, incluyendo una tabla de contenido.

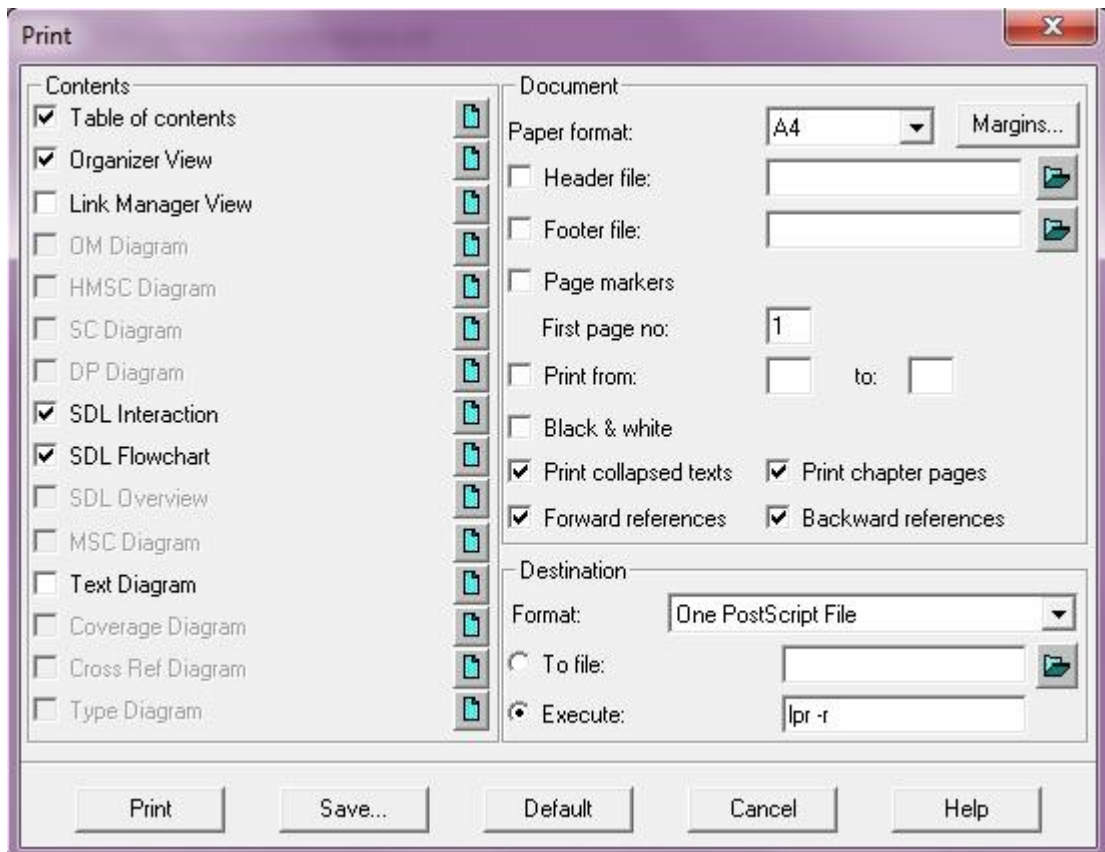


Figura 56: Incluyendo una tabla de contenidos.

Ya ha creado e impreso su primer sistema SDL utilizando la Suite SDL. Su siguiente tarea será comprobar el sistema completo con respecto a la sintaxis y la semántica en SDL.

4.10 Analizando el Sistema completo

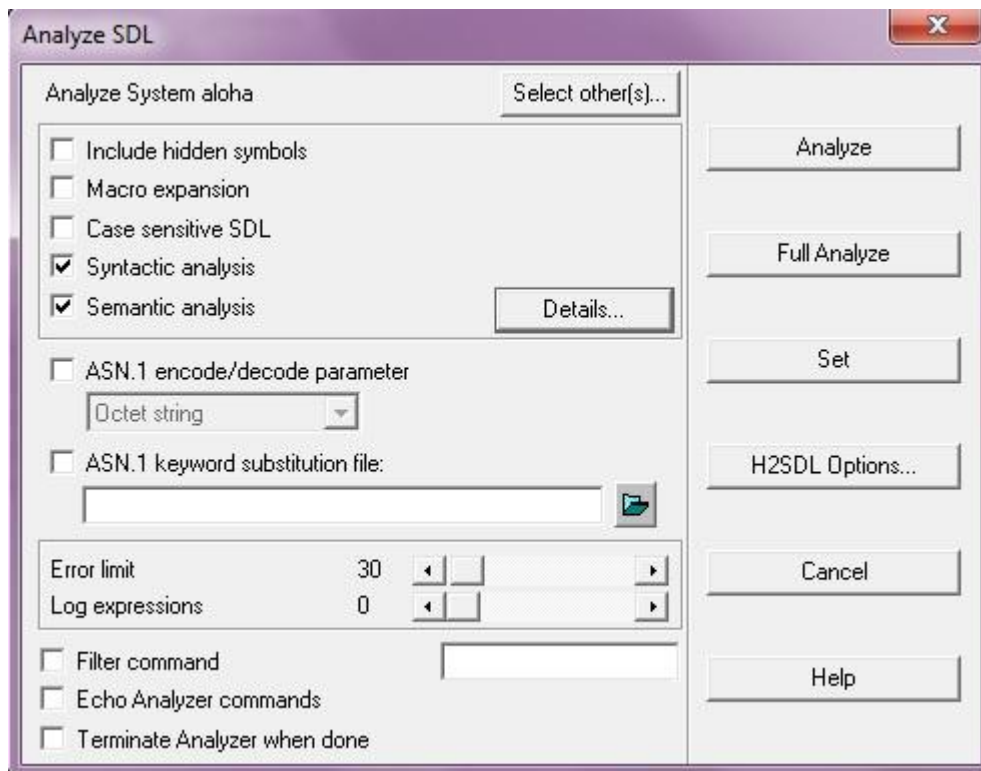
4.10.1 Qué va a aprender

- A realizar el análisis semántico y sintáctico de todo el sistema.
- A generar archivos que contienen definiciones y referencias cruzadas.

4.10.2 Habilitación de análisis semántico

Para analizar el sistema, debe permitir que el comprobador semántico actúe de la siguiente manera:

1. Seleccione el icono de diagrama de sistema.
2. Utilice el comando *Analyze* en el menú *Generate*.
3. Ajuste las opciones del analizador de acuerdo con las siguientes imágenes.



Editor y Analizador

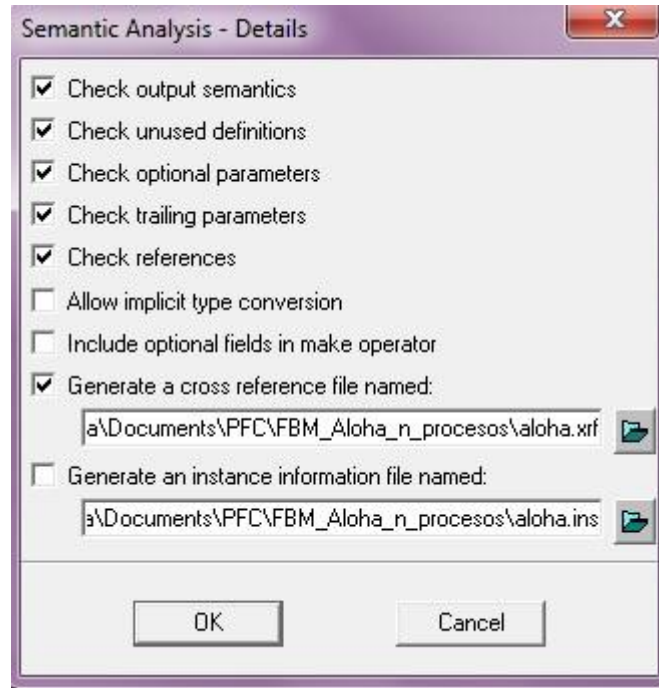


Figura 58: Detalles del analizador.

Asegúrese de que la herramienta genera un archivo con referencias cruzadas, activando la opción *Generate a cross reference file named*. Necesitará este archivo más adelante en este tutorial.

4. Pulse el botón *Analyze*(Analizar).
5. Cuando en la barra de estado de la ventana *Organizer*, aparezca que el análisis ha concluido, abra la ventana *Organizer Log* para ver los errores reportados por el analizador.
6. Si es necesario, corrija los errores y repita el procedimiento.
7. En la ventana *Organizer Log* debe aparecer la siguiente lista:

```
+ Analysis started
SDL Analyzer
Conversion of SDL to PR started
Conversion to PR completed
Syntactic analysis started
Syntactic analysis completed
Semantic analysis started
Semantic analysis completed
+ Analysis completed
|
```

Termine el ejercicio guardando todo.

Quizá quiera imprimir el sistema de nuevo.

4.11 Uso del Visor índice

En esta herramienta, vamos a practicar con el visor de índice. El visor de índice es una herramienta dedicada que ofrece una representación gráfica de las definiciones y referencias a entidades SDL disponibles en un sistema SDL. Gestiona virtualmente toda la información SDL relacionada con un sistema. Un requisito previo para este ejercicio es un archivo de referencia cruzada actualizado que contenga el último análisis semántico del sistema. Si ha cambiado cualquier diagrama SDL desde el último análisis del sistema debe volver a crear el archivo.

4.11.1 Qué va a aprender

- A iniciar el visor de índice
- Buscar definiciones
- Buscar referencias

4.11.2 Inicio del visor índice

1. Acceda al menú *Tools* desde la ventana *Organizer*. Desde el sub-menú *SDL* pulse la opción *Index Viewer*.



- Como alternativa puede pulsar el botón de acceso rápido. Se le pedirá que guarde los diagramas sin guardar. Se analiza el sistema SDL y un nuevo archivo de referencia cruzada se genera automáticamente.
2. Se abre la ventana *Index Viewer*. Comience abriendo el nuevo archivo creado **aloha.xrf** (a menos que haya utilizado el botón de acceso rápido, en cuyo caso se habrá abierto automáticamente).
 3. El visor índice lee el archivo, interpreta el contenido y lo muestra en forma gráfica.

Editor y Analizador

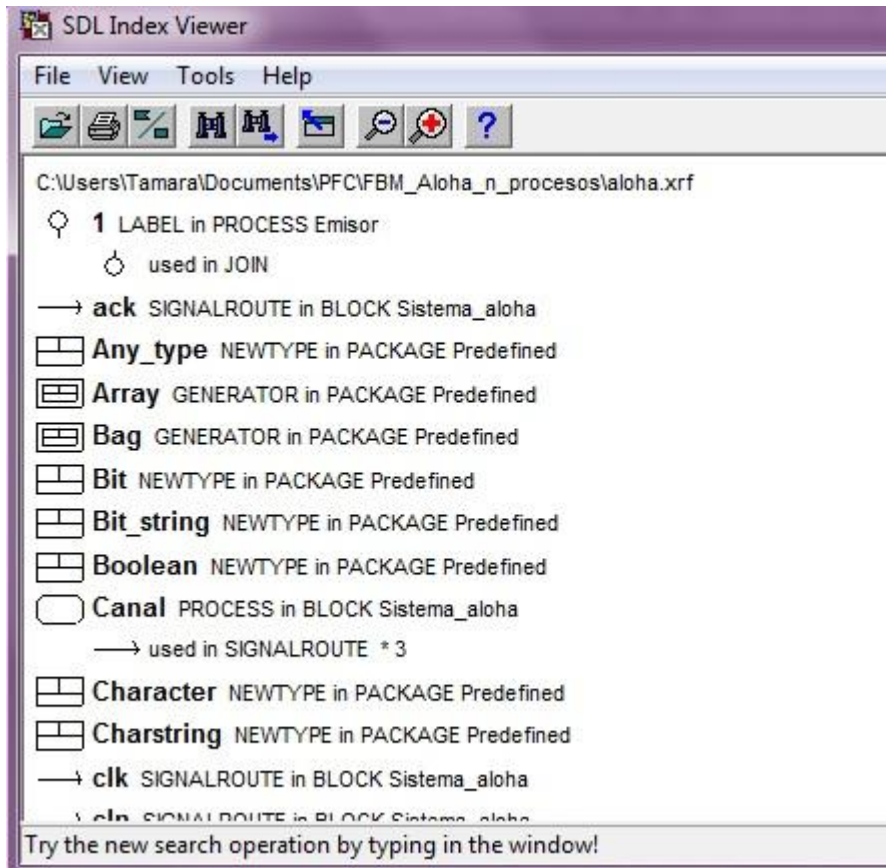


Figura 59: La ventana *Index Viewer*

En el siguiente ejercicio, va a utilizar el visor de índice para identificar la definición de una señal, así como todas las posibles situaciones en las que se le da uso.

4.11.3 Encontrar una definición

Echemos un vistazo a la definición RandInt. De forma predeterminada las definiciones se encuentran ordenadas por orden alfabético, pero no necesita ir bajando por la ventana para encontrar la definición.

Hay un botón de búsqueda rápida que se puede utilizar para encontrar cualquier texto en la ventana. Sin embargo, si desea buscar el nombre de una definición, hay una forma más rápida.

1. Simplemente comience tecleando el nombre RandInt. Cuando comience a teclear, se iniciará una búsqueda de nombres en negrita. Mientras escribe cada letra en el teclado, la barra de estado situada en la parte inferior de la ventana irá mostrando lo que se va buscando, y la ventana se desliza para mostrar la primera coincidencia de nombres. Después de unos cuantos golpes de teclado, se muestra la señal seleccionada.

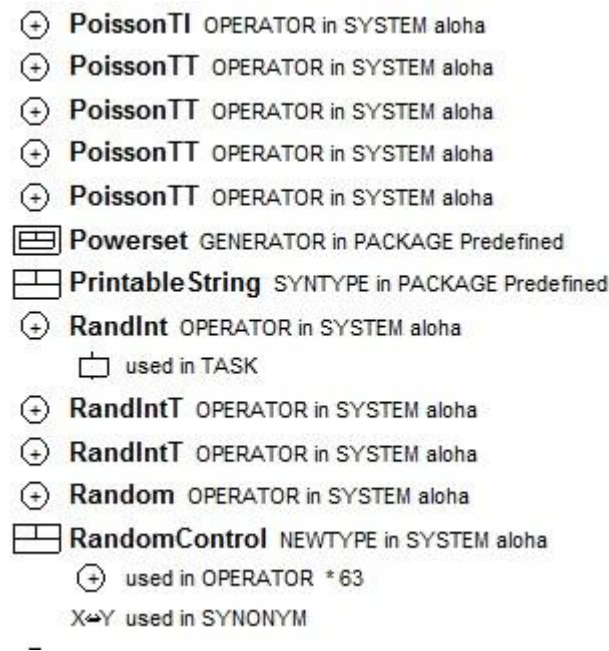


Figura 60: Encontrando la señal RandInt

2. Seleccione el icono RandInt.
3. Desde el menú *Tools*, seleccione el comando *Show Definition*.
 - También puede seleccionar dos veces el icono.

4.11.4 Encontrar Referencias

Debajo del icono RandInt en el visor de índice, se encuentran todos los usos de la señal RandInt, incluyendo los iconos de las entidades en SDL a los que la señal ha hecho referencia. En este caso en un símbolo tarea.

Para concluir el ejercicio, debe encontrar los lugares en los que la señal hace referencia a un símbolo tarea.

1. Seleccione el icono tarea.
2. En el menú *Tools* pulse la opción *Show Use*.
3. Se desplegará el diagrama SDL en que se ha utilizado la señal.

4.12 Hasta ahora...

Hasta el momento, debería haber aprendido las funciones básicas de SDL Suite; la creación, gestión, edición e impresión de diagramas SDL. También ha aprendido a comprobar la sintaxis y la semántica de los diagramas SDL. Por último se ha familiarizado con el visor de índice.

4.13 ANEXO: SISTEMA ALOHA

system aloha 1(1)

```
/*#include "random.pr"*/
```

SIGNAL
tic,
Intencion(Boolean),
collision(Boolean),
mensaje,
respuesta(Boolean);

SYNONYM
MAXRTX Integer = 15,
RANURA Duration = 1,
SEMILLA RandomControl = DefineSeed(5);

SYNONYM
N_nodos Integer = external,
PG Real = external;

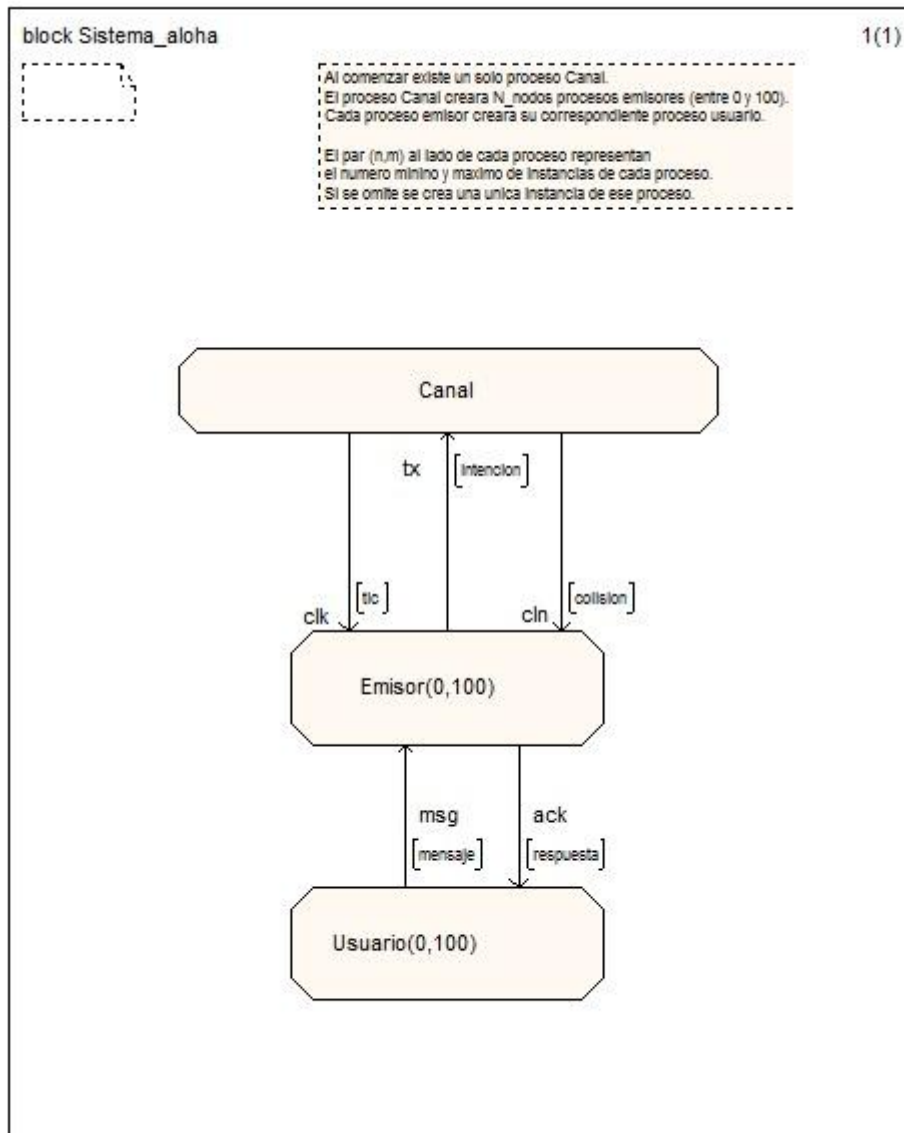
Definimos sinonimos (constantes) externamente (-external).
Estos sinonimos seran leidos, bien desde teclado, bien desde
un fichero de sinonimos (.syn).

Las definiciones externas permitiran cambiar los parametros PG y N_nodos
sin tener que compilar en cada simulacion.

Sistema_aloha

1

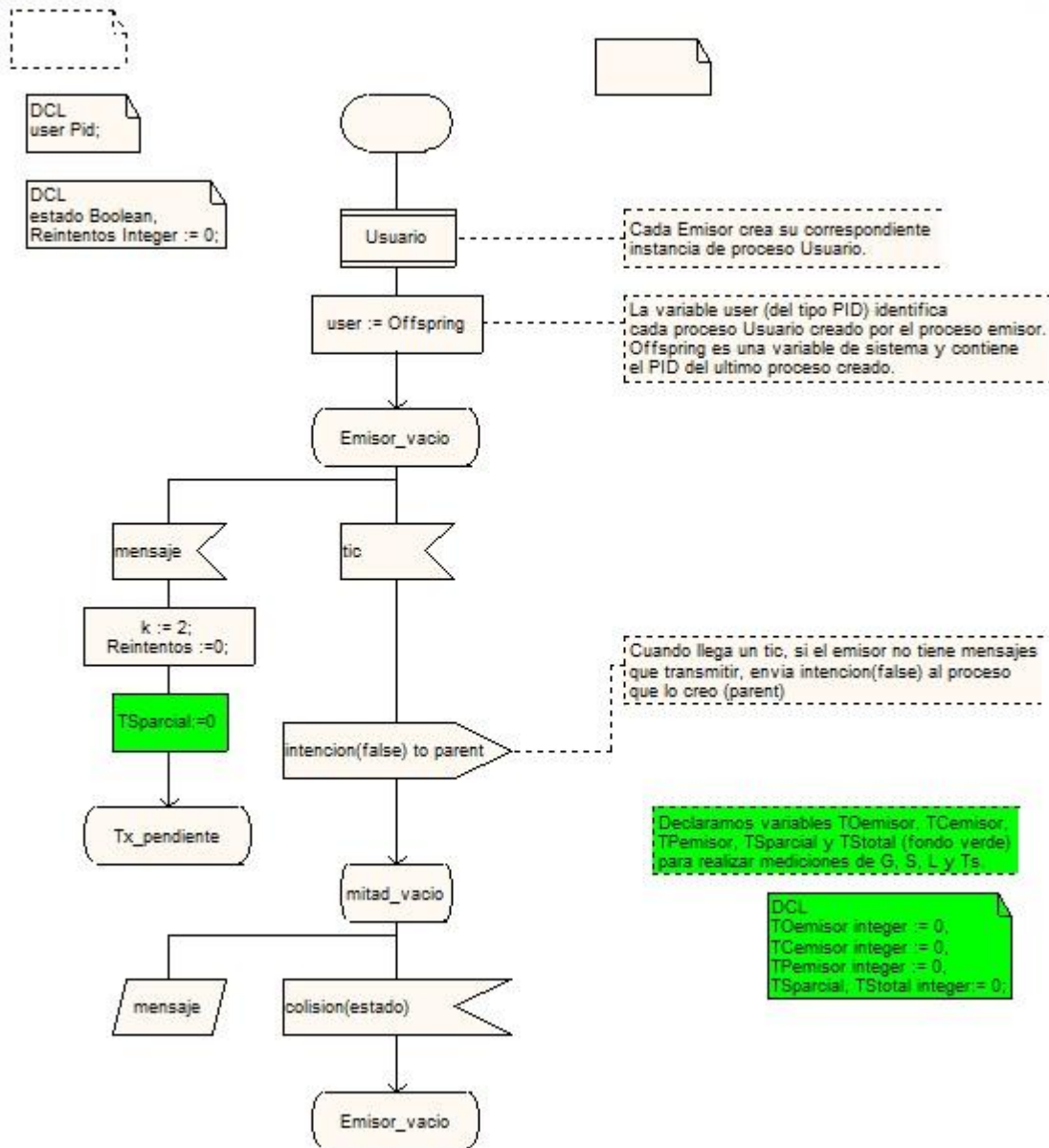
Editor y Analizador



Editor y Analizador

process Emisor

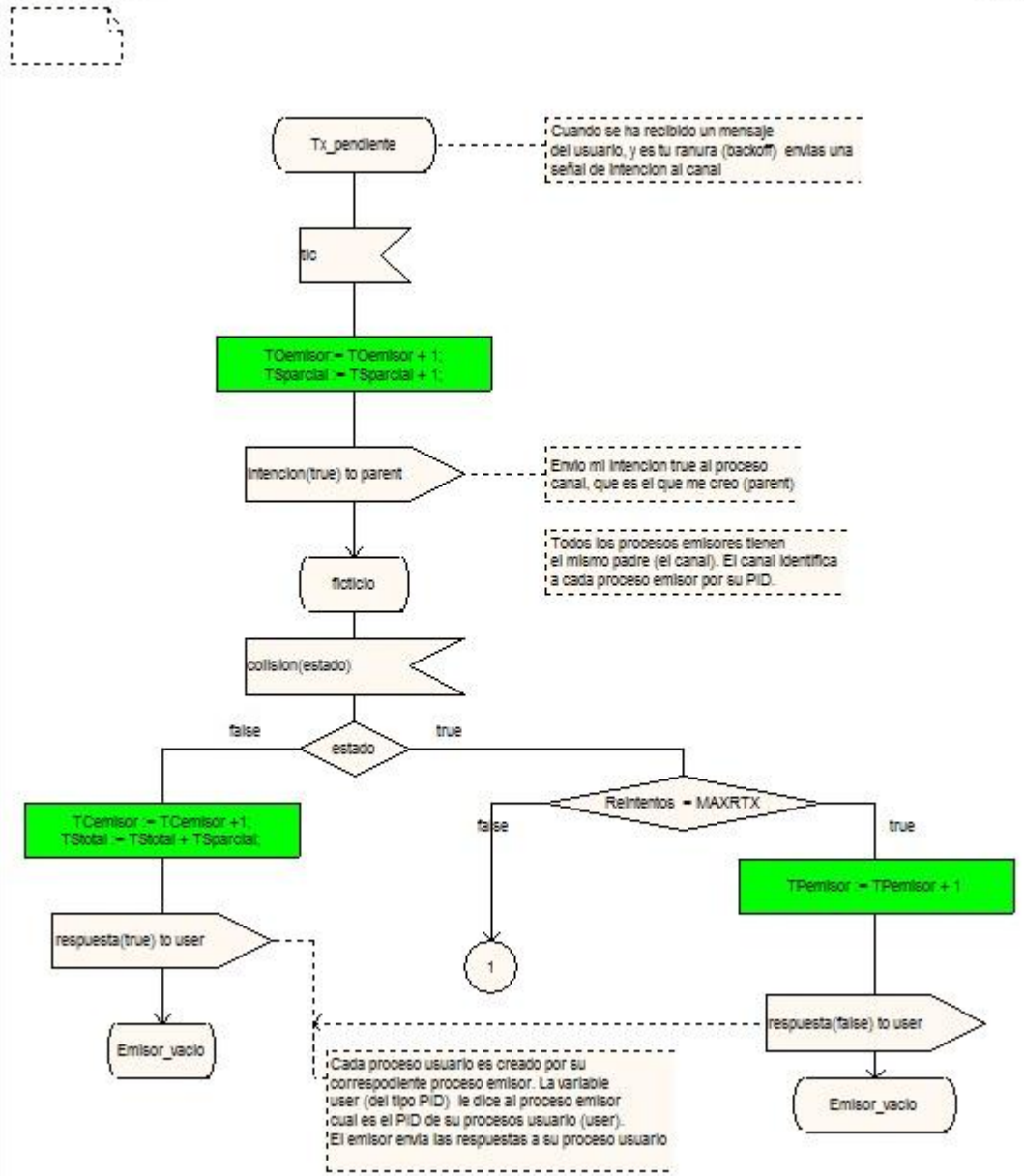
1(3)

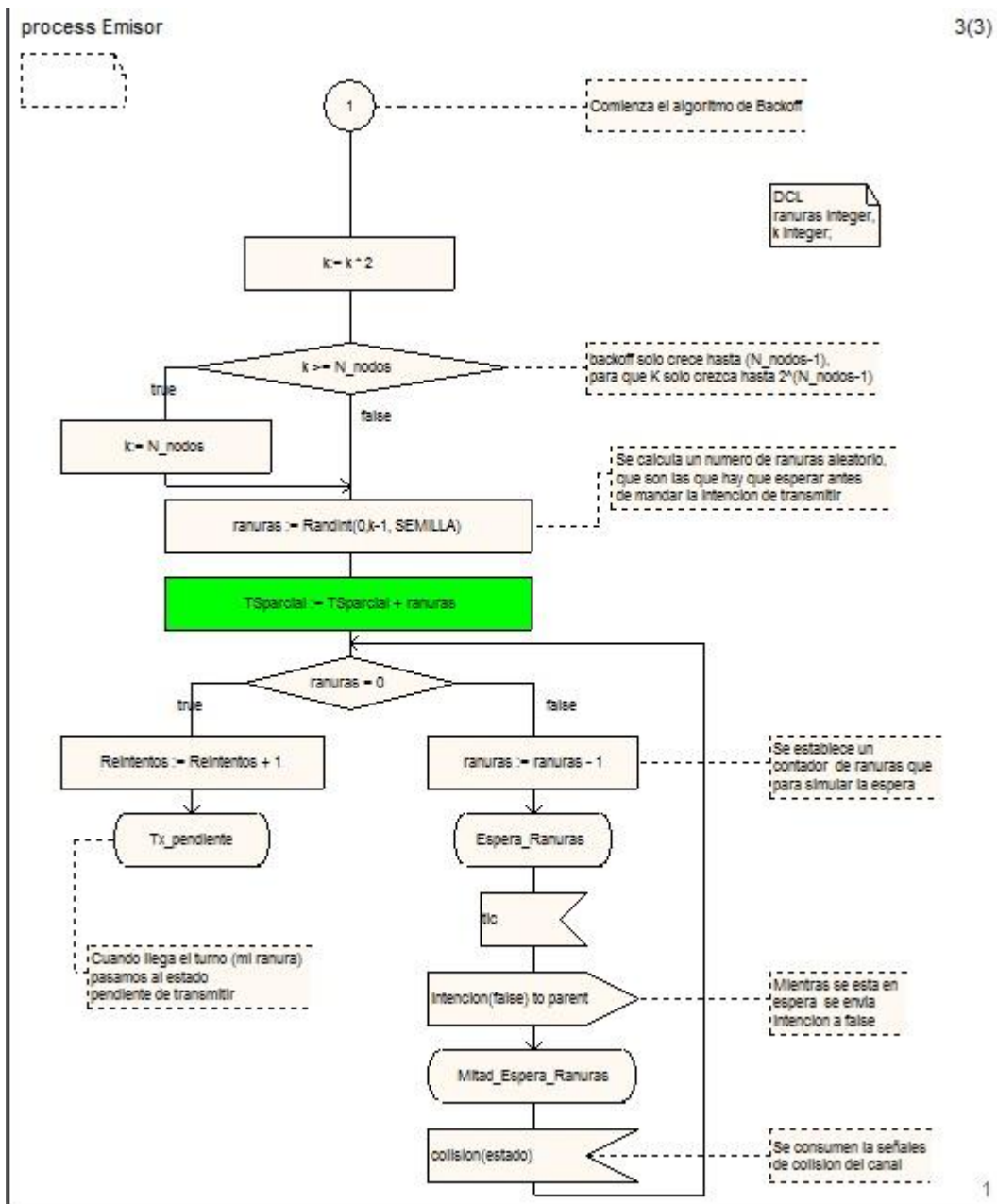


Editor y Analizador

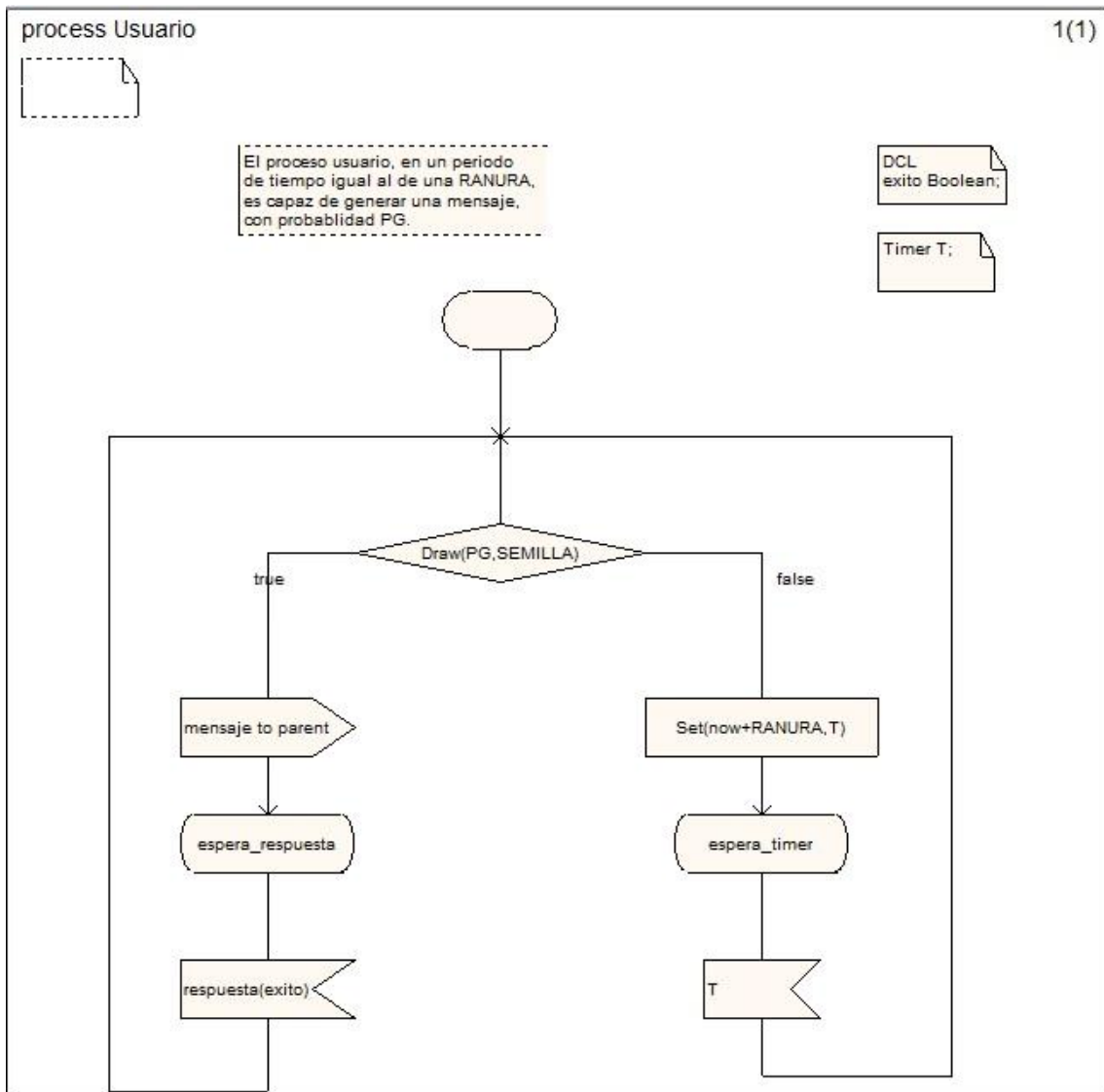
process Emisor

2(3)

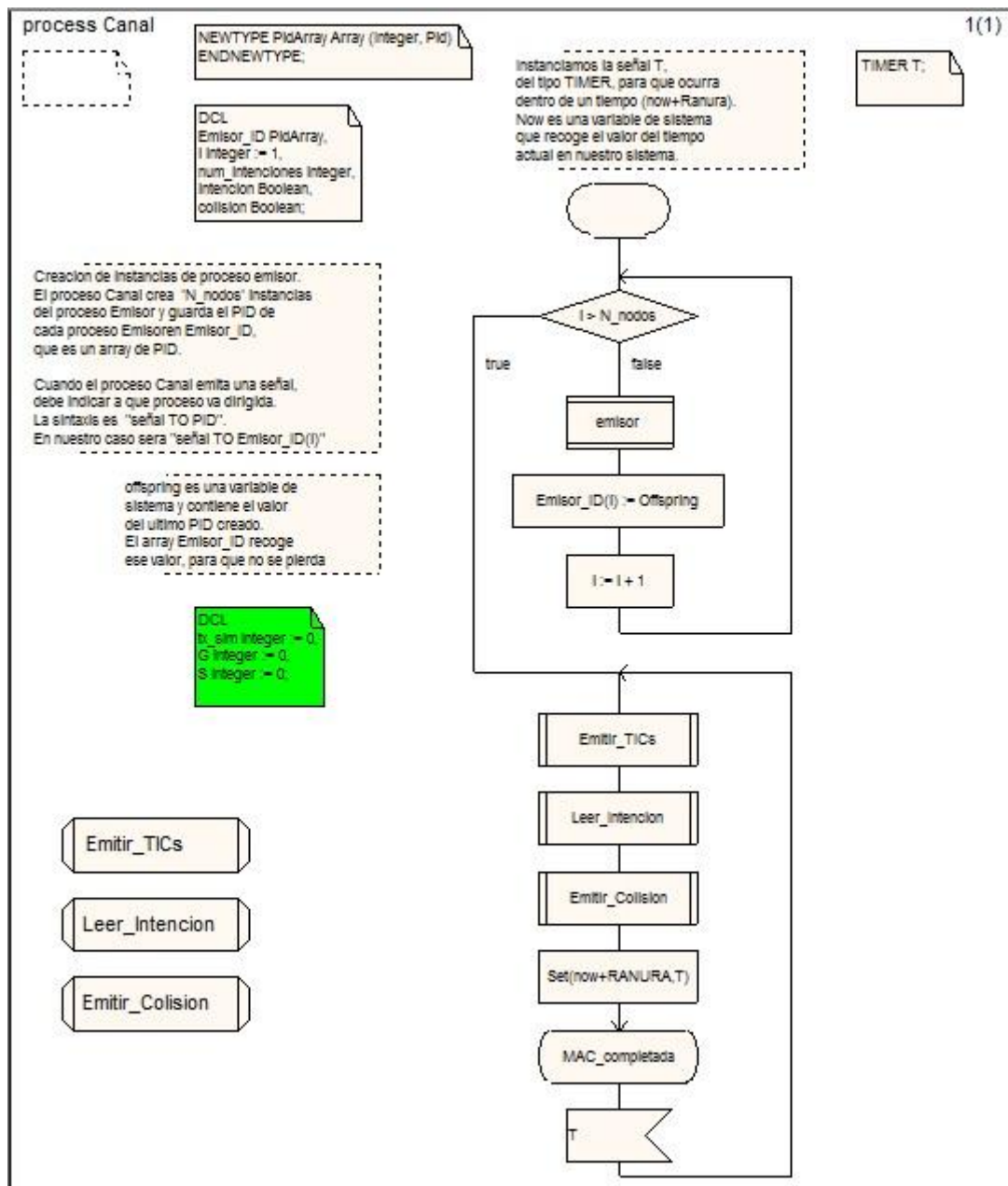




Editor y Analizador



Editor y Analizador

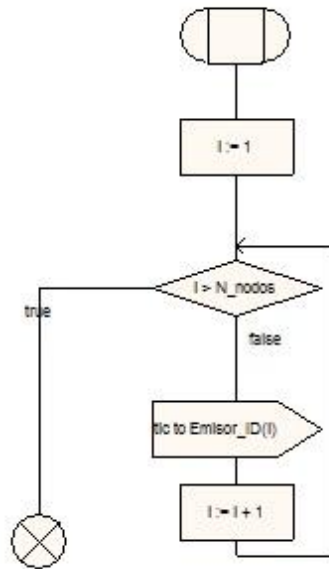


Editor y Analizador

procedure Emitir_TICs



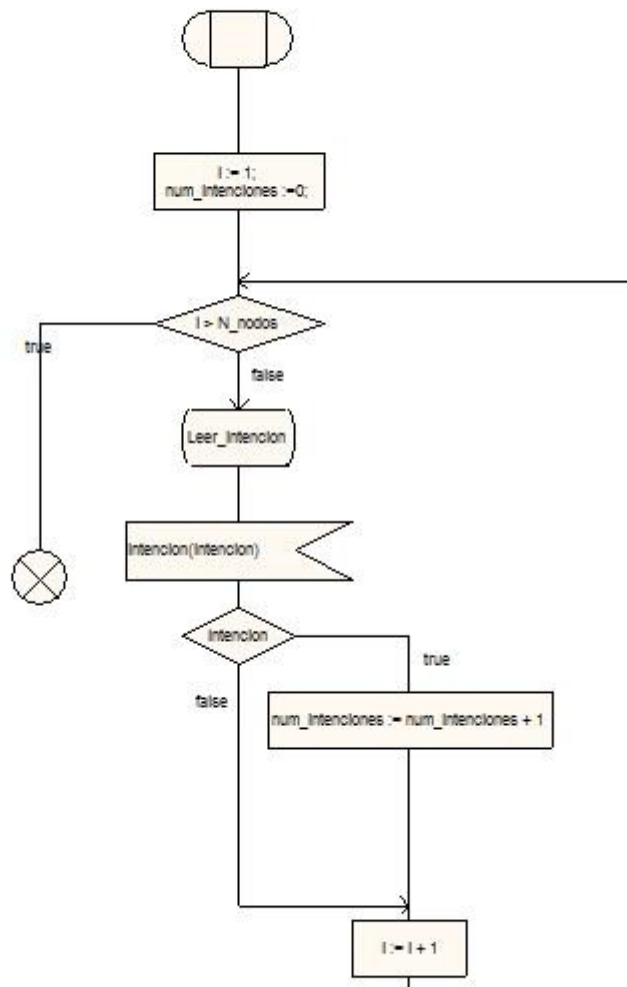
1(1)



Editor y Analizador

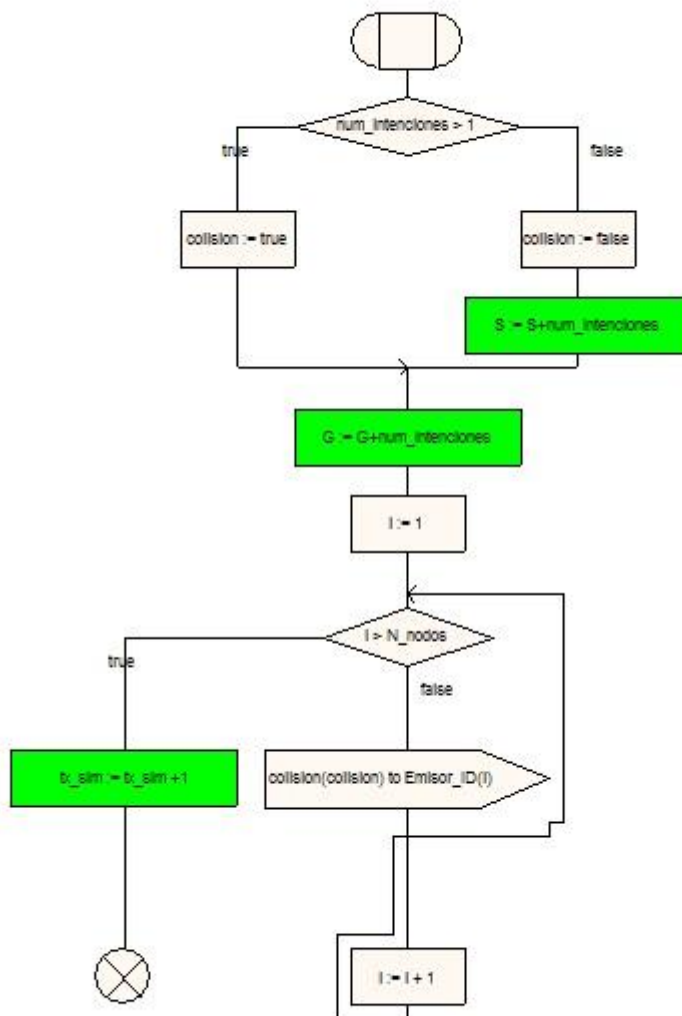
procedure Leer_Intencion

1(1)



procedure Emitir_Colision

1(1)



5.SIMULADOR SDL

El simulador SDL es la herramienta utilizada para comprobar el funcionamiento de un sistema SDL. En este tutorial realizará prácticas de primera mano en el sistema aloha.

Para ser correctamente asimilado, se presupone que ha realizado los ejercicios explicados anteriormente.

Con el fin de aprender a utilizar el simulador, debe realizar los ejercicios que se describen a continuación.

5.1 Propósito de este tutorial

El propósito de este tutorial es que se familiarice con los requisitos esenciales de funcionalidad de simulación en la Suite SDL.

Típicamente, simulación significa ejecutar el sistema bajo el control del usuario, paso a paso, establecer puntos de interrupción, examinar el sistema, procesos y variables, y el envío de señales y rastreo de la ejecución, como lo haría en un depurador, pero aplicado en el dominio SDL.

Este tutorial esta realizado como una visita a través de la Suite SDL donde deberá realizar una serie de ejercicios.

El ejemplo utilizado para la simulación es el sistema creado con anterioridad **Sistema aloha**.

5.2 Generar e iniciar un simulador

Una vez diseñado y analizado el sistema SDL completo, es posible simular el sistema, es decir, inspeccionar y comprobar su comportamiento real.

Para poder simular el **sistema aloha**, debe primero generar un simulador ejecutable y luego iniciar el simulador con una interfaz de usuario adecuada.

5.2.1 Qué va a aprender

- A generar un simulador ejecutable
- A iniciar la interfaz de usuario del simulador
- A iniciar un simulador de la interfaz de usuario

5.2.2 Generando el simulador

Para generar un programa de simulación ejecutable, haga lo siguiente:

1. Asegúrese de que selecciona el icono del diagrama del sistema en la ventana *Organizer*.
2. Seleccione el comando *Make* desde el menú *Generate*. La ventana *Make* es desplegada:

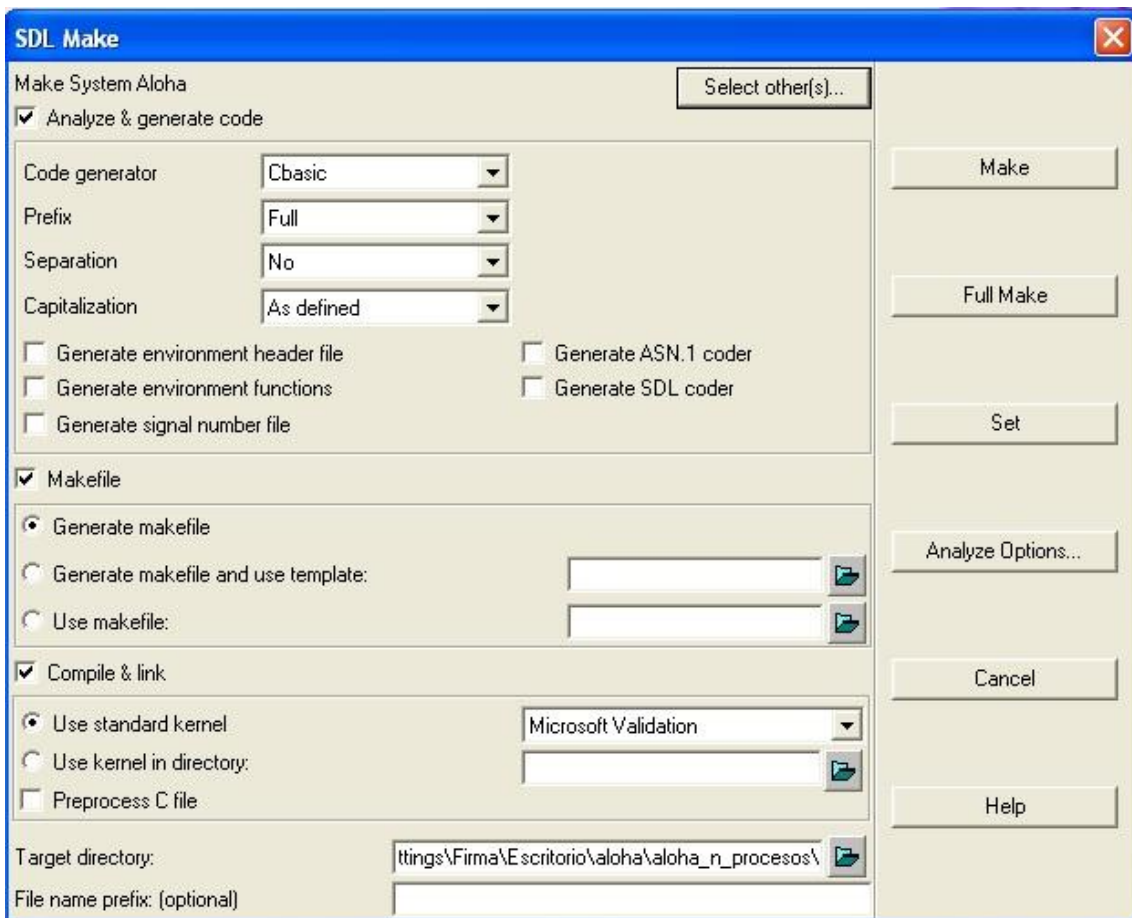


Figura 61: La ventana *Make*

Simulador SDL

3. Ajuste las opciones tal y como se muestran en la figura:
 - *Analyze & generate code* marcada.
 - *Makefile* y *Generate makefile* marcadas.
 - *Compile & link* marcada.
 - **Use standar kernel**. En este caso Microsoft Validation.
4. Pulse el botón *Make*.
5. Seleccione *Organizer Log* desde el menú *Tools*. La barra de estado de *Organizer* debe mostrar “Analyzer done” y la ventana *Organizer Log* no debe mostrar ningún error entre los mensajes “Make started” y “Make completed”.
6. Si se han producido errores, abra de nuevo la ventana *Make*, y marque el botón *Full Make*. Esta vez, no deben aparecer errores.
 - Otro problema podría ser con su compilador de C del sistema. En ese caso, si aún recibe errores, pruebe cambiando a un simulador de kernel que refleje su compilador C, y repita el proceso *Make*.

5.2.3 Inicio del Simulador

El simulador generado se almacena en un archivo llamado **aloha_smc.exe** en el directorio desde el que se inició la Suite SDL. El simulador contiene un sistema que proporciona un conjunto de comandos que pueden ser utilizados para monitorear, controlar y supervisar la ejecución del simulador.

Es posible que usted pueda ejecutar el simulador directamente desde su sistema operativo de forma inmediata, en cuyo caso tiene que introducir todos los comandos al monitor del sistema textualmente, utilizando una sencilla interfaz de línea de comandos.

La Suite SDL proporciona una interfaz gráfica fácil de usar para el simulador que se inicia desde *Organizer*.

1. Desde el menú *Tools*, seleccione el submenú *SDL* y el comando *Simulator UI*.

La ventana *Simulator UI* es desplegada:

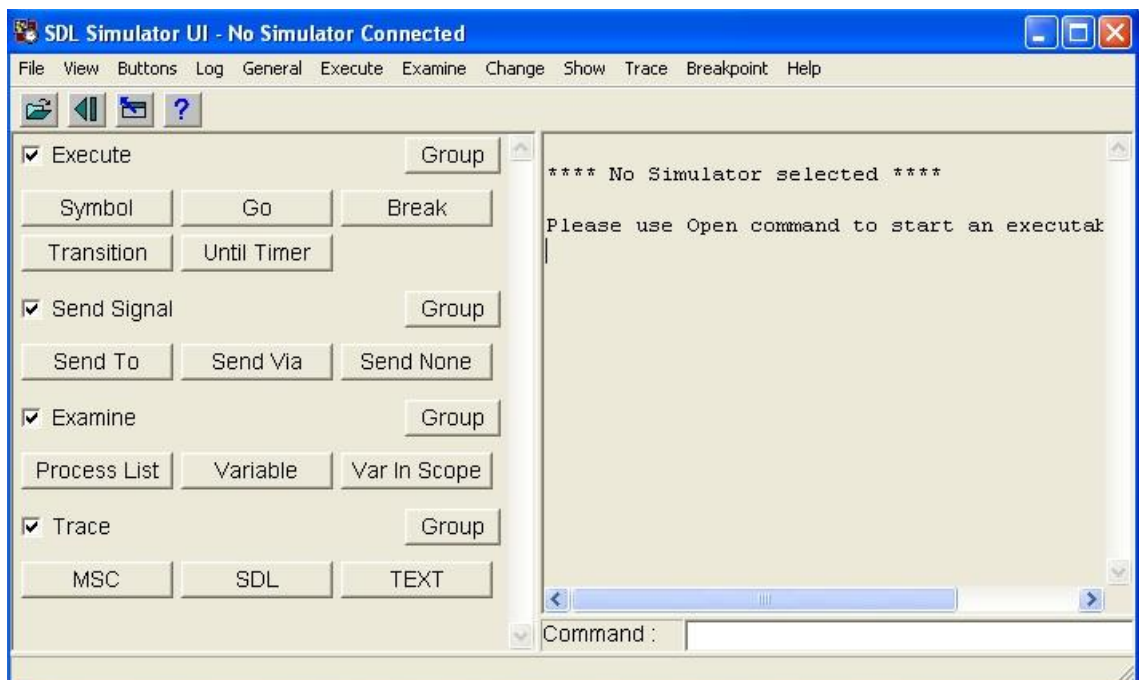


Figura 62: Ventana Principal de *Simulator UI*

El área de texto a la derecha informa que no se está ejecutando el simulador.

Simulador SDL

El área de texto muestra el área de entrada/salida del monitor, los comandos introducidos, los resultados de los comandos y los mensajes de error.



2. Para iniciar un simulador, seleccione *Open* (Abrir) desde el menú *File* o pulse el botón de acceso rápido *Open*.
3. En el cuadro de diálogo de selección de archivos, el archivo generado simulador descrito arriba debe aparecer. Selecciónelo y haga clic en OK o en *Open*.

Le pedirá que introduzca el fichero donde se encuentra el valor de las variables externas, puesto que de momento no se va a utilizar, introduzca los valores manualmente.

N_nodos = 5

PG = 0.95

4. El área de texto de Simulator UI lo recibe con un mensaje de bienvenida reconociendo que el simulador se ha iniciado:

```
Welcome to the SDL SIMULATOR. Simulating system aloha.
```

Cuando se inicia un simulador, las instancias de procesos estáticos del sistema se crean, pero sus transiciones iniciales no se ejecutan.

Simulator UI está listo para aceptar comandos.

5.3 Ejecutar transición por transición

5.3.1 Qué va a aprender

- A mostrar el siguiente símbolo que se ejecutará.
- Interpretar trazo impreso y gráfico.
- Ejecutar la siguiente transición
- Ejecutar los comandos con los botones

5.3.2 Ejecución de transiciones de inicio

En este ejercicio, vamos a ejecutar las transiciones de inicio en las instancias de proceso del sistema. Sin embargo, primero se debe definir la cantidad de información que desea imprimir durante la ejecución. El comando **Set-Trace** se utiliza para ajustar el nivel de rastreo textual.

Por ahora, introducirá comandos textualmente utilizando el campo de texto *Command*: justo debajo del área de texto. Este campo se llama línea de entrada.

1. Haga clic en la línea de entrada para colocar el cursor. Introduzca el comando **set-trace 6** y pulse <Enter>. El valor 6 especifica que quiero información completa acerca de las acciones que se realizan durante las transiciones. El comando introducido se traslada a la zona de texto y el simulador confirma el valor de rastreo, el área de texto mostrará.

```
Welcome to the SDL SIMULATOR. Simulating system aloha.  
  
Command : set-trace 6  
Default trace set to 6  
  
Command :
```

Un rastro gráfico (GR) también está disponible, lo que significa que la ejecución se traza en los diagramas SDL seleccionando el siguiente símbolo para ser ejecutado. El trazo marcado por GR está desactivado por defecto al iniciar la simulación.

Simulador SDL

2. Para habilitar el trazo GR, introduzca el comando **set-gr-trace 1**. El valor 1 especifica que se mostrará el siguiente símbolo para ser ejecutado en el Editor SDL.
3. Para que aparezca una ventana del Editor SDL, debe introducir el comando **show-next-symbol**. La ventana del Editor SDL se utilizará para el rastreo gráfico de la ejecución de simulador durante la sesión de simulación.
4. Si fuera necesario, puede cambiar y mover el tamaño de la ventana Simulator UI y Editor SDL para que ambas sean completamente visibles y encajen en la pantalla. Otro consejos son:



- Antes de redimensionar el Editor de SDL, puede ocultar el menú de símbolos y la ventana de texto mediante el uso de los botones de acceso rápido. La ventana editor no será utilizada para editar ningún diagrama, sólo para su visualización.
- Compruebe, y si es necesario, ajuste la opción del Editor de SDL *Always new window* desmarcándola. Para ello acceda al comando *Editor Options* desde el menú *View*.
- Puede reducir la anchura de la ventana Simulator UI, pero esto sólo afecta a la anchura del área de texto. Puede reducir la altura, pero debe asegurarse de que los botones de la parte izquierda aún se encuentren visibles.
- Durante la realización de este tutorial, no necesita observar la ventana *Organizer*, por lo que puede ocultarla o reducirla al mínimo mediante el uso del sistema de ventanas.

Simulador SDL

5. Ejecute el comando **Next-Transition** simplemente introduciendo **n-t** en la línea de entrada. Todos los comandos pueden abreviarse siempre y cuando su abreviatura no sea única entre todos los comandos disponibles.

- En el área de texto, la información de la transición contiene las instancias de procesos, el nombre del estado inicial, el valor del tiempo de simulación etc. Termina con *NextState*, dando el nombre de estado del proceso resultante.

```
*** TRANSITION START
*   Pid      : Canal:1
*   State    : start state
*   Now      : 0.0000
*   DECISION Value:false
*   CREATE Emisor:1
*   ASSIGN Emisor_ID... := Emisor:1
*   ASSIGN i := 2
*   DECISION Value:false
*   CREATE Emisor:2
*   ASSIGN Emisor_ID... := Emisor:2
*   ASSIGN i := 3
*   DECISION Value:false
*   CREATE Emisor:3
*   ASSIGN Emisor_ID... := Emisor:3
*   ASSIGN i := 4
*   DECISION Value:false
*   CREATE Emisor:4
*   ASSIGN Emisor_ID... := Emisor:4
*   ASSIGN i := 5
*   DECISION Value:false
*   CREATE Emisor:5
*   ASSIGN Emisor_ID... := Emisor:5
*   ASSIGN i := 6
*   DECISION Value:true
*   PROCEDURE START Emitir_TICs
*   ASSIGN i := 1
*   DECISION Value:false
*   OUTPUT of tic to Emisor:1
*   ASSIGN i := 2
*   DECISION Value:false
*   OUTPUT of tic to Emisor:2
*   ASSIGN i := 3
*   DECISION Value:false
*   OUTPUT of tic to Emisor:3
*   ASSIGN i := 4
*   DECISION Value:false
*   OUTPUT of tic to Emisor:4
*   ASSIGN i := 5
*   DECISION Value:false
*   OUTPUT of tic to Emisor:5
*   ASSIGN i := 6
*   DECISION Value:true
*   PROCEDURE RETURN Emitir_TICs
*   PROCEDURE START Leer_Intencion
*   ASSIGN i := 1
*   ASSIGN num_intenciones := 0
*   DECISION Value:false
*** NEXTSTATE Leer_intencion
```

Simulador SDL

6. Coloque y haga clic con el puntero en la línea de entrada y presione la tecla de flecha <Arriba>. El comando que ha introducido anteriormente aparece (n-t). Ejecútelos presionando <Enter>.



- Puede utilizar las teclas de flecha arriba y abajo repetidamente en la línea de entrada para mostrar los comandos introducidos anteriormente. Esta característica se conoce comúnmente como un historial de comandos. Puede editar un comando antes de su ejecución, por ejemplo, cambiando el valor de un parámetro.

Se ejecuta la siguiente transición, que como se puede apreciar es la transición inicial del proceso Emisor.

```
*** TRANSITION START
*   Pid      : Emisor:1
*   State    : start state
*   Now      : 0.0000
*   CREATE   Usuario:1
*   ASSIGN   user := Usuario:1
*** NEXTSTATE Emisor_vacio
```

5.3.3 Utilización de los botones del Simulador

Usted puede ejecutar comandos mediante la utilización de botones en lugar de introducirlos textualmente en la línea de entrada. Los botones de comandos están dispuestos en diferentes módulos en la parte izquierda del Simulador. Puede tener una “vista previa” del comando ejecutado por el botón seleccionándolo, la función del comando aparecerá en la barra de estado de la parte inferior de la ventana del Simulador.

A continuación se van a repasar los botones más interesantes dentro del Simulador.

› Dentro del menú *Execute*, existen tres botones fundamentales.



- **TRANSITION:** que ejecuta la siguiente transición dentro de la Simulación. (Es la equivalencia al comando *Next-Transition*).
- **GO:** Que inicia la simulación del Sistema
- **BREAK:** Que le permite parar la simulación cuando lo crea oportuno.

Simulador SDL

› Dentro del menú *Send Signal* se va a destacar la opción *Send Via*:



- SEND VIA: Este botón ejecuta el comando *Output-Via*, como se puede apreciar en el área de texto. Al pulsarlo, se abrirá un cuadro de diálogo pidiendo el valor del primer parámetro, el nombre de la señal. La lista que aparece contiene todas las señales posibles para enviar desde el entorno.

5.4 Como ver el estado interno

En este ejercicio, vamos a presentar algunos de los comandos disponibles para ver el estado interno del sistema. Con la interfaz gráfica de usuario, también es posible ver de forma continua el estado interno sin tener que ejecutar comandos de forma manual.

En el ejercicio anterior, aprendió a interpretar las huellas disponibles. Por el momento no se volverá a centrar en esos detalles, a menos que sean necesarios para señalar algún aspecto importante.

5.4.1 Qué va a aprender

- A reiniciar el simulador sin salir de Simulator UI
- Utilizar las ventanas *Commands* y *Watch*
- Ver e interpretar la cola de listas de procesos
- Ver el puerto de señal de entrada de un proceso
- Ver los valores de variables
- Examinar las instancias de proceso
- Ver temporizadores activos

5.4.2 Reiniciar el simulador

Antes de continuar, es necesario reiniciar la simulación desde el principio y establecer el nivel de seguimiento.

1. Seleccione el comando *Restart* desde el menú *File*.
2. Un cuadro de diálogo le informará que la simulación actual ha finalizado. Confírmelo pulsando OK. El área de texto se borra y el Simulador se restablece.
3. Ajuste el nivel de seguimiento a 6 y el nivel de seguimiento GR a 1, tal y como se encontraba anteriormente. La manera más fácil de hacerlo es utilizar la flecha <Arriba> y <Abajo> en la línea de entrada de texto para encontrar los comandos anteriores **Set-Trace 6** y **Set-GR-Trace 1**. Haga clic en <Enter>.
 - El simulador UI recuerda todos los últimos comandos escritos en una misma sesión, incluso si se ha iniciado una nueva simulación.

5.4.3 Viendo procesos y colas de señal

Para ver el estado interno de forma continua, utilice la ventana *Commands*. Esta ventana muestra si las preferencias se configuran adecuadamente, el proceso listo en cola y la lista de todos los procesos.

Esta información se muestra en “módulos” separados en la ventana *Command* (List-Ready-Queue y List-Process). Estos módulos son similares a los módulos de botones en la ventana principal.

Simulador SDL

1. Seleccione *Command Window* desde el menú *View* para abrir la ventana de comandos. Cambie el tamaño de la ventana para que los dos módulos de comandos sean visibles. Mueva la ventana, para poder ver la ventana del Editor SDL y también la ventana principal.

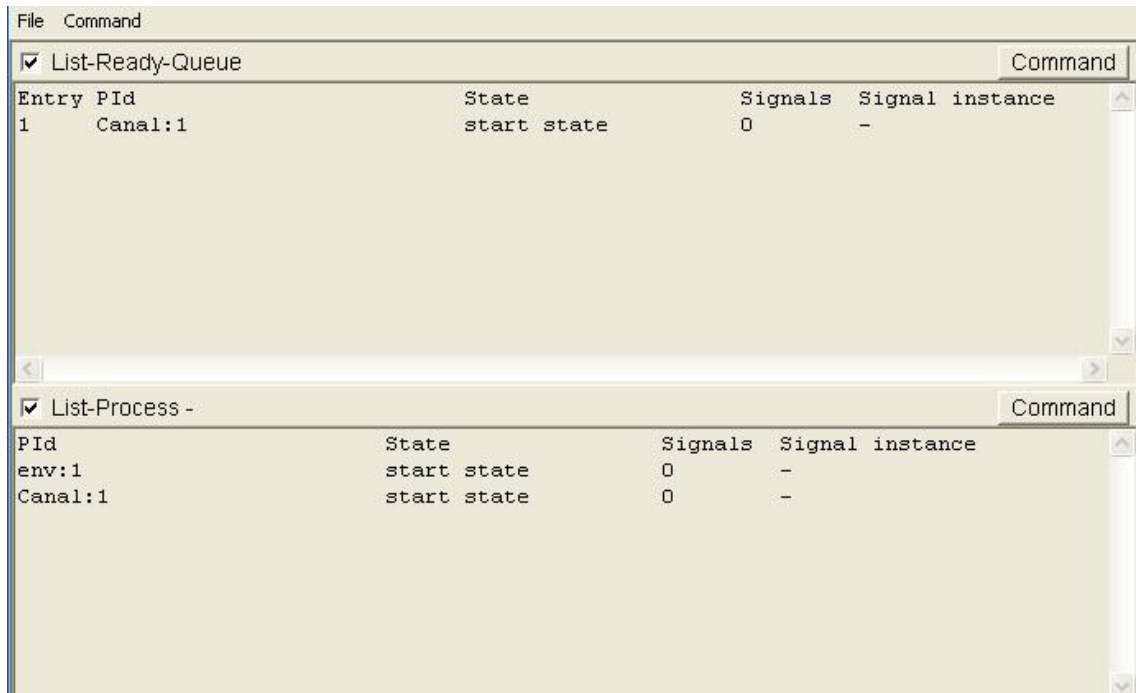


Figura 63: La ventana Command

Nota:

Si la ventana *Command* muestra los comandos o no depende de la preferencia. Si la ventana no muestra estos comandos al arrancar, puede agregar los comandos con la opción *Add Command* desde el menú *Command* y especificar los comandos a agregar.

Los módulos de comando muestran la siguiente información:

- List-Queue-Ready muestra la lista de procesos en cola, es decir, una lista ordenada de todos los procesos que están listos para ejecutar una transición. La lista contiene un número de entrada (la posición en la cola), la instancia de proceso, su estado actual, el número de las señales en su puerto de entrada y el nombre de la señal que causará la siguiente transición.
 - List-Process muestra una lista de todos los procesos activos en el sistema. Muestra la misma información que List-Queue-Ready. En esta etapa, las dos listas son idénticas.
2. Ejecute la siguiente transición y observe los cambios en la ventana *Command*. El proceso Canal es eliminado de la lista Ready-Queue. Lista mostrará ahora los 5 procesos emisores que se encuentran en cola.
 3. También puede imprimir una lista de todas las señales en el puerto de entrada del proceso que se va a ejecutar. El comando necesario no se encuentra disponible a través de un botón de comandos. En su lugar, busque el menú *Examine* en la barra de menús y seleccione el comando *Input Port*. Para cada señal se imprime un número de entrada (la posición en la cola), el nombre de la señal y el remitente.

5.4.4 Visualización de variables e instancias de procesos

Aparte de la ventana de comandos, también puede monitorear continuamente valores de variables mediante el uso de la ventana *Watch*. Se va a controlar la variable **intencion** para ver cómo cambia.

1. Abra la ventana *Watch* desde el menú *View*. Si es necesario, muévala de modo que también se pueda ver la ventana *Command*.
2. En la ventana *Watch*, seleccione *Add* desde el menú *Watch* para agregar una variable a la lista de variables a mostrar.
3. En la ventana de diálogo, especifique tanto el proceso (entre paréntesis), como el nombre de la variable. En este caso (**Canal**) **intencion**.



Figura 64: Añadiendo una variable a visualizar

4. El valor *false* debe aparecer en la ventana Watch:



Figura 65: Añadiendo intencion a la ventana *Watch*

Si es necesario, cambie el tamaño de la ventana para que el valor sea visible.

5. Siga ejecutando transiciones y compruebe los cambios que se producen en la variable **intencion** según se avanza en el sistema aloha.

5.4.5 Otras opciones de visualización

Hay un número de comandos de visualización disponibles en el menú Examine. Puede listar los temporizadores activos en el sistema, comprobar parámetros de señal y temporizador de instancias etc. Va a concluir el ejercicio demostrando que el sistema no está en reposo, aunque la cola ahora esté vacía.

1. Compruebe que el temporizador T sigue encontrándose activo seleccionando *Timer List* desde el menú Explore. Se imprime el nombre del temporizador, la instancia de proceso correspondiente y el tiempo de caducidad.
2. Ejecute la siguiente transición. Trate de examinar la instancia del temporizador mediante el comando **Examine-Timer-Instance** (introduzca en la entrada de texto **ex-tim-ins**). Le informará de que la cola del Temporizador está vacía, es decir T ya no está activo.

5.5 Usando diferentes valores de rastreo

La cantidad de información de rastreo impresa durante las transiciones es fijada por el comando Set-Trace. Hasta ahora, ha utilizado este comando para establecer el valor de rastreo a 6. Cuanto mayor sea el valor de rastreo, más información se imprime.

También puede definir valores de rastreo para diferentes partes del sistema. De esta forma, bloques, tipos de proceso, instancias de proceso etc, pueden tener distintos valores de rastreo. Si un proceso no tiene un valor de rastreo definido, se utiliza el valor del bloque que lo contiene. Si el bloque no tiene un valor definido, se utiliza el valor usado de la siguiente estructura. El sistema siempre tiene un nivel de rastreo definido, que es 4.

En este ejercicio, utilizará la información de rastreo para todo el sistema, excluyendo al proceso Usuario. Para ello establecerá un valor de 0 para el proceso Usuario.

El valor de rastreo GR permanecerá a 1 durante todo el ejercicio.

5.5.1 Qué va a aprender

- A establecer valores de seguimiento para diferentes partes del sistema
- A listar los valores de rastreo actuales
- Ejecutar sólo el siguiente símbolo SDL
- Ejecutar una secuencia de transiciones antes de que imprima su rastro

5.5.2 Configurando valores de rastreo

1. En primer lugar, restaure el simulador.
2. Localice el menú *Trace* y seleccione *Text Level: Set*. En la ventana de diálogo seleccione *system aloha* y establezca el valor de rastreo a 6.

Simulador SDL

3. Realice la misma operación pero esta vez con proceso Usuario y establezca el valor de rastreo a 0. Note que los valores de 0 a 6 van acompañados de su respectiva explicación.

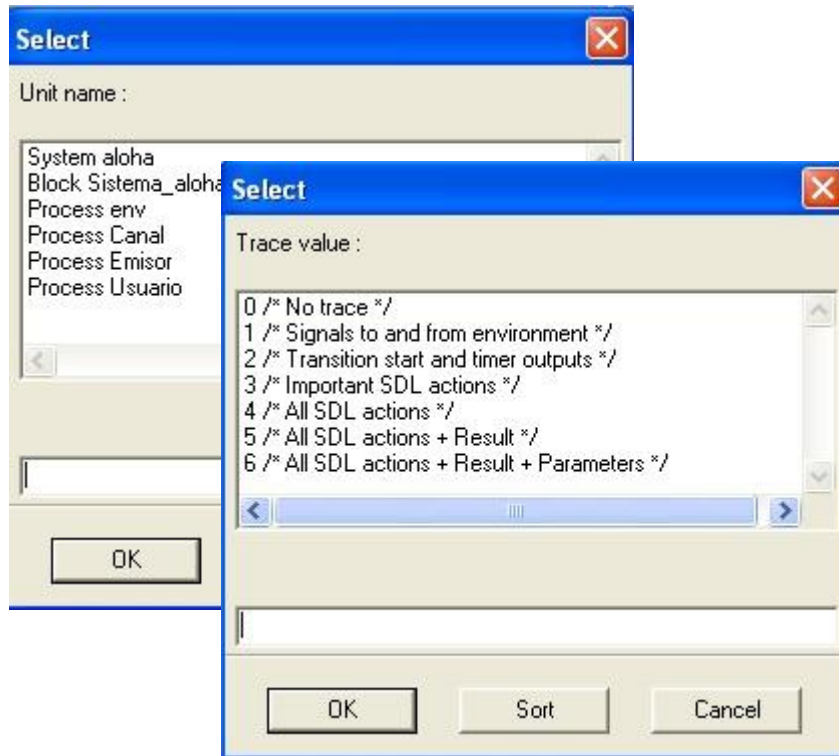


Figura 66: Estableciendo valor de rastreo a 0

4. Compruebe que ha establecido los valores de rastreo correctos utilizando las opciones del menú *Text Level: Show* y *SDL Level: Show*.

```
Command : List-Trace-Values
Default   : 4 = All SDL actions
System aloha : 6 = All SDL actions + Result + Paramete
Process Usuario : 0 = No trace

Command : List-GR-Trace-Values
Default   0 = GR trace off
```

5.5.3 Ejecución símbolo a símbolo

Para poder ver con claridad que el proceso Usuario no aparecerá en el área de texto, seguirá la ejecución en pasos más pequeños que una transición completa: ejecutando un símbolo SDL cada vez. Para ello utilizará el comando *Step-Symbol*.

1. Ejecute el símbolo de inicio del proceso Canal haciendo clic en el botón *Symbole* (perteneciente al módulo *Execute*). Tenga en cuenta que la información no incluirá la relativa al estado siguiente, ya que dicho símbolo aún no ha sido ejecutado.

```
*** TRANSITION START
*   Pid    : Canal:1
*   State  : start state
*   Now    : 0.0000
```

2. Ejecute las siguientes 5 transiciones hasta completar las 5 transiciones del proceso Emisor. Pulse el botón *Symbol* para llegar al primero símbolo perteneciente al proceso Usuario. Compruebe que ninguna información relativa al mismo es impresa en la ventana de texto.

```
*** TRANSITION START
*   Pid    : Emisor:5
*   State  : start state
*   Now    : 0.0000
*   CREATE Usuario:5
*   ASSIGN user := Usuario:5
*** NEXTSTATE Emisor_vacio
Command : Step-Symbol
```

3. Continúe ejecutando símbolos hasta volver al proceso Emisor. Siga ejecutando símbolos hasta volver a entrar al proceso Usuario.
4. Cuando entre al proceso Usuario, puede continuar ejecutando símbolo a símbolo, o puede ejecutar la transición completa usando el botón *Transition*. No se imprimirá información en la ventana de texto.

5.5.4 Ocultando transiciones poco interesantes

Si desea seguir ejecutando transición a transición, se seguirá imprimiendo la información relativa a los procesos Canal e Usuario, pero también debe seguir ejecutando manualmente las transiciones “silenciosas” del proceso Usuario. Para evitar esto, puede utilizarse el comando *Next-Visible-Transition*. Este comando ejecuta una secuencia de transiciones y se detiene después de haber alcanzado un proceso cuyo valor de rastreo es mayor de 0. De esta forma, las transiciones poco interesantes del sistema permanecen ocultas.

1. Ejecute el comando eligiendo *Until Trace* en el menú *Execute*. La ejecución no se detiene hasta que se entra de nuevo en el proceso Emisor. La información impresa pertenece a la última transición ejecutada.
2. Ejecute el comando varias veces más. Puede comprobar que la información impresa pertenece tanto a los procesos Canal como Emisor.
3. Pulsando *Prev Symbol* en el menú *Show* puede seleccionar el último símbolo ejecutado.

5.6 Observando el comportamiento del sistema

5.6.1 Qué va a aprender

- A usar el servicio de registro de señales
- Añadir botones de comandos en la interfaz de usuario del Simulador
- Ejecutar transiciones hasta cierto un cierto valor de tiempo

5.6.2 Configuración de seguimiento y registro de la señal

Durante este ejercicio se verá el comportamiento del sistema. Para lograrlo, se va a configurar el rastreo del sistema a 1. Esto significa que usted solo verá señales enviadas al entorno y ninguna de las acciones realizadas durante transiciones. Con el fin de registrar este comportamiento en un archivo, se utilizará el servicio de registro de señales.

1. Como es habitual, restaure el Simulador.
2. Establezca el nivel de rastreo del sistema a 1.
3. Para registrar las señales enviadas desde y hacia el entorno, introduzca el comando **signal-log** en la línea de entrada de texto y presione <Enter>. (Este comando no tiene ningún botón ni opción de menú asociado).

El comando *Signal-Log* toma dos parámetros, que serán solicitados a continuación en ventanas de diálogo. El primer parámetro es un nombre de unidad. Todas las señales enviadas a, desde o a través de la unidad especificada se registrará en el archivo.

4. Seleccione una de las unidades de la lista, en este caso el proceso Emisor, y pulse OK.

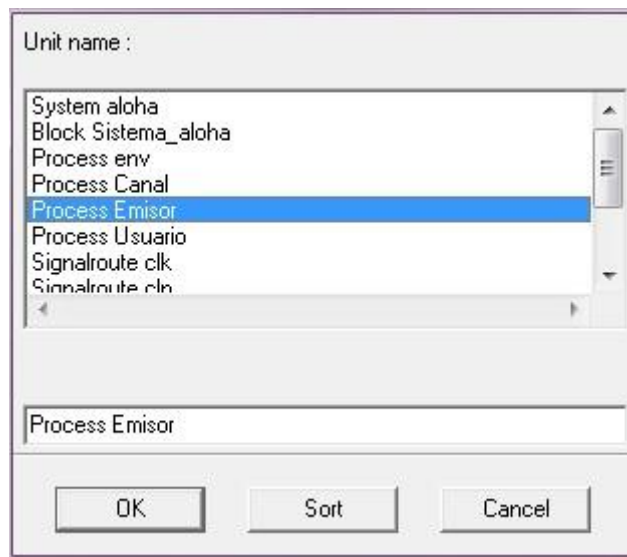


Figura 67: Especificando proceso Emisor

El segundo parámetro es el nombre de un nombre de archivo en que será escrita la información sobre las señales. Un cuadro de diálogo se abrirá a continuación.

5. En el campo Nombre, introduzca el nombre signal.log y pulse OK.

5.6.3 Añadir botones para comandos comunes

En la interfaz de usuario del Simulador, puede añadir botones a los distintos módulos para que representen comandos que usa normalmente.

1. Por ejemplo, dentro del módulo *Send Signal*, puede añadir un botón que ejecute el envío de señales, simplemente accediendo a *Group* y después *Add*.



Figura 68: Añadiendo nuevo botón a un módulo

2. En el cuadro de diálogo, introduzca el nombre de la señal, así como su definición y pulse Aplicar. El nuevo botón aparecerá en el módulo, y la ventana de diálogo estará preparada para la definición de otro botón.



Figura 69: Añadiendo un botón

3. Si lo desea, puede añadir botones en los siguientes módulos de la interfaz de Usuario.

5.6.4 Simulación del Sistema y comprobación del fichero signal.log

Se encuentra preparado para simular el sistema. Para ello utilizará el comando **Proceed-Until** para ejecutar transiciones hasta un cierto punto en el tiempo.

1. Ejecute transiciones hasta que el tiempo sea 10 seleccionando *Until Time* desde el menú *Execute*. Introduzca el valor 10 en el cuadro de diálogo. Esto ejecutará el comando **Proceed-Until 10**. Se ejecutarán todas las transiciones hasta el momento en el que el tiempo de simulación sea igual al valor del tiempo especificado.

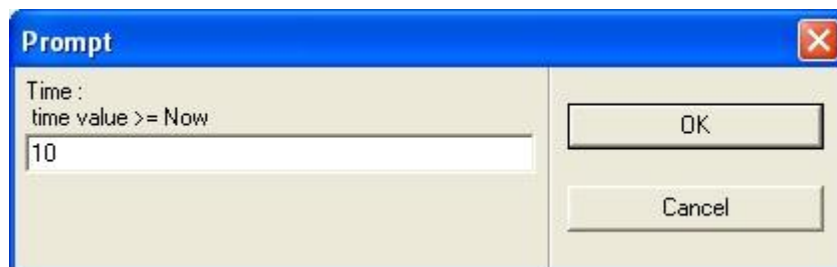


Figura 70: Proceed-Until 10

2. Comience la simulación pulsando el botón GO. El sistema simulará hasta llegar al tiempo establecido.
3. Una vez realizado el paso anterior, salga de la simulación pulsando *Stop Sim* en el menú *Execute*. Este paso es necesario para terminar el registro de la señal. El propio Simulador no es cerrado por este comando.
4. Examine el archivo signal.log desde fuera del Simulador. El archivo contiene una especificación de todas las señales enviadas desde y hacia el proceso Emisor.

Simulador SDL

```
Signal log for system aloha with unit Proces:
0.0000 tic from Canal:1 to Emisor:1
0.0000 tic from Canal:1 to Emisor:2
0.0000 tic from Canal:1 to Emisor:3
0.0000 tic from Canal:1 to Emisor:4
0.0000 tic from Canal:1 to Emisor:5
0.0000 mensaje from Usuario:1 to Emisor:1
0.0000 intencion from Emisor:1 to Canal:1
Parameter(s) : false
0.0000 mensaje from Usuario:2 to Emisor:2
0.0000 intencion from Emisor:2 to Canal:1
Parameter(s) : false
0.0000 mensaje from Usuario:3 to Emisor:3
0.0000 intencion from Emisor:3 to Canal:1
Parameter(s) : false
0.0000 mensaje from Usuario:4 to Emisor:4
0.0000 intencion from Emisor:4 to Canal:1
Parameter(s) : false
0.0000 mensaje from Usuario:5 to Emisor:5
0.0000 intencion from Emisor:5 to Canal:1
Parameter(s) : false
0.0000 colision from Canal:1 to Emisor:1
Parameter(s) : false
0.0000 colision from Canal:1 to Emisor:2
Parameter(s) : false
0.0000 colision from Canal:1 to Emisor:3
Parameter(s) : false
0.0000 colision from Canal:1 to Emisor:4
Parameter(s) : false
0.0000 colision from Canal:1 to Emisor:5
Parameter(s) : false
1.0000 tic from Canal:1 to Emisor:1
1.0000 tic from Canal:1 to Emisor:2
1.0000 tic from Canal:1 to Emisor:3
1.0000 tic from Canal:1 to Emisor:4
1.0000 tic from Canal:1 to Emisor:5
1.0000 intencion from Emisor:1 to Canal:1
Parameter(s) : true
```

Figura 71: Archivo signal.log

5.7 Estableciendo un punto de interrupción

La aplicación del Simulador llevada a cabo en este ejercicio consistirá en establecer un punto de interrupción o breakpoint. Un punto de interrupción se puede utilizar para detener la ejecución y activar un sistema de seguimiento en un determinado punto de interés. Existen cuatro tipos de puntos de interrupción: símbolo, de transición, de salida y variable. Los dos primeros tipos serán explicados a continuación.

5.7.1 Qué va a aprender

- Establecer un punto de interrupción en un símbolo SDL
- Establecer un punto de interrupción en una transición
- Listar todos los puntos de interrupción definidos
- Trazar gráficamente cada símbolo SDL ejecutado
- Obtener una referencia textual STD a un símbolo SDL
- Ejecutar continuamente el sistema

5.7.2 Configuración del sistema

Para ver donde se alcanza un punto de interrupción, se iniciará la ejecución del sistema con el comando *Go*. Este comando ejecuta de forma continua transiciones hasta que se produce un error, se alcanza un punto de interrupción o el sistema se encuentra completamente inactivo. Primero debe configurar el sistema de forma adecuada para la ejecución continua:

1. Restablezca el Simulador.
2. Establezca el valor de rastreo del sistema a 0 para evitar que la información de rastreo sea impresa durante la ejecución (puede usar la entrada *Text Level: Set* desde el menú *Trace* para hacer esto).

3. Establezca el valor de rastreo GR para el sistema a 2. Cada símbolo SDL será seleccionado por el Editor SDL al ser ejecutado, lo que le permite un seguimiento de la ejecución a pesar de que no hay rastro de impresión.

5.7.3 Establecer un símbolo BreakPoint

Un símbolo BreakPoint se fija en un símbolo específico SDL perteneciente a los diagramas de los procesos. Estos puntos de interrupción son comprobados antes de la ejecución de símbolos, es decir, el símbolo no se ejecuta cuando se alcanza el punto de interrupción. A continuación se va a mostrar cómo establecer un punto de interrupción en un símbolo del sistema.

1. Desde el Editor SDL, abra el diagrama del proceso donde se encuentre el símbolo a establecer como breakpoint, en el menú de diagramas.
 - Si por alguna razón, el diagrama no está incluido en el menú, debe abrirlo explícitamente. Puede hacer clic en el botón de acceso rápido *Open* y seleccionar “diagrama”.spr, o haga doble clic en el diagrama desde la ventana *Organizer*.
2. En la interfaz de usuario del simulador, seleccione *Connect sdle* desde el menú *Breakpoint*. (Puede que tenga que redimensionar la ventana para ver el menú). Esto establece una conexión entre el *Simulador* y el *Editor SDL*. Como consecuencia, un nuevo menú de puntos de interrupción aparece en la barra del menú del Editor SDL.
3. Vuelve al diagrama SDL y selecciona el símbolo elegido. A continuación, seleccione el comando *Set Breakpoint* desde el nuevo menú *Breakpoints* que ha aparecido en el Editor (la opción que no va acompañada de puntos). El punto de interrupción ya ha sido definido.
4. Inicie la ejecución del sistema pulsando el botón *Go* en el módulo *Execute*. Observe como se seleccionan los símbolos SDL en rápida ejecución a medida que se ejecutan. Finalmente se alcanza el punto de interrupción y la ejecución se detiene. El símbolo donde se ha establecido el breakpoint será el siguiente en ser ejecutado.

5.7.4 Establecer una transición BreakPoint

Un punto de interrupción de transición se fija en una transición específica del sistema. Estos puntos de interrupción se comprueban antes de que las transiciones sea ejecutadas, es decir, la transición no se ejecuta cuando se alcanza el punto de interrupción. Se va a establecer un BreakPoint en el proceso Usuario, cuando se encuentra en el estado *espera_timer* y recibe el timer T.

1. Para definir el punto de interrupción, seleccione *Transition* en el menú *Breakpoint* del Simulador. Este comando toma un número de parámetros.
2. Seleccione el proceso Usuario y haga clic en OK.

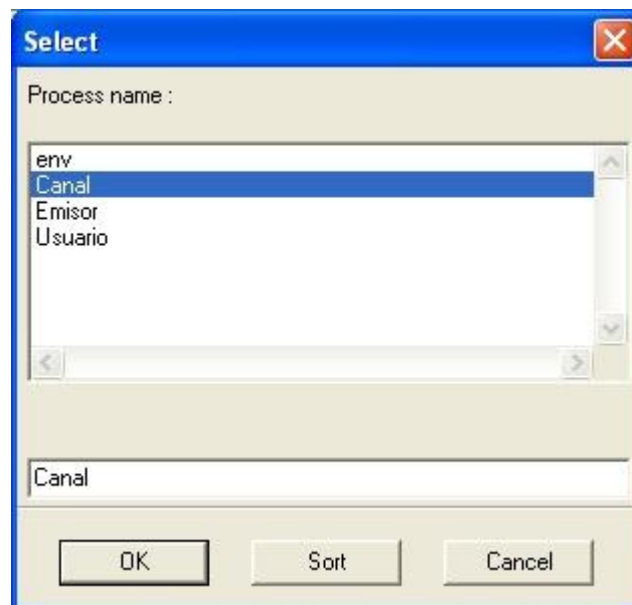


Figura 73: Especificando el proceso

3. Deje el número de instancia en blanco y pulse OK.



Figura74: Deje el número de instancia vacío

4. No especifique nombre de servidor y pulse OK.



Figura 75: No especifique un número de servidor

5. Seleccione *Leer_intencion* y pulse OK.



Figura 76: Especificando el estado espera_timer

6. Seleccione *intencion* y pulse OK.

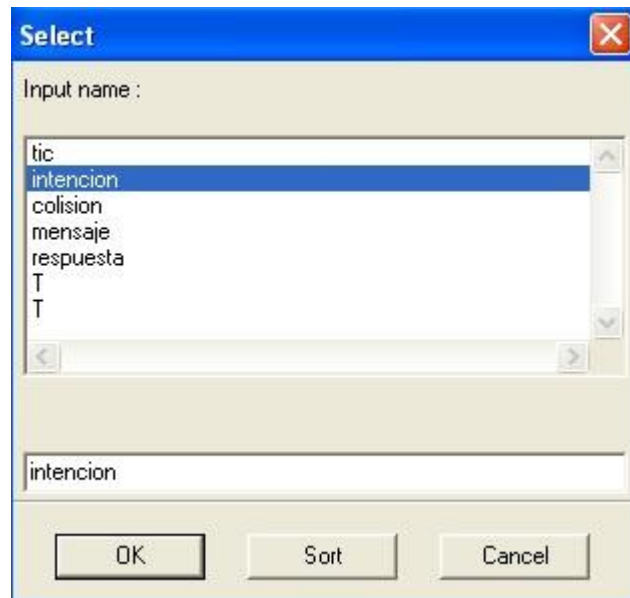


Figura 77: Especificando intencion

7. Simplemente pulse Ok en las restantes ventanas que aparezcan.

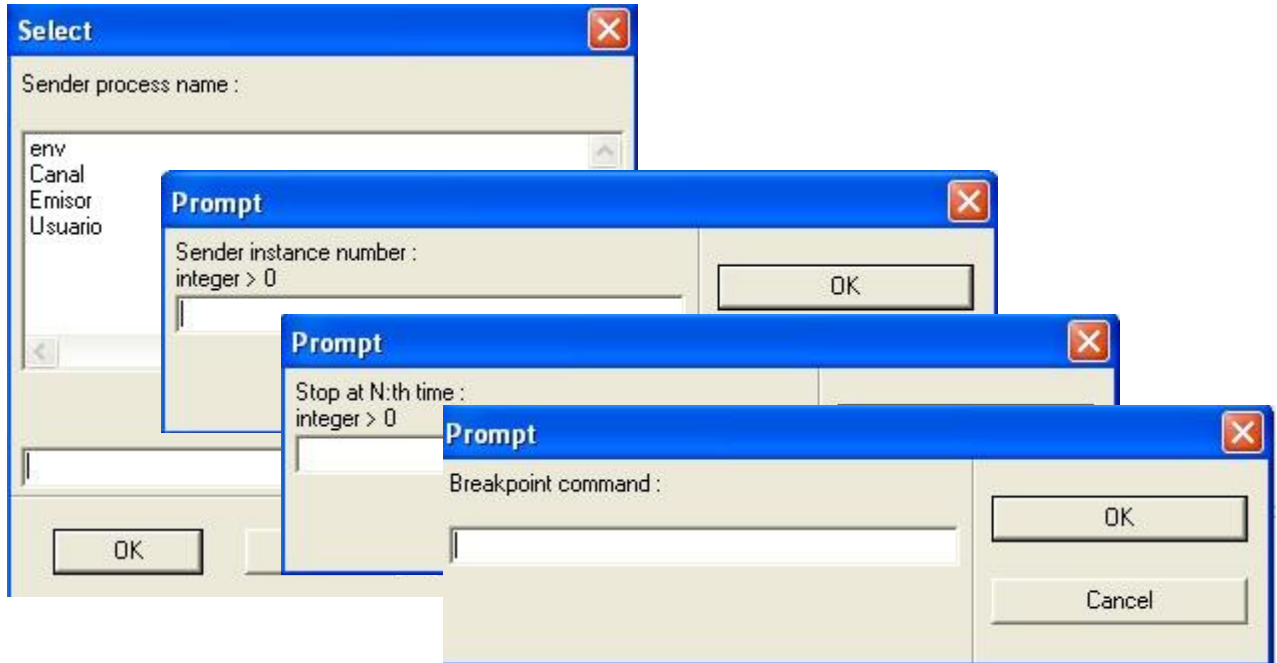


Figura 78: Deje los cuadros de diálogo en blanco

La omisión de la selección del valor de un parámetro se interpreta de modo que cualquier valor, nombre o número coincidirán con ese parámetro. En este caso, cualquier instancia de Usuario y cualquier remitente coincidirán con el punto de interrupción.

8. Para ver como se definió el punto de interrupción, liste los puntos de interrupción existentes desde la opción *Show* en el menú *BreakPoint*.

```
Process name      : Canal
Instance         : any
Service name     : any
State           : Leer_intencion
Input           : intencion
Sender name      : any
Sender instance  : any
Stop each time
```

Simulador SDL

9. Reanude la ejecución de las transiciones haciendo clic en el botón Go. Cuando se alcanza el punto de interrupción, se puede ver que el estado actual del sistema coincide con la definición de punto de interrupción.

```
Breakpoint matched by transition
Pid      : Canal:1
Prd      : Leer_Intencion:1
State    : MAC_completada*, Leer_intencion
Input    : intencion
Sender   : Emisor:1
Now      : 22.0000
```

5.8 Cambiando el sistema

Existen una serie de comandos en el Simulador que nos permiten cambiar el comportamiento del sistema. Estos comandos deben ser utilizados con cuidado, puesto que ya no sería el sistema original el que ha sido simulado después de que se haya hecho uso de uno de estos comandos. Dichos comandos son aún útiles, especialmente en situaciones de depuración, para hacer cambios menores por los que sea posible continuar con la simulación después de que un error haya sido detectado. También pueden ser usados para inducir al sistema a ciertas situaciones, que en otras circunstancias habrían requerido un gran número de transiciones.

5.8.1 Qué va a aprender

- A cambiar y añadir comandos en la ventana de comandos
- Crear un proceso manualmente
- Cambiar el alcance del proceso
- Cambiar el valor de una variable de proceso
- Cambiar el estado de un proceso
- Activar y desactivar un temporizador manual

5.8.2 Algunas preparaciones

En los siguientes dos ejercicios, va a cambiar procesos y temporizadores. Antes de continuar, va a configurar la simulación y la interfaz de usuario del Simulador de una manera adecuada.

1. Restablezca el Simulador.
2. Abra la ventana *Command* desde el menú *View*. Cambiará los comandos ejecutados en la ventana *Command*. El comando *List-Process* va a ser reemplazado por *Examine-Pid*, y se añadirá un nuevo comando, *List-Timer*.
3. En el módulo *List-Process*, seleccione *Edit* desde el menú *Command* en el extremo derecha. (Este menú funciona de la misma manera que el menú *Group* en los módulos de botones).
4. En la ventana de diálogo, cambie el comando por **examine-pid** y de clic en OK.
5. Colóquese en la barra de menús de la ventana *Command* y seleccione *Add Command*. En el cuadro de diálogo, escriba el comando *list-timer* y pulse OK.
6. Un nuevo módulo de comandos se habrá añadido a la ventana. Cambie el tamaño de la ventana de manera que los tres módulos sean visibles.
 - También puede reducir el número de líneas de texto que aparecen en un módulo seleccionando *Size* desde el módulo *Command*. En el cuadro de diálogo, defina el número de líneas con un control deslizante.

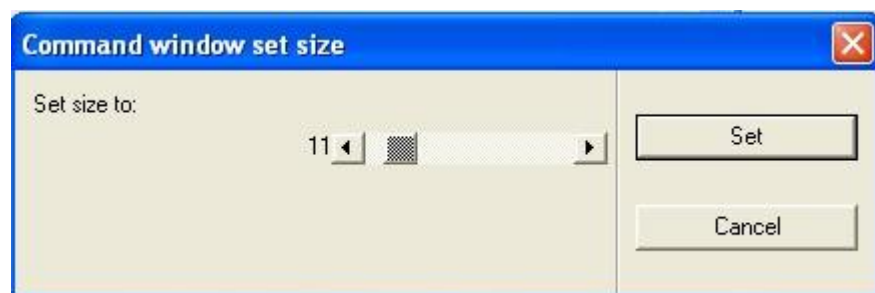


Figura 79: Ajustando número de líneas

Simulador SDL

7. Establezca el valor de rastreo del sistema a 6.
8. Ejecute las dos primeras transiciones para asegurarse de que tanto el proceso Canal como Emisor se haya iniciado.
9. Si el rastreo GR en la ventana Editor SDL hace difícil la visualización del resultado en la ventana *Command*, establezca el rastreo GR del sistema a 0.

5.8.3 Creación de procesos y cambio en estado del Timer

Dentro del Simulador, puede crear una nueva instancia de proceso usando el comando *Create*.

1. Para ello, seleccione la opción *Create Process* dentro del menú *Change*. En la siguiente ventana, seleccione el proceso que quiere crear y pulse OK.

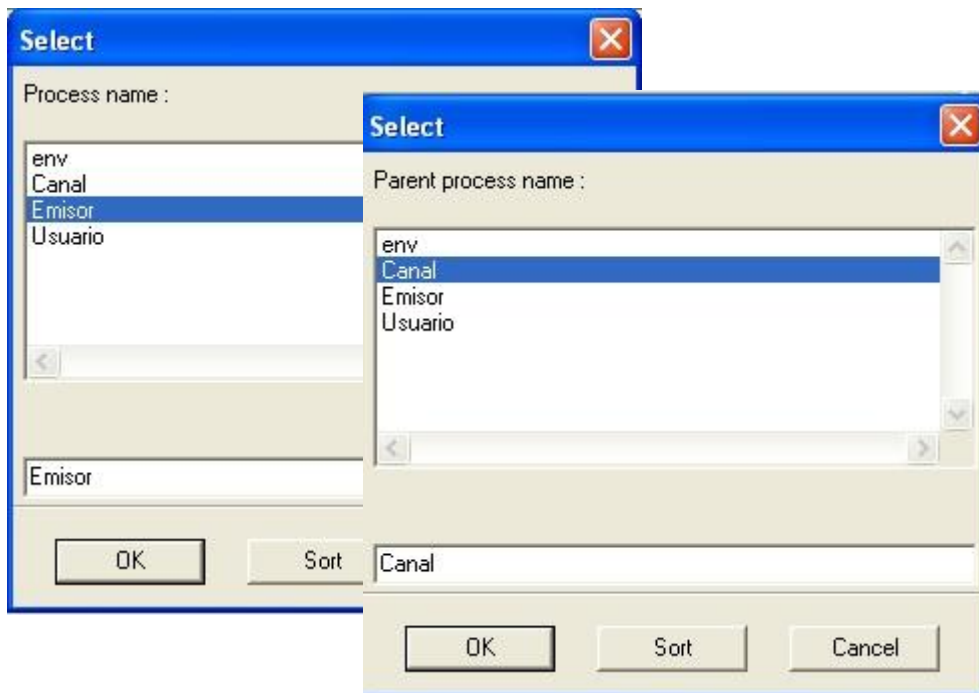


Figura 80: Creando proceso Emisor desde proceso Canal

Simulador SDL

2. Tenga en cuenta, que primero debe cambiar el proceso actual, conocido como *process scope*. Para cambiarlo seleccione *Set Scope* desde el menú *Examine*. Seleccione el proceso Canal y pulse OK.

En este ejercicio, también puede ejecutar y reajustar los temporizadores del sistema.

1. Restablezca el temporizador seleccionando *Reset Timer* desde el menú *Change*. Seleccione *Timer T* y pulse OK.



Figura 81: Seleccionando Timer

2. Para establecer Timer elija *Set Timer* desde el menú *Change*, seleccione T e introduzca el valor deseado.

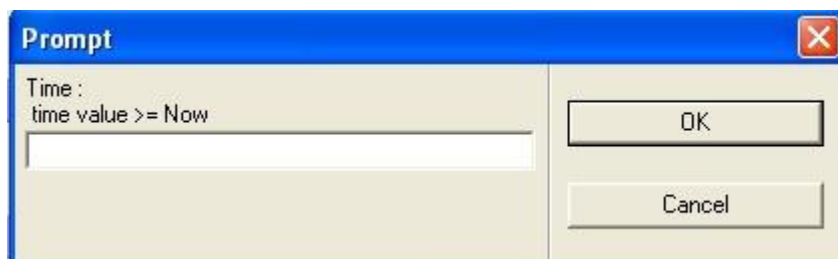


Figura 82: Introduzca valor de timer

5.9 Gráficos de Generación de MSC

En este ejercicio, se va a demostrar el poder de MSC como un método para ilustrar, de manera gráfica, el comportamiento dinámico del sistema. Esto se puede realizar fácilmente al simular el sistema mediante el uso de rastro MSC. MSC transforma algunos de los eventos que ocurren en SDL en eventos MSC, como envío de señales y creación dinámica de procesos. El trazo se puede seguir de manera gráfica en un editor MSC durante su ejecución.

Se va a ejecutar el simulador y a generar la traza MSC de los eventos que se producen.

5.9.1 Qué va a aprender

- A iniciar y detener el registro de eventos MSC
- A rastrear desde MSC hasta un diagrama SDL.

5.9.2 Inicializando MSC

1. Restablezca la simulación.
2. Asegúrese de que la traza GR está desactivada para evitar que la ventana del Editor SDL se actualice (seleccione *SDL Level: Show* desde el menú *Trace* y verifique que GR trace es 0).

De forma predeterminada, el trazo MSC está habilitado para todo el sistema, usted debe, sin embargo, iniciar el registro interactivo de todos los eventos MSC.

3. Elija *MSC Trace: Start* en el menú *Trace*. Se abrirá una ventana de diálogo, donde se le solicitará que especifique la cantidad de símbolos a incluir en la traza MSC. Seleccione la opción 1 “MSC with States” y pulse OK.

Simulador SDL

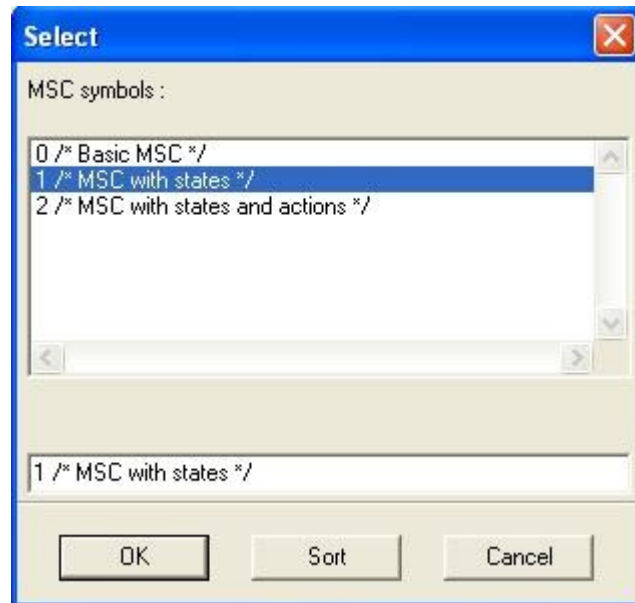


Figura 83: Especificando la inclusión de estados

4. Se abre un Editor MSC, mostrando un diagrama MSC llamado “Simulator Trace”. Si es necesario, mueva y cambie el tamaño de la ventana para que se vea correctamente en la pantalla junto con la interfaz de usuario.



- Antes de redimensionar el editor MSC, puede ocultar el menú de símbolos y la ventana de texto mediante el uso de los botones de acceso rápido. No va a editar el MSC generado, por lo que no es necesaria la utilización de esas ventanas.

Inicialmente, el sistema tiene dos instancias activas, el proceso Canal_1_1, así como env_0, introducido para representar el entorno del sistema.

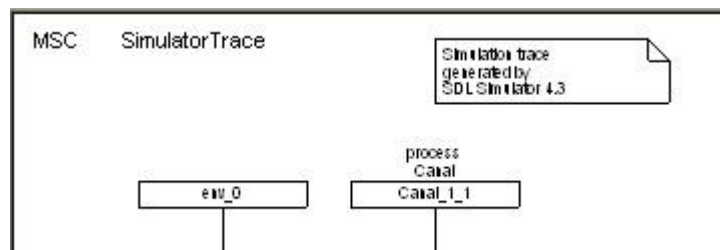


Figura 84: Apariencia inicial de MSC

5.9.3 Seguimiento de la ejecución MSC

1. Comience la simulación del sistema, para ello haga clic en el botón *Transition* y observe el comportamiento del sistema en la traza MSC.

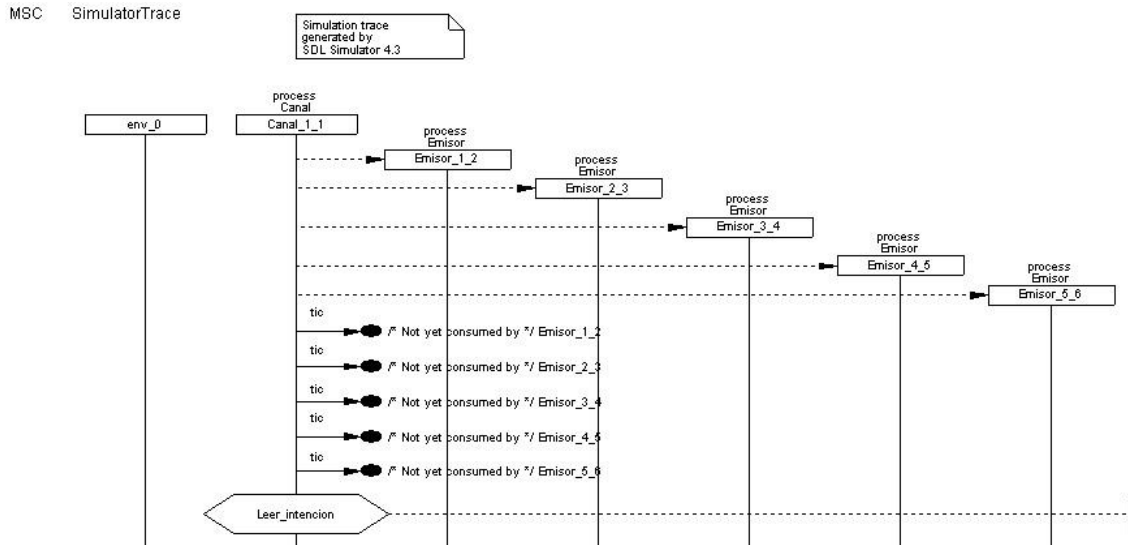


Figura 85: envío de señales tic

Como se puede apreciar, los tics enviados por canal a los diferentes emisores son catalogados de momento como un mensaje “perdido”, indicado por el círculo relleno. También se puede ver el texto Emisor_1_2 asociado al círculo, que nos está indicando el receptor de ese mensaje.

2. Ejecute la siguiente transición. Como se puede ver ha aparecido el símbolo de condición Leer_Intencion en el eje de instancia del proceso Canal. Este símbolo muestra que el proceso ha comenzado la ejecución.

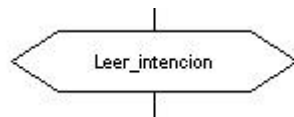
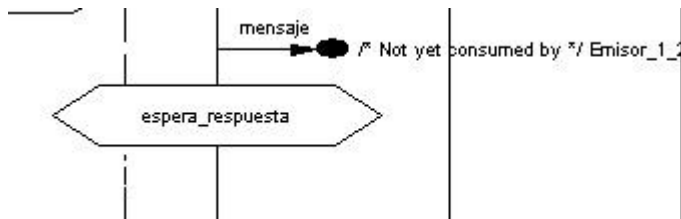


Figura 86: símbolo condición

Simulador SDL

3. Ejecute 4 transiciones más hasta llegar el envío de “mensaje” del usuario hacia el proceso Emisor_1_2.



4. Continúe realizando transiciones y observe como todos los usuarios envían “mensaje” a emisor.
5. Siga realizando transiciones hasta observar el envío de “colision” por parte del Canal.

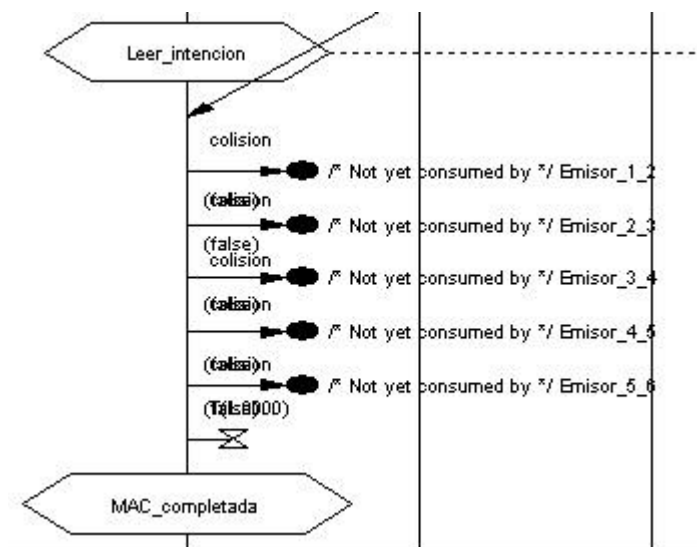


Figura 87: envío de la señal colisión.

6. Como se puede ver, puede apreciar todo el funcionamiento del sistema siguiendo la trama MSC. Para realizar la simulación de forma más rápida que ejecutando transición a transición. Pulse el botón *Go* desde el módulo *Execute*.
7. Compruebe como se va ejecutando todo el sistema gráficamente en la traza MSC.
8. Para detenerlo, haga clic en el botón *Break* en el módulo *Execute*.

5.9.4 Volver a un diagrama SDL

A partir del diagrama MSC generado, puede volver a los diagramas SDL de origen.

1. Desde la ventana del editor MSC, seleccione *Info Window*. Se abre una ventana, que contiene información sobre el objeto gráfico que está actualmente seleccionado. (La cantidad y tipo de información depende del objeto gráfico seleccionado).

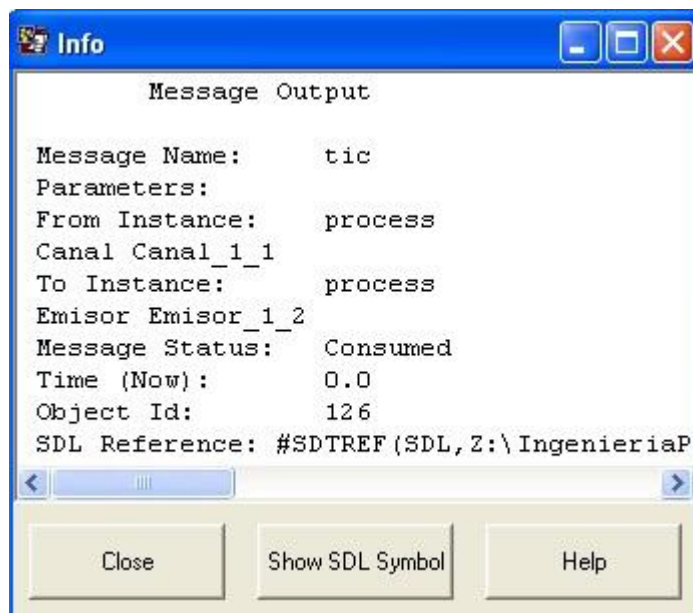


Figura 88: La ventana *Info Window*

2. Seleccione diferentes objetos del Editor MSC y nota como la información en la ventana *Info* cambia.
3. Si ha seleccionado el mensaje haciendo clic en un punto que está más cerca al punto de inicio del mensaje que a su punto final, se seleccionará el símbolo de salida SDL correspondiente.
 - De lo contrario, se seleccionará el símbolo de entrada SDL correspondiente.

5.9.5 Finalizando traza MSC

1. Detenga el registro de eventos MSC seleccionando *MSC Trace: Stop* desde el menú *Trace*.
2. En el Editor MSC, guarde el diagrama MSC generado bajo el nombre **SimulatorTrace.msc**.
3. Salga del Editor MSC.

5.10 El mapa de cobertura

En este ejercicio final del Simulador SDL, aprenderá a usar el mapa de cobertura. El mapa de cobertura es una herramienta gráfica que muestra la cantidad de un sistema cubierto durante una simulación en los términos de ejecutar las transiciones o símbolos. Comprobando la cobertura del sistema, puede ver que partes no se han ejecutado en la simulación hasta ahora.

5.10.1 Qué va a aprender

- A crear un archivo de cobertura
- A iniciar el Visor de Cobertura
- Interpretar y cambiar la información del árbol de cobertura
- Abrir un gráfico de una cobertura más detallada
- Salir de la interfaz de usuario del Simulador

5.10.2 Puesta en marcha del mapa de cobertura

1. Restablezca el Simulador.
2. Ejecute varias transiciones del sistema y vea cuanto se ha explorado del sistema hasta el momento. Simplemente ponga en marcha el mapa de cobertura del Simulador, se muestra la información de cobertura actual.
3. Seleccione *Coverage* desde el menú *Show*. Se abre la ventana principal *Coverage Viewer*.

5.10.3 Usando Coverage Viewer



1. Si se muestra *Symbol Coverage Tree*, puede cambiarlo a un árbol de cobertura de transición haciendo clic en el botón de acceso rápido *Tree Mode*.
2. Para ver todo el árbol de cobertura de transición, haga clic en el botón de acceso rápido *All Nodes*.

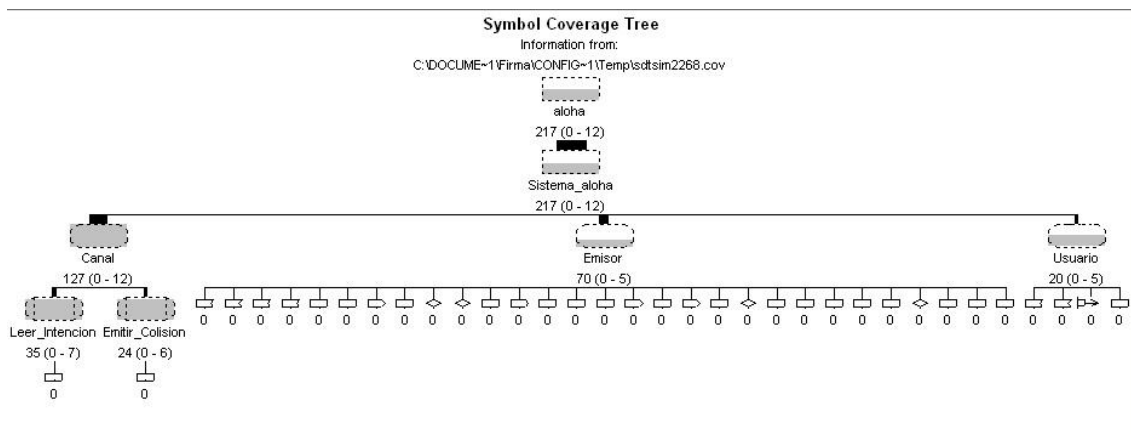


Figura 89: Symbol Coverage Tree

- Dado que el árbol de cobertura es bastante grande, es posible maximizar la ventana (utilizando el gestor de ventanas) al trabajar con el mapa de cobertura. Cuando vuelva al Simulador, puede restaurar el tamaño de la ventana a su tamaño original.



- Alternativamente, también puede alejar el contenido de la ventana con un número de clics en el botón de acceso rápido.

El árbol de cobertura muestra la estructura de diagrama del sistema aloha. Debajo de cada diagrama de proceso, puede ver todas las posibles transiciones

Simulador SDL

de dicho proceso, definidas por un estado, y las posibles entradas de señal de ese estado.

El número debajo de cada símbolo en el árbol representa el número de veces que ese símbolo se ha ejecutado hasta el momento. Además, cada símbolo se encuentra relleno de un color gris para indicar en qué medida se ha ejecutado. Las partes del sistema que no han sido ejecutadas tienen un valor de “cero” y se encuentran “vacías”. Las que se han ejecutado por completo tienen un valor distinto de cero y se encuentran completamente coloreadas.



3. Para ver sólo aquellas transiciones que no han sido ejecutadas, haga clic en el botón de acceso rápido *Least*. Ahora puede ver que señales deben ser enviadas y a que estados para ejecutar el resto del sistema.

- El botón de acceso rápido *Least* muestra los símbolos que han sido ejecutados un menor número de veces.



4. De la misma manera, solo para ver las transiciones ejecutadas al menos una vez, haga clic en el botón de acceso rápido *Most*. Puede ver las señales que se han enviado hasta ahora en el sistema.

- El botón de acceso rápido *Most* muestra los símbolos que han sido ejecutados el mayor número de veces.

5. Para ver todo el árbol de nuevo, haga clic en el botón de acceso rápido *All Nodes*. Si desea ver una transición en el Editor SDL, haga doble clic en uno de los símbolos de entrada de señal o los símbolos de inicio de estado.

El visor de cobertura también puede mostrar un árbol de cobertura de símbolos, para saber cuántas veces se ha ejecutado cada símbolo.

6. Cambie a *symbol coverage tree* haciendo clic en el botón *Tree Mode*. Debajo de cada diagrama de proceso, verá un pequeño icono para cada símbolo SDL. Para ver que símbolo SDL representa cada icono, haga doble clic sobre él.

7. Cambie de nuevo a *transition coverage tree* y vuelva al simulador.

5.10.4 Examinando detalles de cobertura



1. Seleccione el diagrama del sistema desde el visor de cobertura. Desde el menú *Tools*, seleccione *Show Details*, o haga clic en el botón de acceso rápido. El gráfico de cobertura que aparece muestra la cantidad de transiciones que se han ejecutado un número determinado de veces.

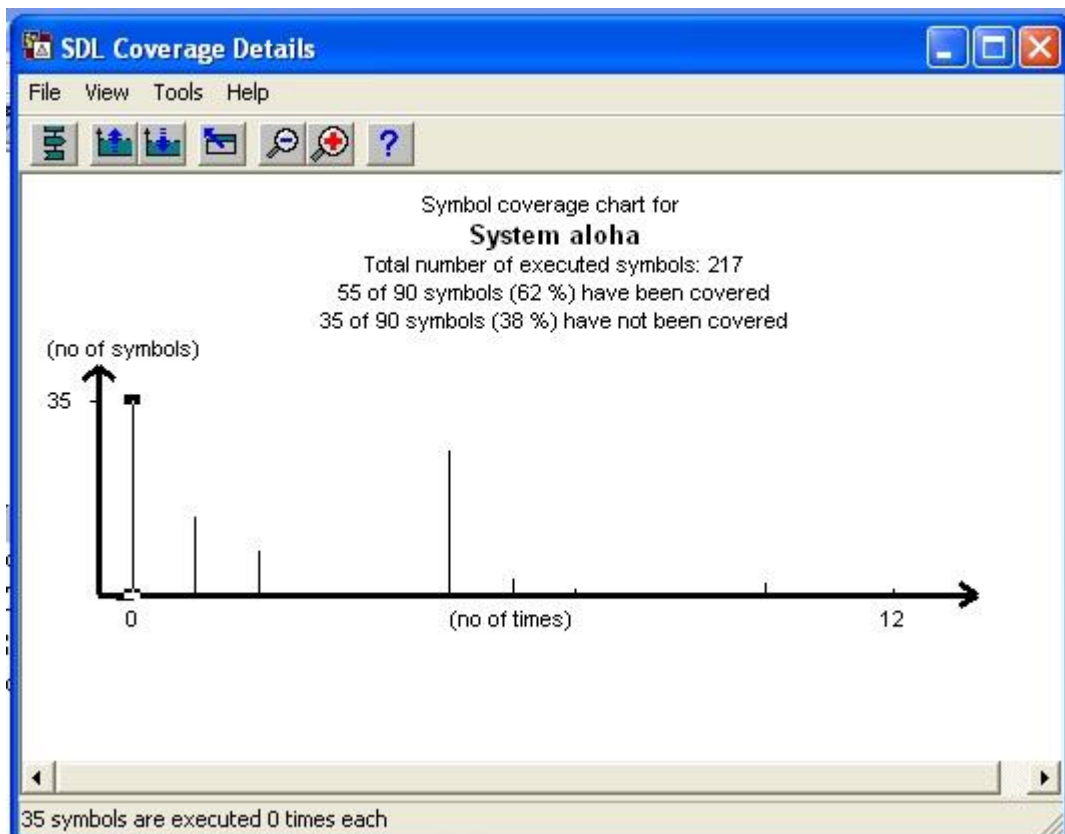


Figura 90: Ventana *Coverage Details*

2. Seleccione una barra cada vez y fíjese en la información de la barra de estado en la parte inferior de la ventana. Verá cuantas transiciones se han ejecutado 0, 1,2 veces. Para ver las transiciones que no se han ejecutado, proceda de la siguiente manera:



3. Seleccione la “barra cero” a la izquierda. Haga clic en el botón de acceso rápido *Show Editor*. Se abre el editor SDL, mostrando las transiciones que aún no han sido ejecutadas.
 - Si las transiciones se pueden encontrar en más de un diagrama, una ventana de confirmación se emite en cada diagrama abierto.

Simulador SDL

Simplemente haga OK para ver las transiciones del siguiente diagrama.

4. Seleccione otro símbolo en la ventana principal del árbol de cobertura. La ventana *Coverage Details* se actualiza para mostrar la cobertura de ese símbolo.
5. Manipule el visor de cobertura todo lo que quiera.

5.10.5 Abandonando la interfaz de usuario Simulador

Ahora puede cerrar el Simulador.

1. Primero, salga de *Coverage Viewer* desde el menú *File*.
2. A continuación, salga del simulador desde el menú *File*. Se le pedirá si desea guardar los cambios realizados en las variables, de comandos y de botones. Si opta por guardarlos, se convertirán en los valores predeterminados la próxima vez que inicie el Simulador desde ese directorio.

5.11 Hasta ahora...

Ha aprendido a “animar” un Sistema SDL mediante la generación, ejecución y el rastreo de un simulador.

Si su configuración incluye la herramienta de validación, le sugiero que proceda con los ejercicios en el validador.

6.VALIDADOR SDL

El validador SDL es la herramienta que se utiliza para validar el comportamiento de los sistemas SDL, usando las técnicas de exploración de estado. En este tutorial, practicaré “de primera mano” sobre el sistema aloha.

Para ser correctamente asimilado, este tutorial presupone que ha realizado los ejercicios dispuestos anteriormente referentes al Editor y Analizador, así como el de Simulación.

Con el fin de aprender a utilizar el validador, lea con atención el tutorial y realice los ejercicios en su sistema operativo tal y como se describen a continuación.

6.1 Propósito de este tutorial

El propósito de este tutorial es que se familiarice con los requisitos esenciales referentes a la funcionalidad de validación de la Suite SDL. Con validación nos referimos a la exploración de los estados de un sistema SDL mediante métodos de gran alcance y herramientas que encontrará prácticamente cualquier tipo de posibles errores en tiempo de ejecución que puedan ser difíciles de encontrar con simulación regular y técnicas de depuración.

Este tutorial está diseñado como una visita guiada a través de la Suite SDL, donde realizará en sus equipo una serie de ejercicios prácticos.

Se va a utilizar para ello el ejemplo que se ha estado usando, el sistema aloha.

Nota:

Debe tener un compilador C instalado en su sistema informático para validar un sistema SDL. Asegúrese de saber a qué compilador C se tiene acceso antes de empezar el tutorial.

6.2 Generar e iniciar un validador

Además de la simulación de un sistema, también es posible validar el sistema mediante la utilización del Validador SDL. Un validador se puede utilizar para automáticamente encontrar errores e incoherencias en el sistema, o para verificar el sistema en contra de los requisitos.

De la misma forma que para un simulador, debe generar un validador ejecutable y comenzar con una interfaz de usuario adecuada.

6.2.1 Qué va a aprender

- A generar y ejecutar un validador ejecutable rápidamente.

6.2.2 Inicio rápido de un validador

Un validador puede ser generado e iniciado de la misma manera descrita anteriormente en el simulador, utilizando la ventana *Make*, desde el menú *Tools* en *Organizer*.

Sin embargo, a continuación se va a mostrar una forma más rápida.

1. Asegúrese de que el icono del diagrama del sistema se encuentra seleccionado en *Organizer*.



2. Haga clic en el botón de acceso rápido *Validate*. Ocurrirán los siguientes eventos en rápida sucesión:

- Se genera un validador ejecutable. Aparecen mensajes en la barra de estado similares a cuando se genera un simulador, que terminan con “Analyzer done”. Es la misma acción que se hubiera producido si hubiera abierto la ventana *Make* y seleccionado un kernel adecuado. Si lo desea, puede verificar que el kernel de validación ha sido usado observando el final de la ventana *Organizer log*.

Validador SDL

- Se inicia una interfaz gráfica de usuario para el validador. En la barra de estado de *Organizer* se podrá leer “Validator UI started”. Es la misma acción que se produce seleccionando manualmente *Validator UI* desde el menú *Tools*.
- Se inicia el validador generado. Validator UI muestra el mensaje “Welcome to SDL Validator”. Es la misma acción que se produce si se realiza manualmente mediante el botón de acceso rápido *Open* y abriendo el ejecutable **aloha_val.exe**.

Le pedirá que introduzca el fichero donde se encuentra el valor de las variables externas, puesto que de momento no se va a utilizar, introduzca los valores manualmente.

N_nodos = 5

PG = 0.95

Validador UI tiene este aspecto:

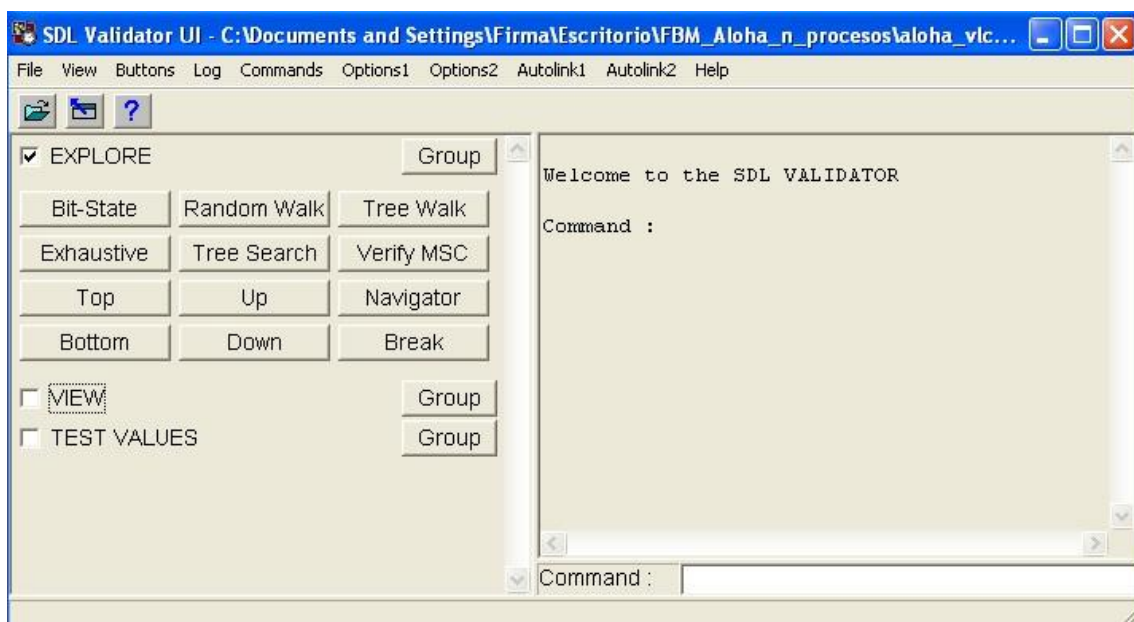


Figura 91:Validator UI

Como puede ver. La interfaz gráfica de usuario del validador es muy similar a la interfaz gráfica de usuario del simulador que ha aprendido a utilizar en ejercicios anteriores. Sin embargo los módulos de botones a la izquierda son diferentes, e incluye algunos menús adicionales.

Validador SDL

Un validador contiene el mismo sistema de monitoreo que un simulador. La única diferencia es el conjunto de comandos disponibles.

Cuando se inicia un validador, se crea la instancia de proceso estática en el sistema, en este caso canal, pero sus transiciones iniciales no son ejecutadas. El proceso es ejecutado en el proceso canal. Puede comprobarlo viendo la cola de listas de proceso.

1. Localice el botón View en la parte izquierda de la ventana, y haga clic en el botón *Ready Q*. La primera entrada en la lista es canal, esperando a ejecutar su transición inicial.
 - Si el módulo View parece estar vacío, lo que tiene que hacer es clic en el botón a la izquierda del nombre del módulo. El módulo de botones se expande. Puede contraer y expandir cualquier módulo de botones utilizando estos botones de conmutación.



Figura 92: Módulo de botones desplegable.

- Los botones en el módulo View ejecutan el mismo tipo de comandos que aquellos en el Simulador.
2. Si fuera necesario, cambie el tamaño de la ventana Validador para que todos los módulos de botones sean visibles. También puede reducir el ancho del área de texto. En ejercicios venideros, tendrá varias ventanas abiertas al mismo tiempo.

6.2.3 Conceptos básicos de un validador

Antes de empezar a trabajar con los ejercicios de validación, debe tener una comprensión de los conceptos básicos del validador SDL.

- Al examinar un sistema SDL utilizando el validador, el sistema SDL está representado por una estructura llamada “behavior tree” (árbol de comportamiento). En esta estructura en árbol, un nodo representa un estado del sistema SDL completo. Al conjunto de todos esos estados del sistema se les conoce como el espacio de estados del sistema.

Validador SDL

- Moviéndose por el árbol de comportamiento puede explorar el comportamiento del sistema SDL y examinar cada estado del sistema que se encuentre. Esto se llama exploración de espacio de estado y se puede realizar ya sea manualmente o automáticamente.

- El tamaño y la estructura del árbol de comportamiento se determina por un número de opciones de espacio de estado en el validador. Estas opciones afectan al número de estados del sistema que se generan para una transición en un proceso SDL, y el número de posibles ramificaciones de estado en el árbol de comportamiento.

6.3 Navegando en un árbol de comportamiento

En este primer ejercicio, vamos a explorar el sistema de estados del sistema aloha navegando manualmente en el árbol de comportamiento. El Validador se comportará de manera similar al comportamiento del Simulador. Sin embargo también hay diferencias importantes, que serán resaltadas más adelante.

Por defecto, el validador está configurado de manera que da como resultado un espacio de estados lo más pequeño posible. En esta configuración, una transición entre dos estados del árbol de comportamiento corresponde siempre a una transición completa en el proceso gráfico SDL. Además el número de posibles ramificaciones de un estado tiene un número limitado.

6.3.1 Qué va a aprender

- A utilizar la herramienta *Navigator*.
- A imprimir el trazo impreso y el trazo GR.

6.3.2 Configuración de la exploración

Al explorar de forma interactiva el árbol de comportamiento, utilizará una herramienta de validación llamada *Navigator*.

1. Inicie *Navigator* haciendo clic en el botón *Navigator* dentro del módulo *Explore*.



Figura 93: herramienta *Navigator*

Navigator muestra parte del árbol de comportamiento alrededor del actual estado del sistema. En general, el cuadro superior representa el árbol de comportamiento de transición que conduce al estado actual, es decir, la transición que acaba de ser ejecutada. Los cuadros a continuación representan las posibles desde el estado actual. Se etiquetan como Next 1, Next 2... y aún no han sido ejecutadas. Dado que el sistema se encuentra en su estado inicial, no hay ningún nodo superior El siguiente nodo es el inicio de transición de Canal.

2. Para poder ver el trazo impreso, abra la ventana *Command* desde el menú *View*. (El trazo no se imprime en la ventana general de validación).
3. Para activar el trazo GR de símbolos SDL, seleccione *Toggle SDL trace* desde el menú *Commands* en la ventana de validación, *SDL trace* se encuentra ahora habilitado. Sin embargo, un editor SDL no se abrirá hasta que la primera transición sea ejecutada.

6.3.3 Usando el navegador

1. En *Navigator*, ejecute la siguiente transición haciendo doble clic en el nodo Next 1. Ocurrirá lo siguiente, en orden:
 - En *Navigator*, el nodo superior Up 1 muestra la transición recién ejecutada, mientras que el nodo Next 1 muestra que la siguiente ejecución será la transición de inicio de Emisor.



Figura 94: Última y siguiente transición

- Un editor SDL se abre y los símbolos que sólo se ejecutaron quedan seleccionados. Tenga en cuenta la diferencia en comparación con el simulador, donde el editor SDL selecciona el símbolo que va a ser ejecutado.
- La ventana de comandos muestra el trazo impreso para la transición ejecutada, la transición de inicio Canal.

```
Print-Trace 1 Command
*** Transition START
*   PID    : Canal:1
*   State  : start state
*   DECISION Value: false
*   CREATE Emisor:1
*   ASSIGN Emisor_ID... := Emisor:1
*   ASSIGN i := 2
*   DECISION Value: false
*   CREATE Emisor:2
```

Figura 95: Trazo impreso de la transición ejecutada

Validador SDL

2. Si es necesario, mueva y cambie el tamaño de todas las ventanas abiertas para que sean todas perfectamente visibles en la pantalla.
3. Haga doble clic en el nodo Next 1 para ejecutar la siguiente transición. La transición de inicio Emisor puede ser observada en la ventana de comandos y en el Editor SDL.

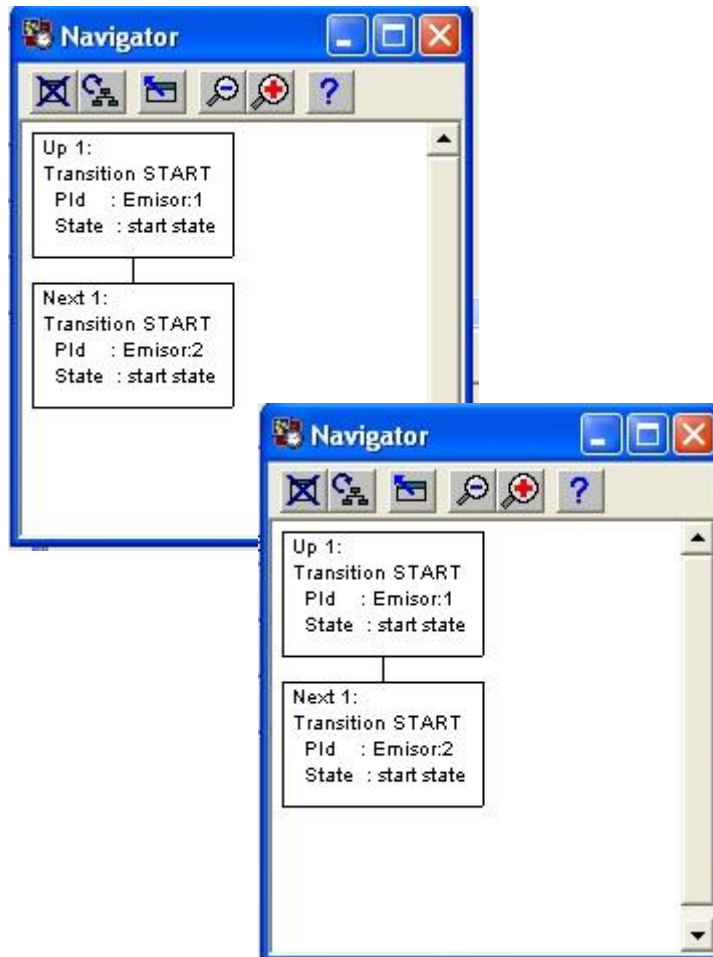


Figura 96: Transiciones en *Navigator*

Validador SDL

4. Siga ejecutando las siguientes transiciones hasta llegar al lugar donde se produce el error dinámico del sistema aloha. En lugar de mostrar la siguiente transición, *Navigator* muestra un mensaje de error.

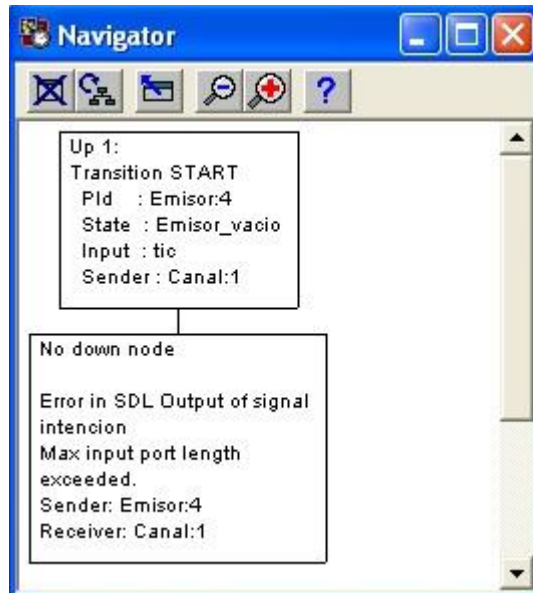


Figura 97: Error dinámico

- El mensaje de error también puede ser visualizado en la ventana *Commands*, en la parte inferior del módulo *Print-Trace*

```
Error in SDL Output of signal intencion
Max input port length exceeded.
Sender: Emisor:4
Receiver: Canal:1
```

Figura 98: Mensaje error en el módulo *Print-Trace*

Como se puede observar el error se produce porque se ha excedido el número de puertos, esto se debe a que el sistema tiene el número máximo de puertos predeterminado a 3, y se necesita aumentarlo a 5 para cubrir los 5 nodos que forman nuestro sistema. La solución a este error se verá en un próximo apartado.

No puede ir más debajo de esta rama del árbol de comportamiento, ya que un error, por defecto, trunca el árbol en el estado actual.

5. Suba en el árbol un par de niveles haciendo clic consecutivamente en el nodo Up 1. Esta forma de volver atrás no es posible cuando se ejecuta una simulación, como habrá podido comprobar en el tutorial sobre el Simulador.

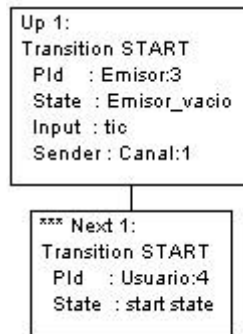


Figura 99: Marcando una transición por la que ya se ha pasado

Tenga en cuenta que el nodo Next 1 está marcado con tres asteriscos ***. Se utiliza para indicar que ya se ha pasado por esta transición.

Si el número de transiciones de un estado es grande, puede resultar difícil visualizarlos todos en *Navigator* cuando se utiliza una estructura de árbol. Para superar este problema puede cambiar la disposición para ver una lista de estructura.

- Haga clic en el botón de acceso rápido *Toggle Tree* para ver como es la estructura de la lista. Ahora es más fácil ver las señales posibles.

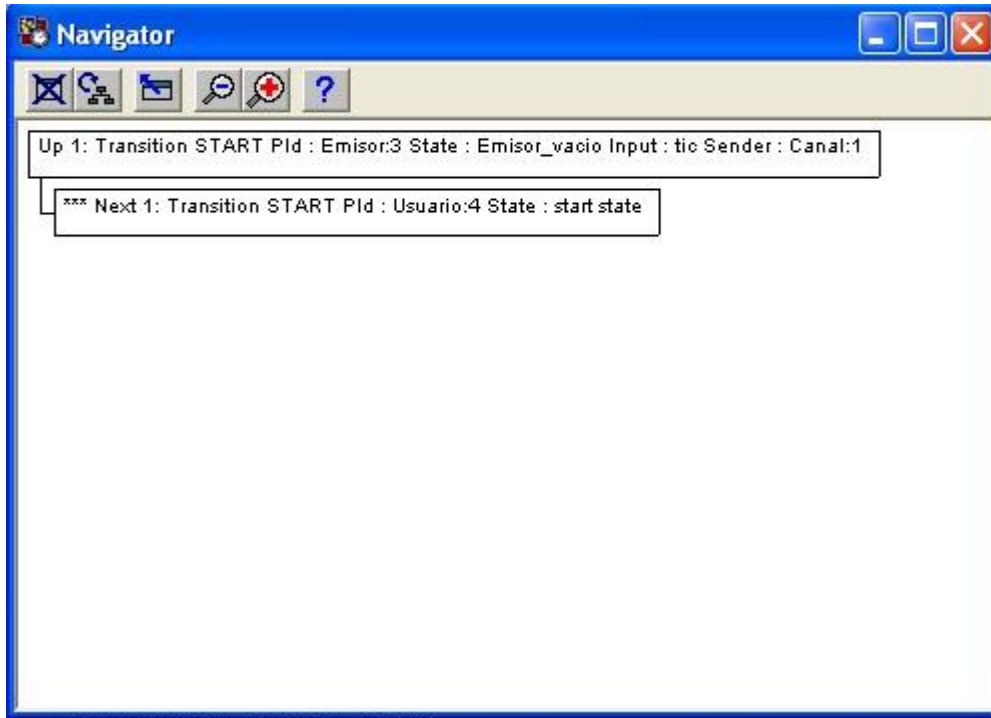


Figura 100: estructura de la lista

- Cambie de nuevo a la estructura de árbol.

No vamos a continuar más allá para conocer la conducta del árbol de comportamiento en este ejercicio.

6.4 Más posibilidades de rastreo y visualización

En este ejercicio, vamos a echar un vistazo a algunas posibilidades de trazado adicional y visualización del validador.

6.4.1 Qué va a aprender

- A imprimir un trazo completo del estado de inicio.
- A utilizar los comandos de visualización
- A utilizar el recurso de rastreo MSC
- A llegar a un estado mediante el uso de los comandos de ruta

6.4.2 Uso de los comandos de visualización

1. Asegúrese de que se encuentra en el mismo estado después de haber realizado el último paso del ejercicio anterior.

Para ver una huella impresa completa del estado de inicio del estado actual, puede utilizar el comando **Print-trace**. Es un parámetro que necesita un número de niveles para imprimir una copia impresa.

2. En la línea de entrada del Validador, introduzca el comando **pr-tr 9** (puede usar cualquier número). El trazo se imprime en el área de texto de la ventana principal. Nos proporciona una visión general de lo que ha sucedido en el sistema SDL hasta ahora.
3. El Validador SDL soporta las mismas posibilidades de visualización que el Simulador SDL. Haga clic en el botón *Timer list* en el módulo *View* para contemplar el temporizador activo establecido por el proceso Canal.
4. Puede examinar los distintos emisores y usuarios mediante el botón *Set Scope*, y seleccionando el proceso Emisor o Usuario.

6.4.3 Usando MSC

Además de trazos textuales y gráficos, el validador también puede realizar una traza MSC.

1. En primer lugar, seleccione *Toggle SDL Trace* desde el menú *Commands*. Se abrirá un editor MSC un gráfico de secuencia de mensajes desde el inicio hasta el estado actual.
 - También puede cerrar el editor SDL para evitar tener muchas ventanas abiertas en pantalla.

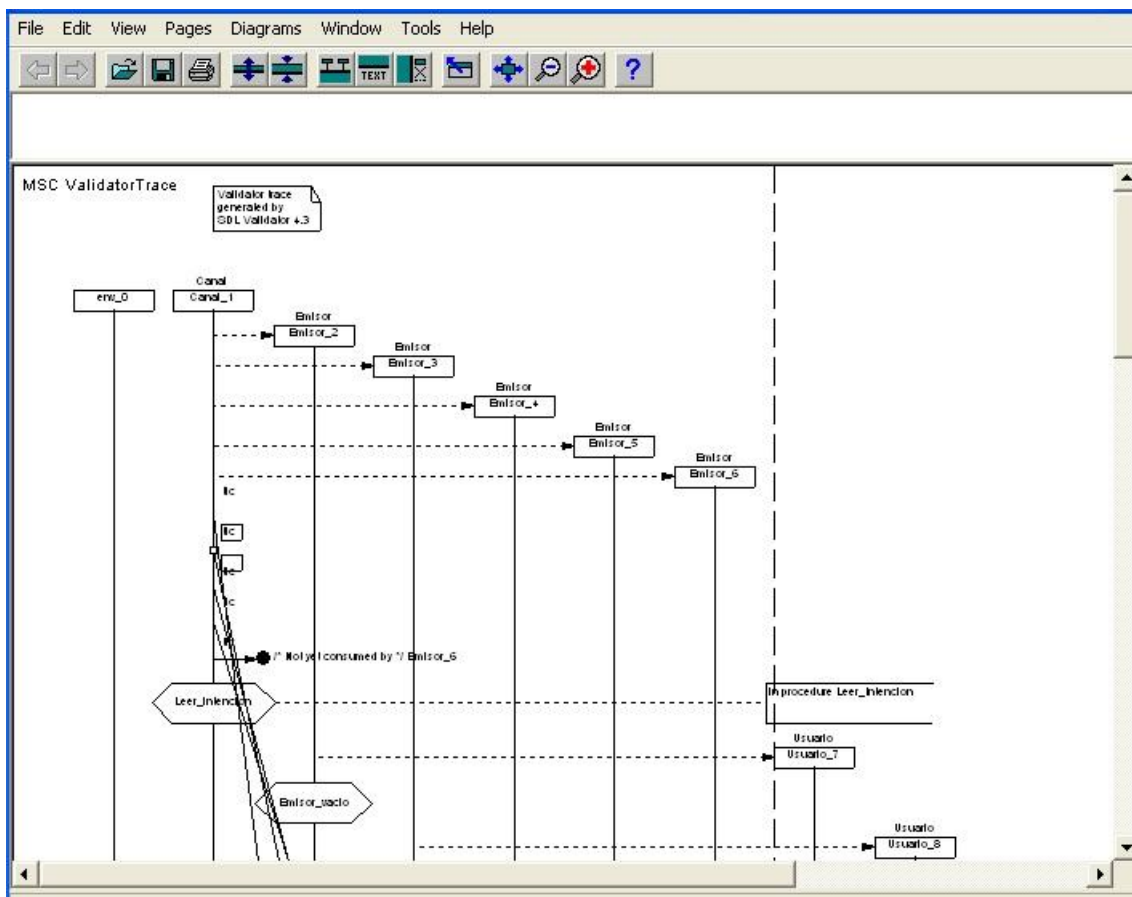


Figura 101: MSC Validation Trace

2. Suba unos niveles en *Navigator*. Note como la selección en el editor MSC cambia para reflejar el evento MSC correspondiente al estado actual.

3. Baje de nuevo y compruebe el comportamiento del editor MSC al ir cambiando de estado.
4. Desactive *Toggle MSC Trace* desde el menú *Commands*. A menos que otros MSC sean abiertos, el editor MSC permanecerá cerrado.

6.4.4 Ir a un estado mediante los comandos de ruta

Puede utilizar los comandos **Print-Path** y **Goto-Path** para volver a un estado en el que ya ha estado anteriormente.

1. Ejecute el comando Print-Path en la línea de entrada de texto. Nos mostrará el camino recorrido en el árbol de comportamiento desde el estado de inicio hasta el estado actual.

```
Command : Print-Path  
1 1 1 1 0
```

- Los números mostrados en la ruta son los mismos que los números de transición en *Navigator*.
2. Suba unos niveles en *Navigator*.
 3. En el área de texto, busque la ruta impresa anteriormente, y seleccione los números de la ruta, incluido el cero.
 4. En la línea de entrada, escriba **Goto-Path**, se desplegará una ventana donde escribirá los números de la ruta
 5. Pulse <Enter> para ejecutar el comando. Habrá llegado al estado anterior.
 - Si comete un error al ingresar los números de ruta, puede borrar la línea de entrada utilizando la tecla <Abajo> e intentándolo de nuevo.

6.5 Validando un Sistema SDL

En ejercicios anteriores, ha navegado de forma manual en el árbol de comportamiento. También ha encontrado una situación de error mediante el estudio de *Navigator* y la huella impresa de la ventana de comandos.

En este ejercicio, se va a mostrar cómo encontrar errores y posibles problemas mediante la exploración automática del espacio de estados del sistema aloha.

Esto se conoce como la validación de un Sistema SDL.

6.5.1 Qué va a aprender

- A llevar a cabo una exploración de espacio automática
- Examinar los errores indicados mediante la utilización de *Report View*
- Cambiar las opciones de espacio y exploración de estados
- Restringir el espacio de estados sin afectar el comportamiento
- Comprobar la cobertura del sistema de una exploración
- Utilizar las reglas definidas por el usuario
- Llevar a cabo una exploración aleatoria

6.5.2 Realización de una exploración Bit State

La exploración automática de estado se puede realizar utilizando diferentes algoritmos. El algoritmo llamado de exploración Bit State se puede utilizar de manera eficiente para validar razonablemente grandes sistemas SDL. Se utiliza una estructura de datos llamada tabla *hash* para representar los estados del sistema que se generan durante la exploración.

Una exploración de espacio de estado automática siempre comienza desde el estado actual del sistema. Como queremos explorar el sistema a lo largo completo, primero debemos volver al estado de inicio del árbol de comportamiento.

1. Vaya a la parte superior del árbol haciendo clic en el botón *Top* en el módulo *Explore*.
2. Inicie la exploración Bit State haciendo clic en el botón *Bit-State*. Unos segundos después se abrirá una herramienta llamada *Report Viewer*. Pronto será descrita esta ventana, mientras tanto, simplemente aléjela de la ventana principal.
3. Cuando haya terminado la exploración del sistema a lo largo, algunas estadísticas se imprimen en el área de texto, serán de la siguiente manera:

```
** Starting bit state exploration **
Search depth      : 100
Hash table size  : 1000000 bytes

** Bit state exploration statistics **
No of reports: 1.
Generated states: 15.
Truncated paths: 0.
Unique system states: 14.
Size of hash table: 8000000 (1000000 bytes)
No of bits set in hash table: 28
Collision risk: 0 %
Max depth: 14
Current depth: -1
Min state size: 88
Max state size: 816
Symbol coverage : 32.22
```

Validador SDL

De la información impresa, debe tener en cuenta lo siguiente:

- *Search depth : 100*

La profundidad de búsqueda limita la exploración, es la profundidad máxima o nivel del árbol de comportamiento. Si se alcanza este nivel durante la exploración, la actual trayectoria del árbol se trunca y la exploración continúa en otra rama del árbol. Es posible cambiar la profundidad de búsqueda mediante el establecimiento de una opción en el Validador.

- *No of reports : 1*

La exploración encontró una situación de error. Se examinará este error en el próximo ejercicio.

- *Truncated paths : 0*

Especifica el tiempo que se alcanzó la profundidad máxima.

- *Collision risk : 0 %*

El riesgo de colisiones era muy pequeño en la tabla hash, que es la que se utiliza para representar los estados del sistema generados. Si este valor es mayor que cero, el tamaño de la tabla hash debe ser aumentado mediante el establecimiento de una opción, de lo contrario algunos caminos pueden verse truncados. Esta situación no se producirá en este tutorial.

- *Symbol coverage: 32.22*

Si la cobertura es menor de 100, se considera que no todos los estados se han ejecutado y por tanto la validación no se puede considerar terminada. Se analizará más ampliamente en un apartado próximo.

6.5.3 Verificando Informes

Las situaciones de error mostradas durante una exploración pueden ser examinadas en la ventana *Report Viewer*. Esta ventana muestra el informe en forma de cajas pertenecientes a una estructura de árbol.

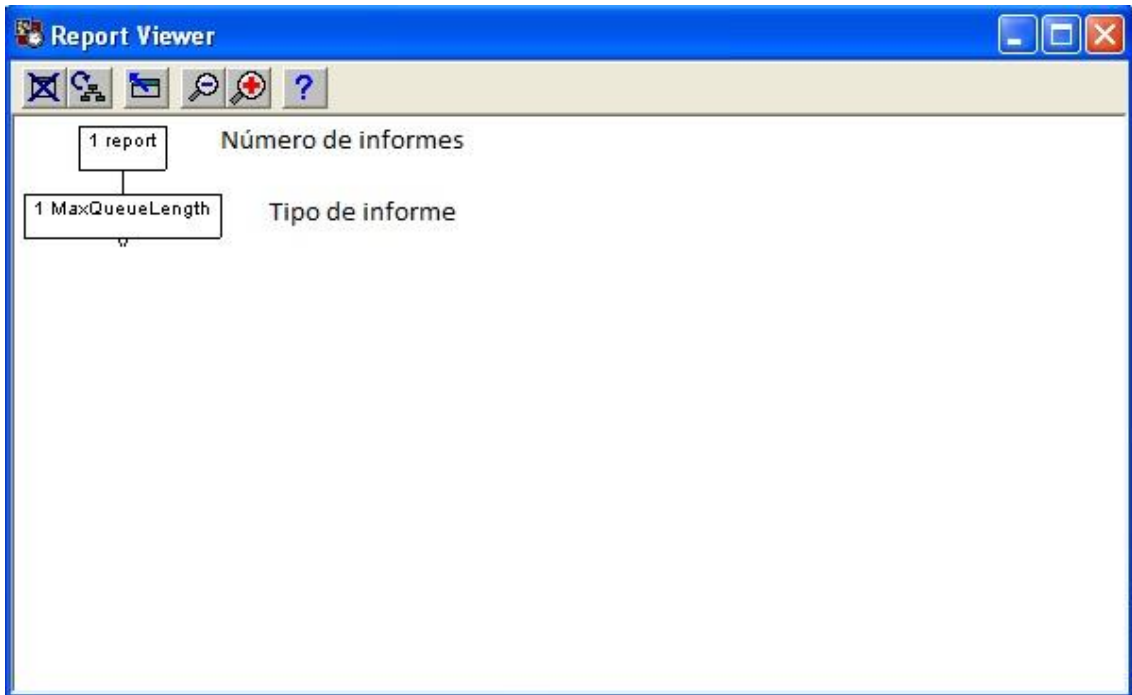


Figura 102: Ventana Report Viewer

- El cuadro superior muestra el número de informes que hay (en este caso sólo uno).
 - En el siguiente nivel del árbol de informes, hay una casilla para cada tipo de informe, indicando el número de informes de ese tipo.
 - En el siguiente nivel, es posible ver los informes reales. Sin embargo, este tipo de árbol está colapsado de forma predeterminada, indicado por el pequeño icono en forma de triángulo que se encuentra debajo de la caja de informes.
1. Para ampliar el informe, haga doble clic en la casilla del informe MaxQueueLength. Aparecerá un cuadro que nos informa del error que

Validador SDL

hemos encontrado antes de forma manual. Además, muestra la profundidad de la situación del error.

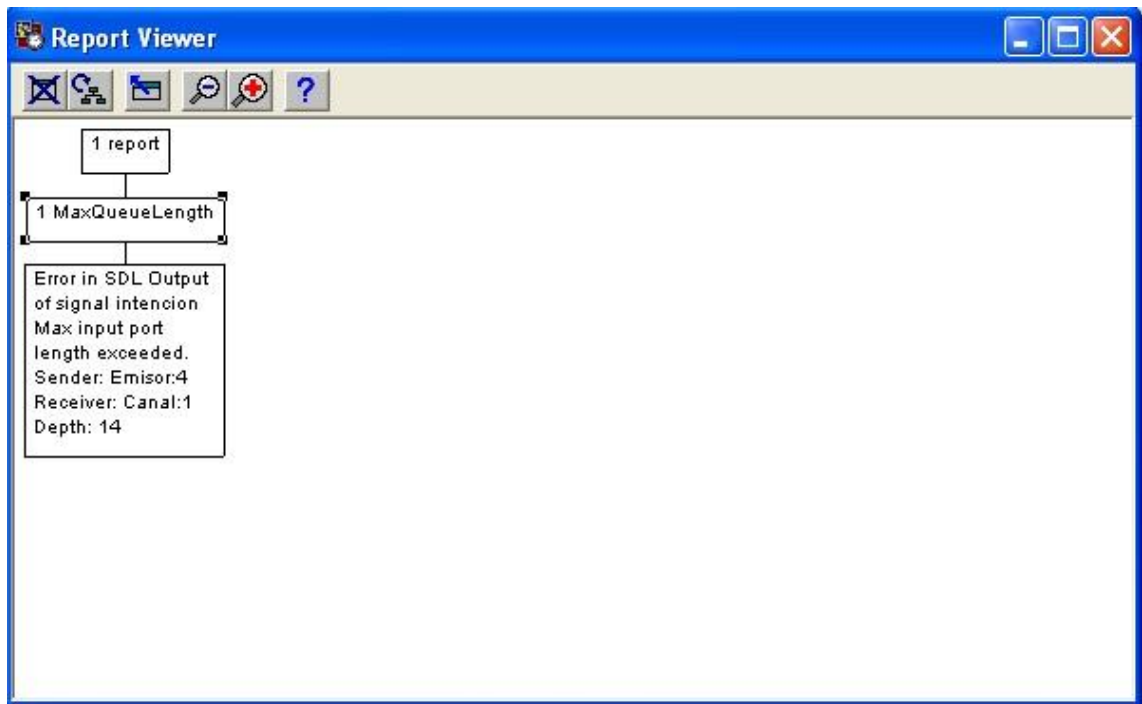


Figura 103: Informe expandido

Si se fija en la ventana *Navigator* y *Command*, se puede ver que el validador se encuentra aún en el estado inicial del sistema, incluso aunque se haya realizado una exploración de estados. A continuación va a ir al estado en el que se ha producido el error.

2. Haga doble clic en el cuadro de informe en *Report Viewer*. Ocurrirán las siguientes cosas:
 - La huella impresa de la situación de error se muestra en el área de texto del Validador y en la ventana *Command*.
 - *Navigator* se mueve hasta el estado de error y lo muestra.
 - Se abre un editor MSC, mostrando el trazo MSC hasta el estado donde se ha producido el error. Como puede ver se ha producido un error al enviar la señal intención.

Validador SDL

Una vez ha utilizado *Report Viewer* para ir a una situación específica, puede moverse fácilmente subiendo y bajando por la ruta de acceso a este estado. Simplemente tiene que utilizar los botones *Up* y *Down* del menú *Explore*, en lugar de hacer doble clic en los nodos de *Navigator*.

6.5.4 Solución del error producido

El mensaje de error indica que se ha excedido el número de puertos en cola. Esto se debe a que el número por defecto establecido por el validador es 3. Puesto que se están manejando 5 nodos, debe aumentarse este valor a 5. Este proceso se realizará más adelante para una correcta validación del Sistema.

Para ello introduzca el comando **Define-Max-Input-Port-Length 5** en la ventana de comandos o acceda a la opción *Input Porth Length* desde el menú *Options1*.

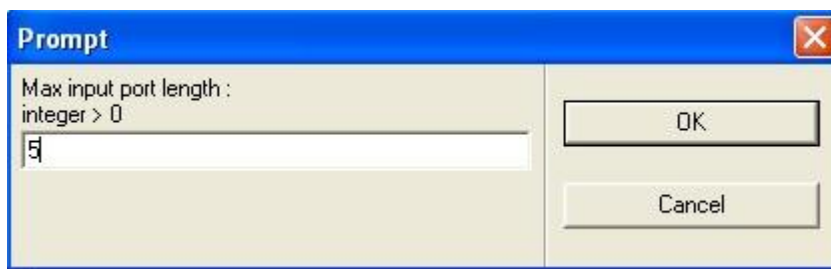


Figura 104: Estableciendo valor puertos de entrada

A continuación, realice una nueva exploración Bit State y compruebe que ocurre.

```
** Bit state exploration statistics **  
No of reports: 0.  
Generated states: 101.  
Truncated paths: 1.  
Unique system states: 101.  
Size of hash table: 8000000 (1000000 bytes)  
No of bits set in hash table: 202  
Collision risk: 0 %  
Max depth: 100  
Current depth: -1  
Min state size: 88  
Max state size: 816  
Symbol coverage : 93.33
```

Figura 105: Exploración Bit-State

Validador SDL

Como se puede ver, la exploración del Sistema es mucho más amplia, sólo ha quedado un 7% del sistema sin explorar. En el próximo apartado se mostrará cómo lograr una exploración del sistema del 100%.

La ventana *Report View* muestra que el error ha sido solucionado.

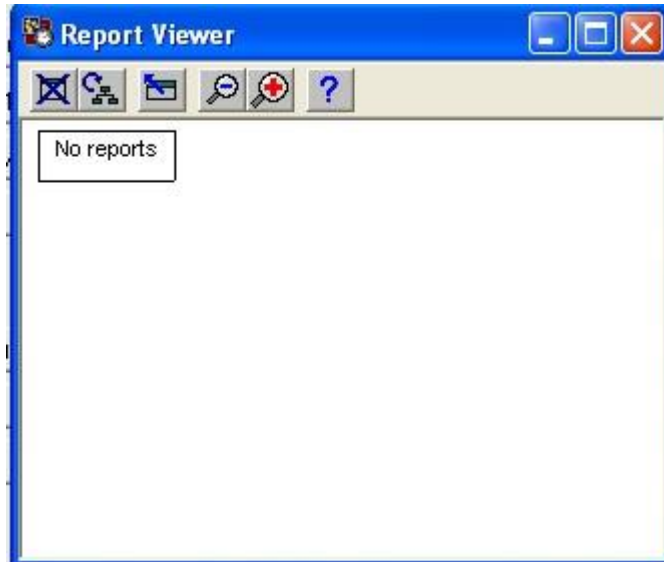


Figura 106: Ventana Report View

6.5.5 Exploración de un espacio de estado más grande

Ahora va a ejecutar un exploración Bit State un poco más avanzada, con una configuración diferente de las opciones de espacio de estado. Esto hará el espacio de estado más grande.

1. Vuelva a la parte superior del árbol pulsando el botón *Top*.
2. En el menú *Options1*, seleccione *Advanced*. Esto establece un número de opciones de espacio de estado disponibles en un solo paso, como se puede ver por los comandos ejecutados en el área de texto.

```
Command : def-sched all
Command : def-prio 1 1 1 1 1
Command : def-max-input-port 2
Max input port length is set to 2.

Command : def-rep-log maxq off
No log for MaxQueueLength reports
```

3. Además vamos a aumentar la profundidad de la búsqueda de la exploración de 100 (el valor predeterminado) a 1000. Desde el menú *Options2*, seleccione *Bit-State : Depth*. En el cuadro de diálogo, escriba 1000 y haga clic en Aceptar.

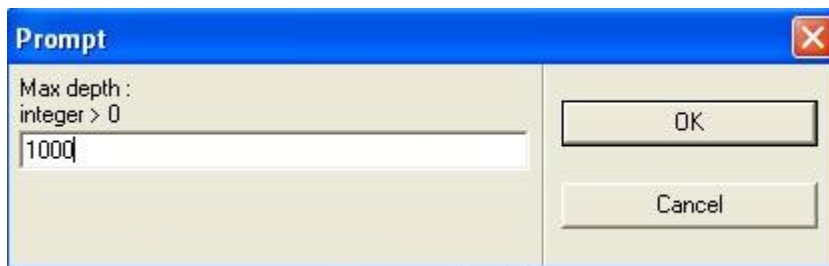


Figura 107: Especificando profundidad 300

Puesto que el árbol de comportamiento se vuelve más grande al variar los valores de estas opciones, la exploración necesitará más tiempo para finalizar. Por lo tanto se mostrará cómo detener la exploración manualmente.

Validador SDL

4. Inicie una nueva exploración, En el área de texto, se imprimirán mensajes de estado. Detenga la exploración después de la aparición de uno de estos mensajes pulsando el botón *Break* en el módulo *Explore*. El área de texto mostrará algo de la siguiente manera:

```
*** Break at user input ***

** Bit state exploration statistics **
No of reports: 0.
Generated states: 924000.
Truncated paths: 29889.
Unique system states: 400959.
Size of hash table: 8000000 (1000000 bytes)
No of bits set in hash table: 739269
Collision risk: 9 %
Max depth: 1000
Current depth: 994
Min state size: 84
Max state size: 776
Symbol coverage : 100.00
```

Figura 108: Cobertura 100%

Como se puede comprobar, la cobertura ha aumentado a un 100%.

5. Continúe la exploración haciendo clic de nuevo en Bit State. Se abrirá una ventana, preguntándole si desea continuar la exploración interrumpida o reiniciarla desde el principio.



Figura 109: Continuando la exploración

6. Seleccione *Continue* y pulse Ok. Espere a que la exploración termine por sí misma.

6.5.6 Restringir el espacio de estado

El validador permite limitar el espacio de estado de varias maneras. A continuación, va a explorar uno de esos métodos, que es muy eficiente en muchos casos. Se realiza mediante el comando Define-Variable-Mode.

Este comando se utiliza para instruir al validador sobre ciertas variables que tiene que ignorar en el momento de contrastar los estados durante la exploración, El modo se puede ajustar para cada variable a “Skip” o “Compare”. La implicación de establecer el modo en “Skip” es que la búsqueda se puede reducir incluso si se encuentra un nuevo estado durante la misma. Esto sucede si la única diferencia entre el nuevo estado y el anterior es que los valores de algunas de las variables omitidas son diferentes.

Ahora va a aplicar esto al sistema aloha. Elija una variable perteneciente al sistema cuyo valor no afecte a su comportamiento. Indique al Validador que ignore esa variable cada vez que realice una búsqueda. En este caso la variable será Sender.

1. Vaya a la parte superior del árbol haciendo clic en Top.
2. Introduzca el comando **define-variable-mode** en la línea de comandos del Validador, y seleccione el proceso Canal en el cuadro de diálogo, la variable Sender aparece en el segundo cuadro de diálogo y la opción “Skip” en el último. Ahora ha dado instrucciones al Validador de ignorar la variable Sender.
3. Inicie una exploración pulsando el botón Bit-State. (
4. Cuando se detenga la búsqueda, compárela con la búsqueda anterior. La única diferencia entre una y otra es que en la segunda se ignora la variable Sender.

Es posible reducir el tiempo necesario de exploración mediante la comprobación de las variables en el sistema. Busque las variables que no tienen ningún impacto en el comportamiento. También busque las variables que no cambian su valor durante la exploración. Como por ejemplo, matrices que se inicializan con la puesta en marcha del sistema pero luego nunca cambian. El modo para estos tipo de variables se debe establecer en “Skip”.

6.5.7 Comprobación de la cobertura de validación

Si *symbol coverage* después de una exploración de estado automático es menor al 100%, *Coverage Viewer* se utiliza para comprobar que partes del sistema no han sido ejecutadas.

1. Vaya a la parte superior del árbol.
2. En primer lugar, debe restaurar el espacio de estado por defecto. Seleccione *Default* desde el menú *Options1*. Tenga en cuenta que *Navigator* cambia de nuevo para una única transición posible desde el nodo superior.
3. Para evitar llegar a todos los estados del sistema, se va a reducir la profundidad de búsqueda de 100 a sólo 10. Use *Bit-State: Depth* en el menú *Options2* y especifique una profundidad máxima de 10.
4. Inicie una exploración Bit State.
5. Para averiguar que partes del sistema aloha no se han alcanzado, abra *Coverage Viewer* desde el menú *Commands*.

Se muestra un árbol de cobertura con todos los símbolos que no se han ejecutado todavía.



6. También puede cambiar a un árbol de transición pulsando el botón de acceso rápido *Tree Mode*.

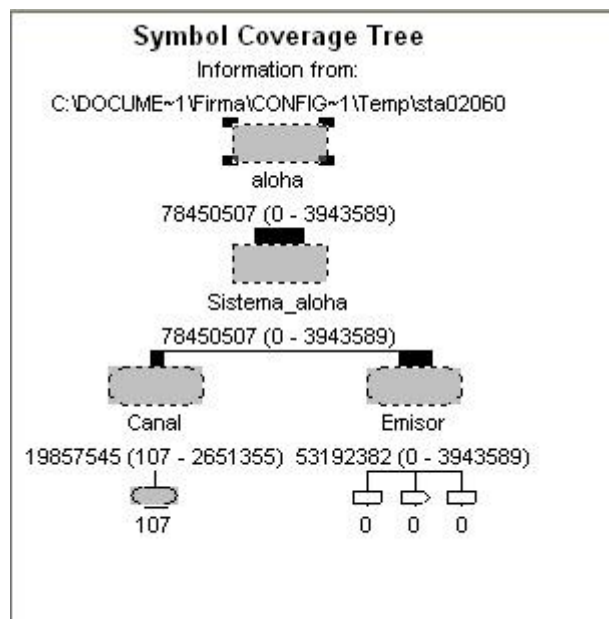


Figura 110: Árbol de cobertura

Puede comprobar las partes del sistema que no han sido exploradas haciendo doble clic en ellas, se abrirá el Editor SDL y le mostrará los símbolos que no han sido explorados.

6.5.8 Ir a un estado mediante reglas definidas por el usuario

Para ir a un estado del sistema en particular, puede usar Navigator manualmente para encontrar el estado estudiando las descripciones de transición y la traza impresa en la ventana de comandos. Esto puede ser tedioso y difícil. En lugar de ello, lo hará de una forma más sencilla, mediante el uso de una regla definida por el usuario.

Al realizar la exploración del espacio de estado, el Validador comprueba un número de reglas predefinidas en cada estado del sistema que se alcanza. Cuando dicha norma se ha cumplido se genera un informe.

En este ejercicio, va a definir una nueva regla para comprobar durante la exploración. La regla se utiliza para encontrar el estado Emisor_vacio en el proceso Emisor.

1. Asegúrese de que todavía se encuentra en la parte superior del árbol.
2. Defina una nueva regla seleccionando *Define Rule* desde el menú *Commands*. En la ventana de diálogo, introduzca la siguiente regla **state (Emisor:1) = Emisor_vacio**.

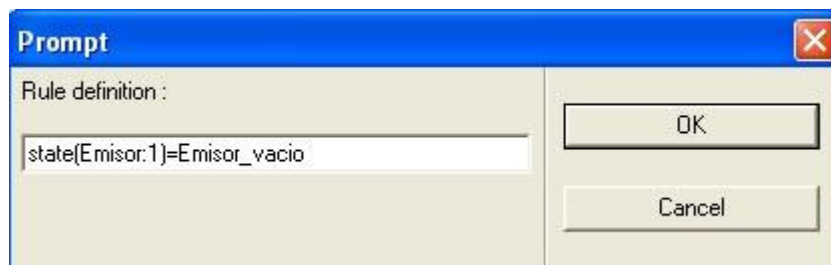


Figura 111: Especificando una nueva regla

Validador SDL

Esta simple regla define que el estado 1 del proceso Emisor será igual a Emisor_vacio.

Mediante la definición de la regla, un informe se generará cuando una exploración de espacio de estado alcance un estado que satisfaga la regla.

3. Inicie una exploración de estado. Puesto que no se ha cambiado ninguna de las opciones desde la última exploración, se imprimirán las mismas estadísticas junto con un informe adicional.
4. Desde *Report Viewer* vaya a la situación en la que la regla se ha cumplido. Ha alcanzado el primer lugar en el árbol de comportamiento en el que se cumple la regla establecida.
5. Ahora indicará al Validador que utilice este estado como la raíz de estado del árbol. Para ello, escriba el comando **define-root** en la línea de entrada de texto y seleccione *Current* en la ventana de diálogo.
6. Ahora podemos cargar las opciones, definir una nueva norma o cargar un MSC, estos nuevos ajustes serán utilizados en todas las nuevas exploraciones basadas en la nueva raíz. También todos los comandos **list/go to path** usarán la ruta de acceso desde la nueva raíz.
7. Antes de continuar, no se olvide de borrar la regla definida por el usuario. Para hacer esto, introduzca el comando **clear-rule** en la línea de entrada de texto.

En este caso, se borrará la regla y comenzará otro tipo de exploración de espacio de estado, paseo aleatorio.

6.5.9 Realizar un paseo aleatorio

Además de la exploración Bit State, hay otro método de exploración conocido como paseo aleatorio. Un paseo aleatorio simplemente explora el comportamiento del árbol seleccionado eligiendo repetidamente un camino al azar por el árbol. Esto es muy útil para los sistemas SDL de gran tamaño.

1. Inicie un paseo aleatorio desde el estado actual haciendo clic en el botón *Random Walk*.
2. Cargue *Coverage Viewer* con la nueva información de cobertura seleccionando *Show Coverage Viewer* desde el menú *Commands*. Cambie a cobertura de transición y muestre todo el árbol. Tenga en cuenta que todas las transiciones se han ejecutado un gran número de veces. Cuando la exploración selecciona una ruta de acceso al azar por el árbol, no existe un mecanismo para evitar que los caminos ya explorados se exploren una vez más. Por lo tanto, la misma transición se puede ejecutar cualquier número de veces.
3. Salga de *Coverage Viewer* desde el menú *File*.
4. Restablezca el sistema seleccionando *Reset* en el menú *Options1*. Ha vuelto a la parte superior del árbol, y la raíz del árbol vuelve a ser la original, el estado de inicio del sistema.

6.6 Verificación de un MSC

Otra importante área de uso para un validador es verificar una Secuencia de Mensajes. Para verificar un MSC se debe comprobar si hay una ruta de ejecución posible para el sistema SDL que satisface el MSC. Esto se realiza mediante la carga de MSC y la realización de una exploración de estado creada de una forma adecuada para la verificación del MSC.

6.6.1 Qué va a aprender

- A verificar un MSC

6.6.2 Verificar un Sistema MSC

En este ejercicio, se va a verificar un MSC realizado en el nivel del sistema, es decir, un MSC que sólo define las señales desde y hacia el medio. El nombre de archivo del MSC es `SystemLevel.msc` y se encuentra en el mismo directorio que los archivos restantes para el ejemplo aloha. El MSC se muestra en la siguiente figura:

1. Reinicie el Sistema. Hágalo seleccionando *Restart* en el menú *File*. Seleccione “No” si le pide guardar opciones. El comando *Restart* finalizará el Validador y lo iniciará de nuevo.
2. Inicie una verificación MSC haciendo clic en el botón *Verify MSC*. Una ventana de selección de archivo se abre para que seleccione el MSC a verificar.
3. Seleccione el archivo MSC que desee verificar, recuerde que el archivo estará nombrado de la siguiente manera “nombre.msc” y haga clic en OK. Se inicia una exploración, guiada por el MSC que hemos cargado.

En las estadísticas impresas, tenga en cuenta que la exploración se completa sin las trayectorias truncadas. Esto es debido a que el MSC cargado restringe el árbol de comportamiento, sólo las partes que tratan

Validador SDL

de eventos en MSC son ejecutadas. La profundidad máxima de la misma es 20.

Aparecerá una línea que indica si el MSC ha sido verificado o no.

En este caso, si el MSC se verificó. El comportamiento descrito en el MSC era de hecho posible. En *Report Viewer* puede comprobar si se ha producido alguna violación. La existencia de estados que verifican el MSC determina el resultado de la verificación.

4. Una traza MSC en el Validador se hace siempre en el nivel de proceso.

6.6.3 Saliendo de la interfaz de usuario Validador

Cierre las ventanas del Validador de la siguiente manera:



1. Para cerrar *Navigator* y *Report Viewer*, haga clic en el botón de acceso rápido *Close* que encontrará en dichas ventanas.
2. Para cerrar la ventana *Command*, seleccione *Close* desde el menú *File*.
3. Salga del validador desde el menú *File*. Se abrirá un cuadro de diálogo que le preguntará si quiere guardar los cambios realizados a las opciones del Validador.



Figura 112: Guardando opciones modificadas

4. Si selecciona *Yes* y pulsa *Ok*, las opciones se guardarán en un archivo llamado **valinit.com**. Este archivo se lee cada vez que el Validador se inicia desde el mismo directorio, o cuando el Validador se reinicia o reinicia Validador UI. Seleccione *No* y haga clic en *OK*.

6.7 Hasta el momento...

Con la práctica de éste y de los tutoriales anteriores, ha aprendido lo básico sobre la Suite SDL y espero que haya disfrutado del “trayecto”. A continuación, se va a realizar el cálculo de ciertas variables pertenecientes al Sistema.

7. CÁLCULO DE VARIABLES

En apartados anteriores ha aprendido las funciones y usos fundamentales tanto del Validador SDL como del Simulador. En este apartado, va a usar esos conocimientos adquiridos para obtener los valores de ciertas variables pertenecientes al sistema así como realizar comparativas entre ellos.

Para ello, se realizará una validación completa del Sistema y posteriormente se realizará una simulación paso a paso para obtener el valor de las distintas variables.

7.1 Qué va a aprender

- A realizar una validación completa del sistema
- Simular mediante el terminal MS-DOS de Windows
- Utilizar un script para obtener las diferentes variables del sistema
- Realizar una comparativa entre las variables obtenidas

7.2 Validación completa del sistema

Para realizar la Validación completa del sistema aloha se procederá como en apartados anteriores.

1. En la ventana *Organizer*, seleccione el Sistema aloha.
2. Desde el menú *Generate*, seleccione la opción *Make*.

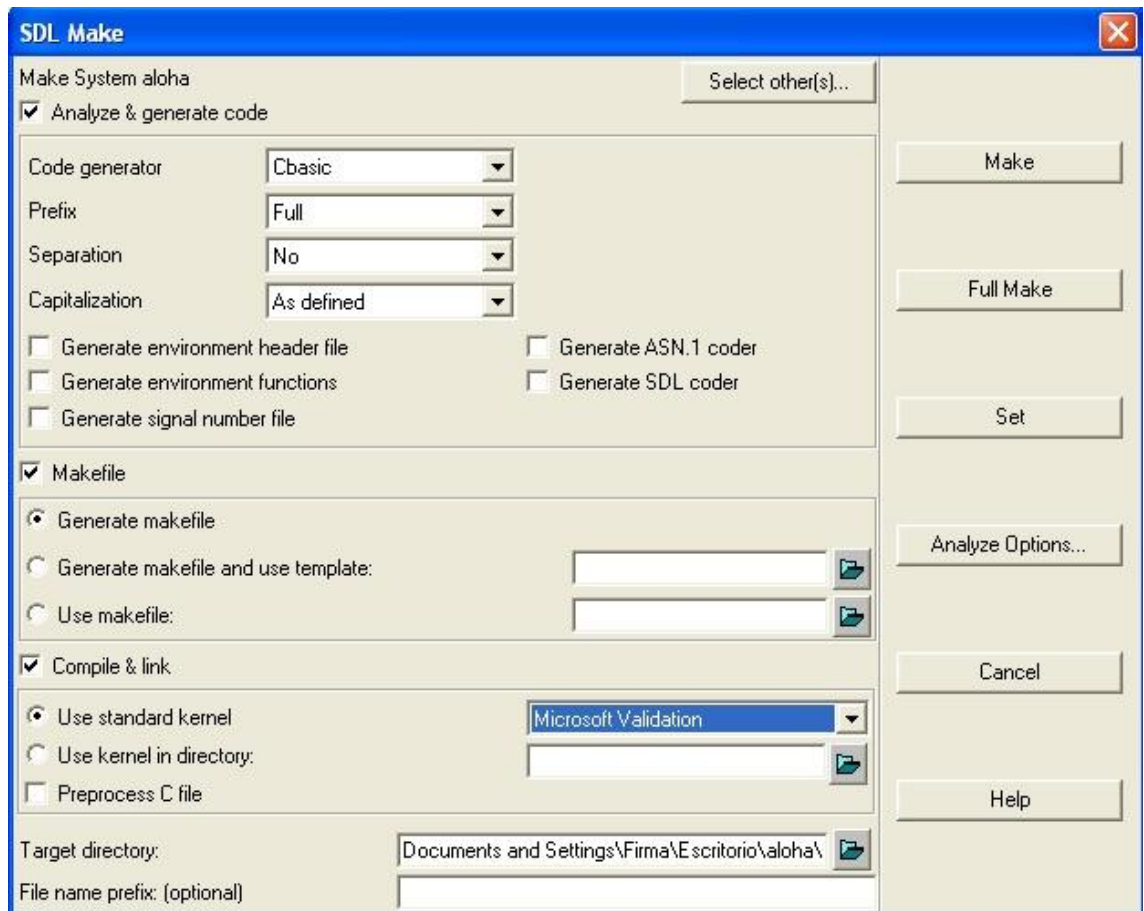


Figura 113: Ventana Make

Asegúrese de que se encuentra marcada la opción *Microsoft Validation*.

Cálculo de Variables

3. Haga clic en la opción *Full Make* y espere hasta que termine la Validación.
4. En la barra de estado de la ventana *Organizer* debe aparecer lo siguiente:

Analyzer done: No errors and no warnings



5. A continuación, haga clic en el símbolo de acceso rápido del Validador. Se desplegará la interfaz de usuario del Validador, tal y como se ha visto en apartados anteriores.

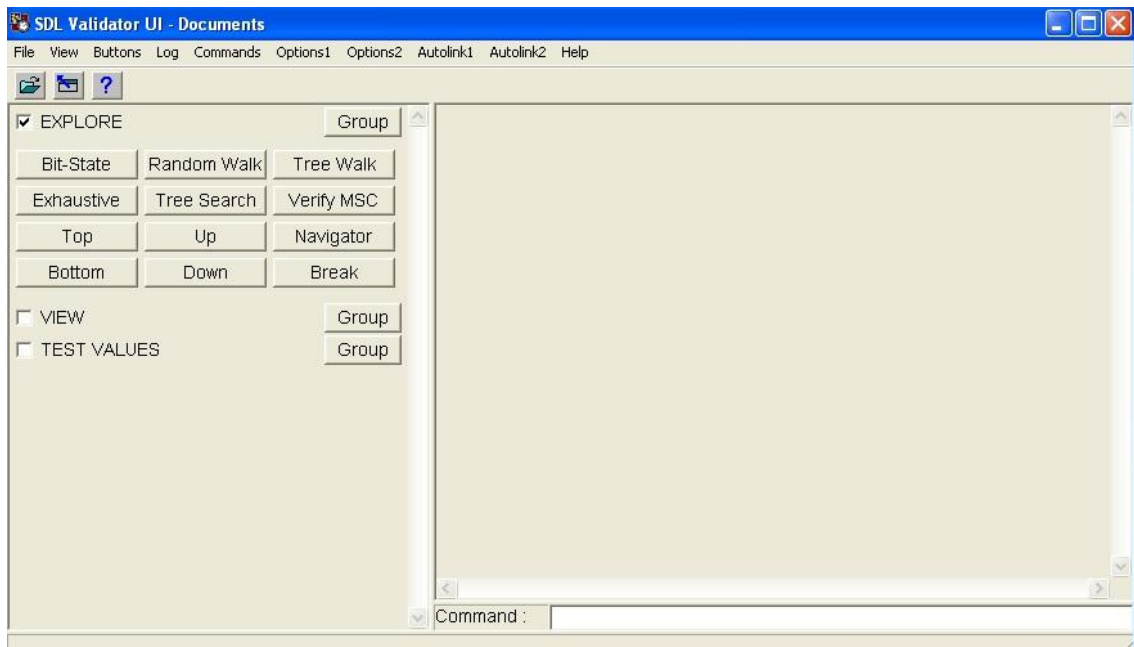


Figura 114: Ventana del Validador

6. Ahora debe abrir el ejecutable que se ha creado anteriormente, para ello vaya al menú *File*, seleccione *Open* y en la ventana de diálogo elija el ejecutable **aloha_vlc**.
7. Puesto que nuestro sistema contiene variables externas (*N_nodos* y *PG*) se abrirá una ventana de diálogo para que introduzcamos o bien un fichero que contenga los valores de dichas variables (por ejemplo puede crear un archivo llamado *siminit.syn* que contenga los valores *N_nodos= 5*, *PG =0.2*), o directamente introducirlos manualmente.

Cálculo de Variables

- Posteriormente, se analizará el sistema, para ello pulse el botón *Random Walk* desde el módulo *Explore*.
- Al terminar la exploración *Random Walk*, se puede apreciar que el sistema apenas ha sido ejecutado, pues la cobertura que nos marca es muy baja, de 34.44.

```
** Random walk statistics **  
No of reports: 1  
Gen states   : 1400  
Max depth    : 13  
Min depth    : 13  
Symbol coverage : 34.44
```

También se despliega la ventana *Report Viewer* indicando que se ha producido un error en el número máximo de nodos en cola.

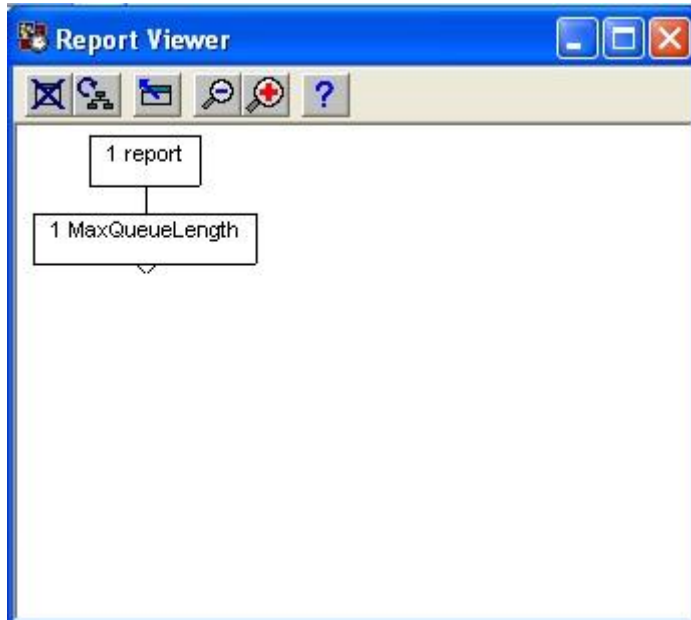


Figura 115: Ventana Report Viewer

Este error se produce debido a que por defecto, el número de nodos en cola viene fijado en 3, y puesto que se está utilizando un sistema de 5 nodos, este valor debe ser cambiado.

Para ello vaya al menú *Options1*, y cambie el valor a 5 en *Input Port Length*; o simplemente introduzca el comando **Define-Max-Input-Port-Length 5** en la línea de entrada de texto.

Cálculo de Variables

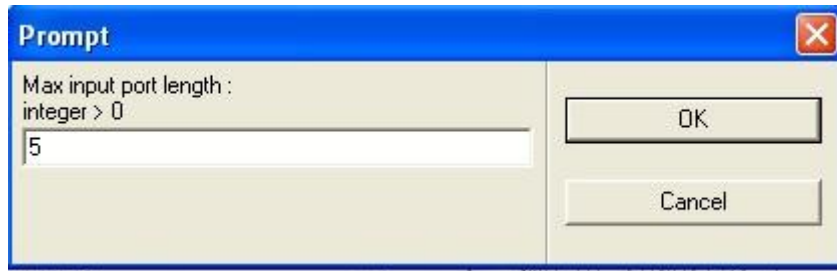


Figura 116: Input Port Length

10. Pulse de nuevo el botón *Random Walk*. Se puede apreciar que la cobertura ha aumentado a un 96.67. Y la ventana *Report Viewer* ya no muestra ningún error.

```
** Random walk statistics **  
No of reports: 0  
Gen states   : 12720  
Max depth   : 100  
Min depth   : 100  
Symbol coverage : 96.67
```

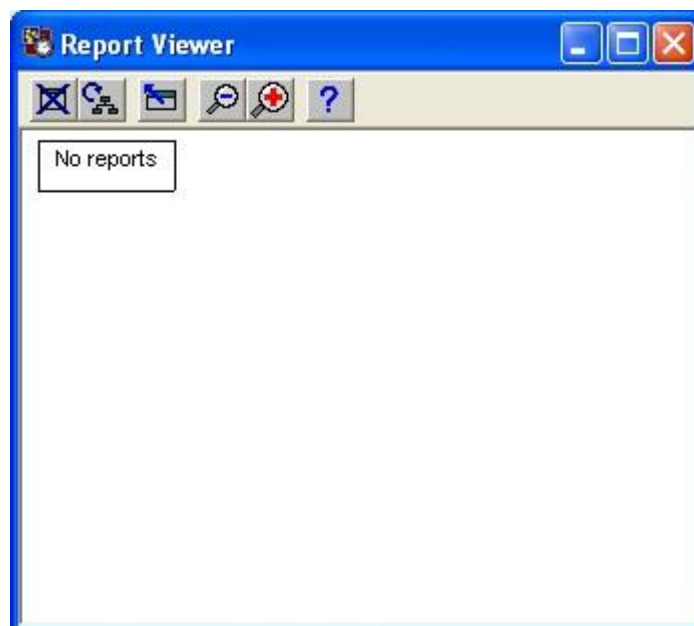


Figura 117: Report Viewer sin reportes

Cálculo de Variables

11. Que el nivel de cobertura sea de un 96.67% significa que un hay un 3% de símbolos que no se han ejecutado. Puede comprobar cuales de estos símbolos son los que no han llegado a ejecutarse, accediendo al menú *Commands* y pulsando la opción *Show Coverage Viewer*.

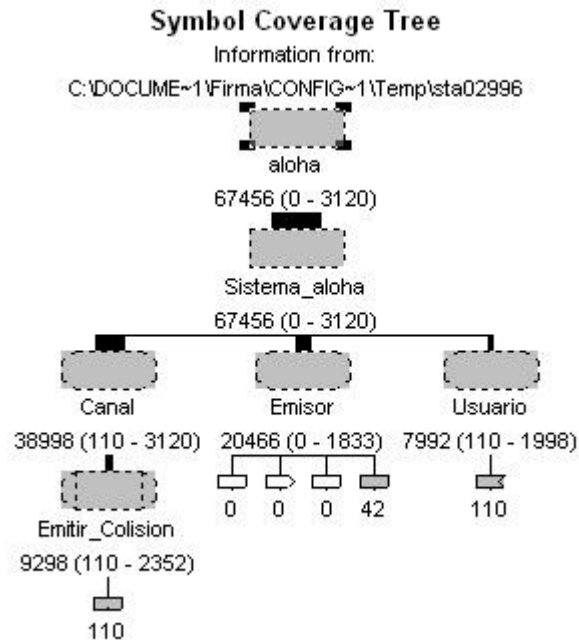


Figura 118: árbol de cobertura.

Haciendo doble clic en los símbolos que no se han ejecutado, se abrirá el Editor SDL y podrá comprobar específicamente que símbolos son.

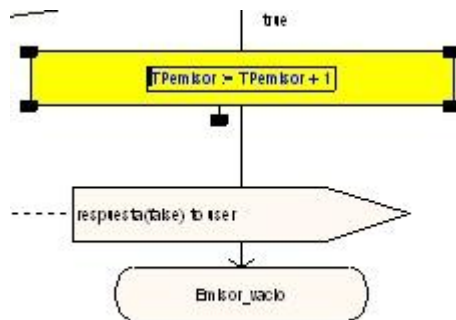


Figura 119: Símbolos que no han sido ejecutados.

Cálculo de Variables

12. Para realizar una validación completa se necesita que el nivel de cobertura sea del 100%. Por lo que aumentaremos la profundidad de *Random Walk*, que por defecto se encuentra establecida en 100.
13. Desde el menú *Options 2*, seleccione la opción *Random Walk: Depth* e introduzca el valor **1000**. Vuelva a pulsar el botón *Random Walk* y compruebe que ahora si se ha ejecutado todo el sistema.

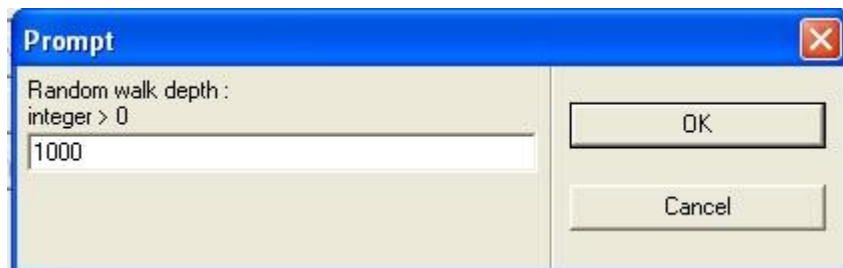


Figura 120: Aumentando la profundidad

```
** Random walk statistics **  
No of reports: 0  
Gen states   : 115397  
Max depth    : 1000  
Min depth    : 1000  
Symbol coverage : 100.00
```

Figura 121: Nivel de cobertura del 100%

14. Salga del Validador haciendo clic en *File, Exit*. Cuando le pregunte si desea guardar los cambios realizados, pulsé la opción YES.
15. Ya se ha realizado una Validación completa del Sistema.

7.3 Simulación del Sistema

Una vez realizada correctamente la validación, se pasará a realizar la Simulación. Para ello acceda al Simulador UI tal y como se ha visto en apartados anteriores.

1. Cree un ejecutable accediendo a la opción *Make* desde el menú *Generate*, debe tener en cuenta que debe encontrarse seleccionada la opción **Microsoft Simulation** y haga clic en *Full Make*.



2. Acceda a la interfaz de usuario del Simulador haciendo clic en el botón de acceso rápido *Simulate*.
3. Una vez en el Simulador, vaya al menú *File*, haga clic en *Open* y abra el ejecutable que se acaba de crear, **aloha_smc**. Introduzca las variables externas $N_{\text{nodos}}=5$ y $PG=0.95$.

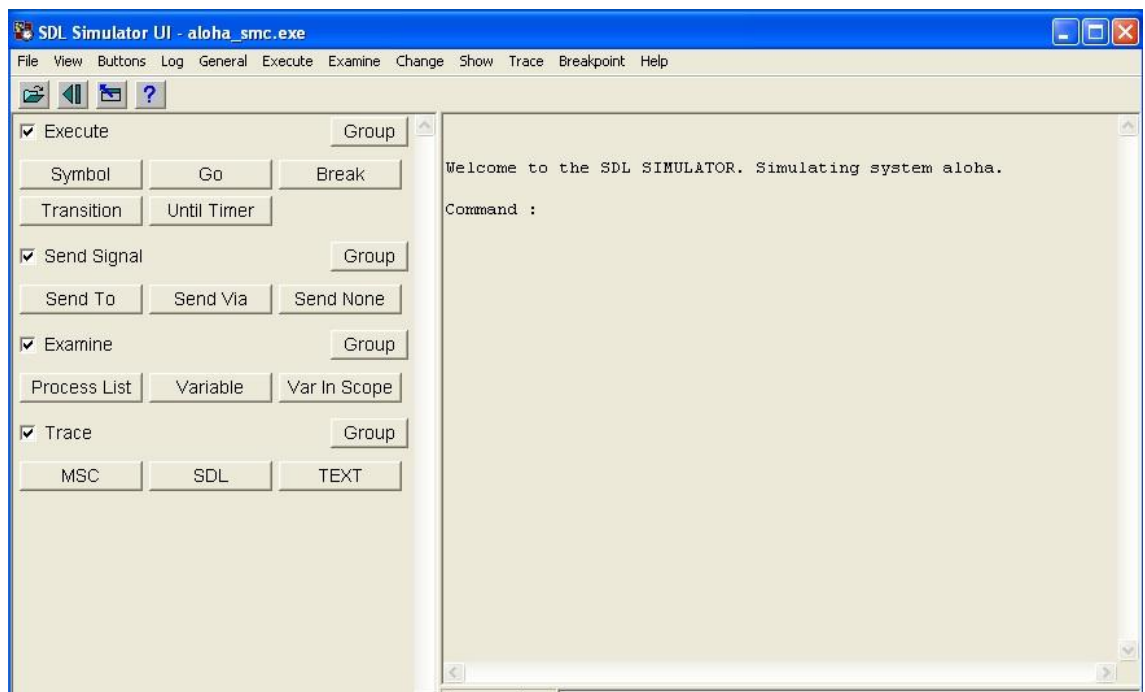


Figura 122: Interfaz de usuario del Simulador

Cálculo de Variables

4. Si se desea ver en pantalla como se produce la ejecución del sistema aloha mediante MSC, introduzca el comando **Start-Interactive-MSC-Log 1**. (La opción 1 simboliza “MSC with States”). Se abrirá un editor de MSC.

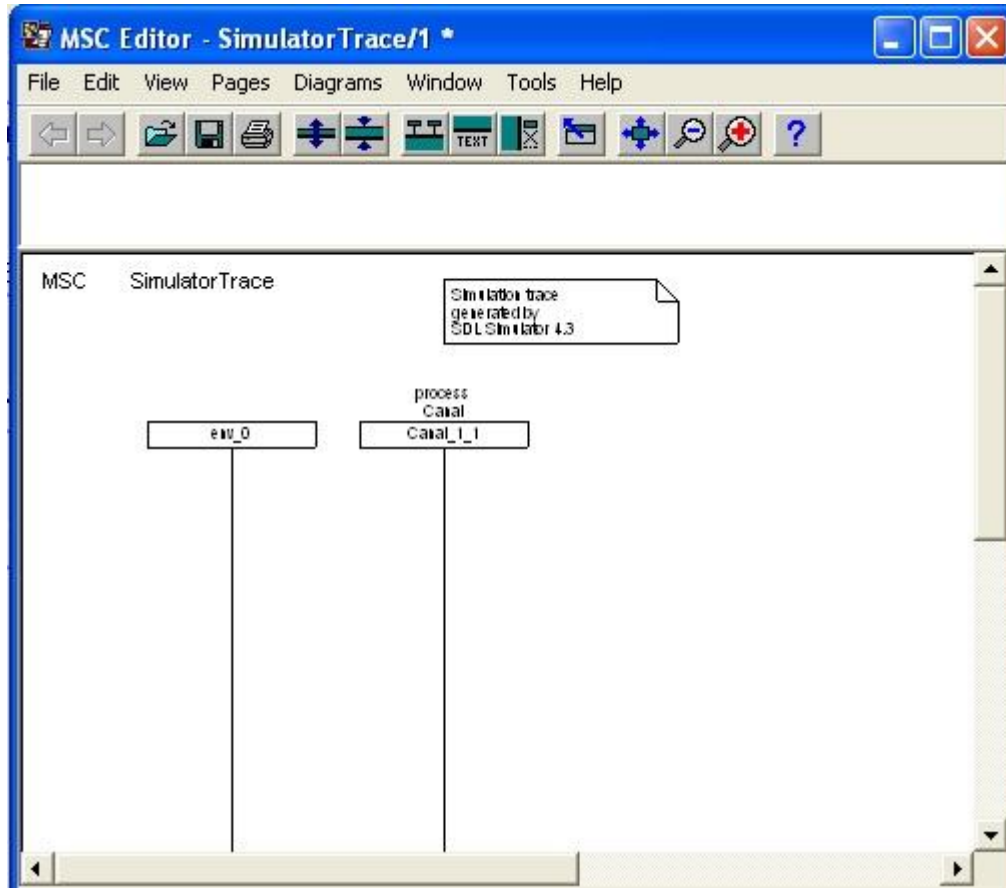


Figura 123: Editor MSC

5. Haga clic en el botón *GO* desde el módulo *Execute*. Compruebe como se va dibujando la ejecución del sistema aloha en el editor MSC. Si se pulsa el botón *Break*, se interrumpirá la simulación y podrá observar si el comportamiento del sistema es el adecuado. Una vez comprobado que tiene buen aspecto, cierre el editor de MSC sin guardar.

7.4 Cálculo de la curva S-G

Situándose de nuevo en el Simulador, puede examinar el valor de las distintas variables del sistema.

1. Pulse el botón *Variable* dentro del menú *Examine*. Seleccione el proceso Emisor, la opción 1 y la variable *TOemisor*.

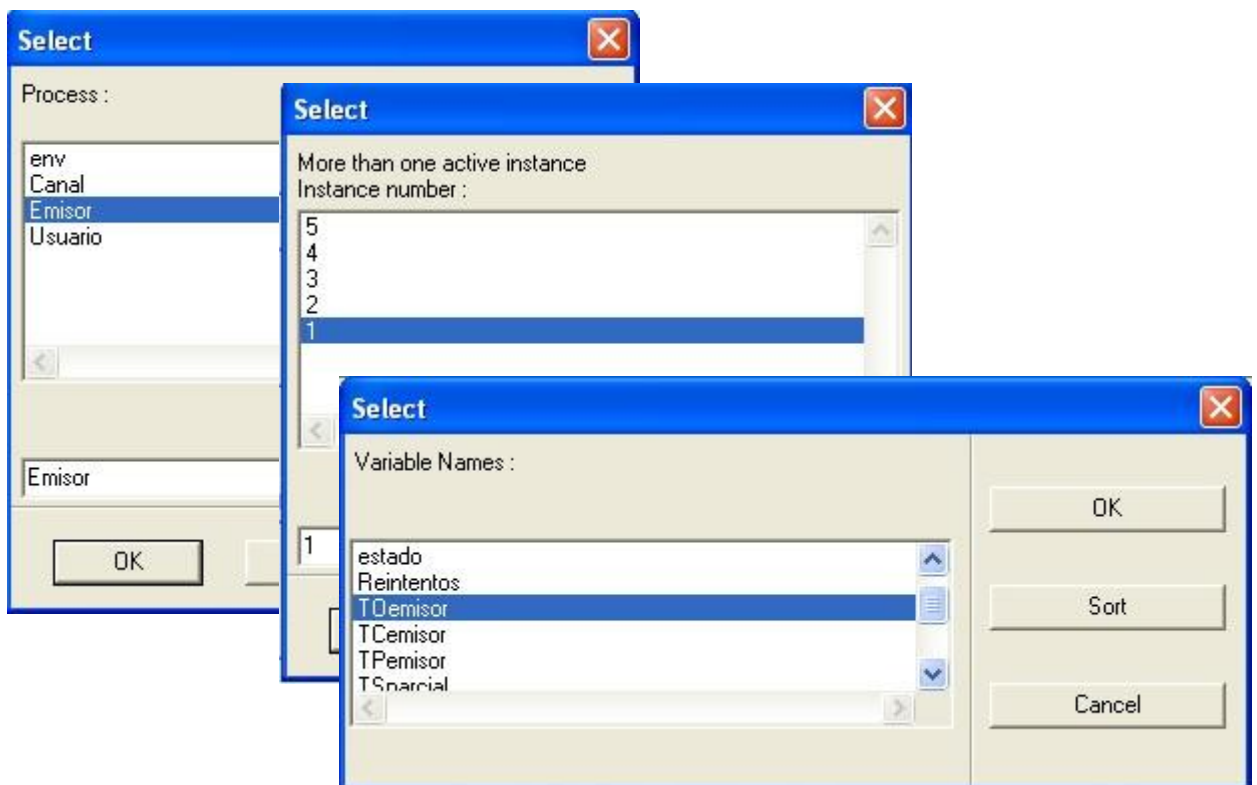


Figura 123: Obteniendo valor de la variable *TOemisor*

Devuelve *TOemisor (integer) =4*. Significa que la variable *TOemisor* de la instancia 1 del proceso Emisor vale 4.

2. Se puede observar también la variable *TCemisor*. Puede realizarse introduciendo el comando **Examine-Variable (Emisor 1 TCemisor)**.

En este caso se obtiene **TCemisor (integer) = 0**.

Cálculo de Variables

Significa que la variable *TCemisor* de la instancia 1 del proceso Emisor vale 0.
Proceda a obtener los valores de las variables de los 5 emisores:

Command: Examine-Variable(Emisor 1 TOemisor)
TOemisor (integer) =4

Command: Examine-Variable(Emisor 2 TOemisor)
TOemisor (integer) =5

Command: Examine-Variable(Emisor 3 TOemisor)
TOemisor (integer) =2

Command: Examine-Variable(Emisor 4 TOemisor)
TOemisor (integer) =6

Command: Examine-Variable(Emisor 5 TOemisor)
TOemisor (integer) =4

Command: Examine-Variable(Emisor 1 TCemisor)
TOemisor (integer) =0

Command: Examine-Variable(Emisor 2 TCemisor)
TOemisor (integer) =1

Command: Examine-Variable(Emisor 3 TCemisor)
TOemisor (integer) =0

Command: Examine-Variable(Emisor 4 TCemisor)
TOemisor (integer) =2

Command: Examine-Variable(Emisor 5 TCemisor)
TOemisor (integer) =0

El valor de las variables obtenidas es tan bajo debido a que sólo han sido ejecutadas 10 ranuras.

El número de ranuras se puede comprobar introduciendo el comando **Now** en la línea de entrada de texto.

Obtendrá *Now = 10.000*.

Cálculo de Variables

3. De la misma forma, se pueden obtener los valores de las variables S y G del canal.

Command: Examine-Variable (Canal G)

G (integer) = 21.

Command: Examine-Variable (Canal S)

S (integer) = 3.

Representan las ranuras ofrecidas y las cursadas globales en 10 ranuras.

Si realiza la suma para todos los emisores, debe coincidir con las globales.

$$4+5+2+6+4= 21$$

$$0+1+0+2+0= 3$$

Se deben realizar los cálculos de las variables para todos los valores de PG, para ello se aconseja tomar 10 puntos entre 0 y 0.2, y entre 0.2 y 1. Como por ejemplo:

| | | | | | | | | | |
|------|------|-------|------|-------|-----|-------|-------|-------|-----|
| 0 | 0.01 | 0.025 | 0.05 | 0.075 | 0.1 | 0.125 | 0.150 | 0.175 | 0.2 |
| 0.25 | 0.3 | 0.35 | 0.4 | 0.45 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |

4. La simulación con el valor de PG = 0 debe dar todo 0 (si no es el caso, la simulación se considera incorrecta).
5. Puesto que realizar las simulaciones con la interfaz de usuario del Simulador (Simulator UI) resulta un poco pesado, se van a automatizar un poco.

7.5 Utilización del terminal MS-DOS de Windows

1. Abra un terminal MS-DOS desde el menú de Inicio de Windows, pulse Ejecutar e introduzca **cmd**.

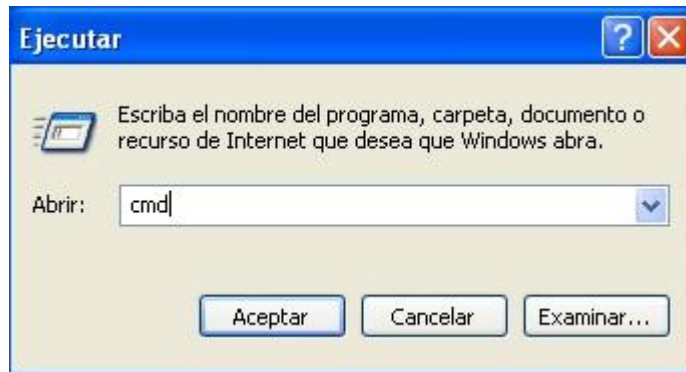


Figura 124: Abriendo un terminal MS-DOS

2. Se abrirá la ventana MS-DOS.

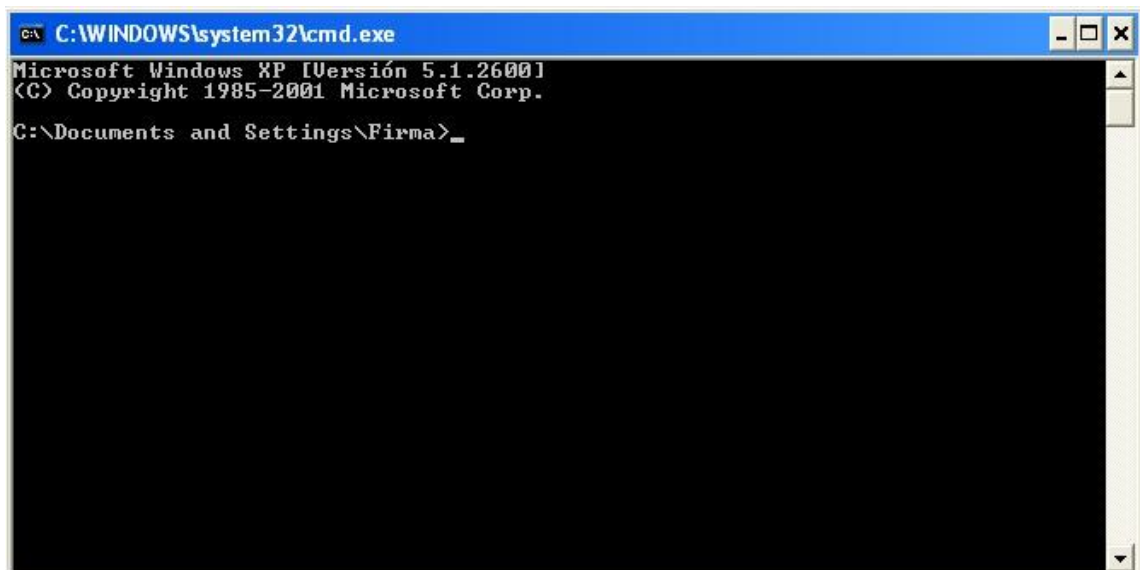


Figura 125: Ventana MS-DOS de Windows

3. Introduzca el comando **cd C:\Documents and Settings\Firma\Escritorio\aloha**. (Introduzca la ruta completa de donde se haya copiado).

Y ejecute **aloha_smc.exe**

Cálculo de Variables

C:\Documents and Settings\Firma\Escritorio\aloha\aloha_smc.exe

El simulador responde de la siguiente manera:

```
C:\Documents and Settings\Firma\Escritorio\aloha>aloha_smc.exe
Connected with the Postmaster.
External synonym file : _
```

Está solicitando el fichero donde se encuentra el valor de las variables externas, por el momento no se va a introducir, por lo que pulse <Enter> y le pedirá que introduzca manualmente las variables externas.

Synonym N_nodos (integer) = 5

Synonym PG (integer) = 0.95

Una vez introducidas las variables, aparecerá el mensaje de bienvenida del Simulador.

```
Welcome to the SDL SIMULATOR. Simulating system aloha.
Command : _
```

4. Para ejecutar 10 ranuras, introduzca el comando: **Proceed-Until 10.**

Se ejecutará el sistema hasta alcanzar el instante 10, y ahora puede obtener el valor de las variables de la misma forma en la que se ha realizado anteriormente.

Command: Examine-Variable(Emisor 1 TOemisor)
TOemisor (integer) =4

Command: Examine-Variable(Emisor 2 TOemisor)
TOemisor (integer) =5

Command: Examine-Variable(Emisor 3 TOemisor)
TOemisor (integer) =2

Command: Examine-Variable(Emisor 4 TOemisor)
TOemisor (integer) =6

Cálculo de Variables

Command: Examine-Variable(Emisor 5 TOemisor)
TOemisor (integer) =4

Command: Examine-Variable(Emisor 1 TCemisor)
TOemisor (integer) =0

Command: Examine-Variable(Emisor 2 TCemisor)
TOemisor (integer) =1

Command: Examine-Variable(Emisor 3 TCemisor)
TOemisor (integer) =0

Command: Examine-Variable(Emisor 4 TCemisor)
TOemisor (integer) =2

Command: Examine-Variable(Emisor 5 TCemisor)
TOemisor (integer) =0

Command: Examine-Variable (Canal G)
G (integer) = 21.

Command: Examine-Variable (Canal S)
S (integer) = 3.

Acaba de utilizarse el Simulador sin el entorno gráfico de la misma forma que se ha realizado anteriormente con el entorno gráfico.

7.6 Automatizar el proceso de Simulación

Para automatizar el proceso de Simulación, se pondrán las variables externas en un fichero (en este caso le llamaremos `siminit.syn`). El simulador, cuando arranque, buscará la variable de entorno `SDTEXTSYNFILE`.

1. Salga del Simulador ejecutando el comando **quit**. La ventana MS-DOS imprimirá el mensaje "Simulation terminated".
2. Desde el terminal MS-DOS ejecute el comando `set` (aparecerán las variables de entorno). Si introduce el comando **set SDTEXTSYNFILE = siminit.syn** conseguirá que la variable de entorno que leerá el Simulador (`SDTEXTSYNFILE`) apunte al fichero `siminit.syn`.
3. Para editar el fichero `siminit.syn` habrá un editor de texto e introduzca las líneas:

```
N_nodos 5  
PG 0.95
```

Guarde los cambios y ejecute el simulador de nuevo.

```
Connected with the Postmaster.  
Using external synonym file "siminit.syn" according to environment variable "SDT  
EXTSYNFILE".  
  
Welcome to the SDL SIMULATOR. Simulating system aloha.  
Command :
```

De esta forma, ya no es necesario introducir manualmente los valores de las variables externas `N_nodos` y `PG`.

4. Salga de nuevo del Simulador introduciendo el comando **quit**.
5. A continuación introducirá los comandos al Simulador. El simulador aceptará una serie de comandos, provenientes del archivo `siminit.com` Para ello edite un archivo llamado `siminit.com` con los siguientes comandos:

Proceed-Until 10

Examine-Variable (Canal S)

Examine-Variable (Canal G)

Examine-Variable (Emisor 1 TOemisor

Examine-Variable (Emisor 2 TOemisor

Examine-Variable (Emisor 3 TOemisor

Examine-Variable (Emisor 4 TOemisor

Examine-Variable (Emisor 5 TOemisor

Examine-Variable (Emisor 1 TCemisor

Examine-Variable (Emisor 2 TCemisor

Examine-Variable (Emisor 3 TCemisor

Examine-Variable (Emisor 4 TCemisor

Examine-Variable (Emisor 5 TCemisor

Exit

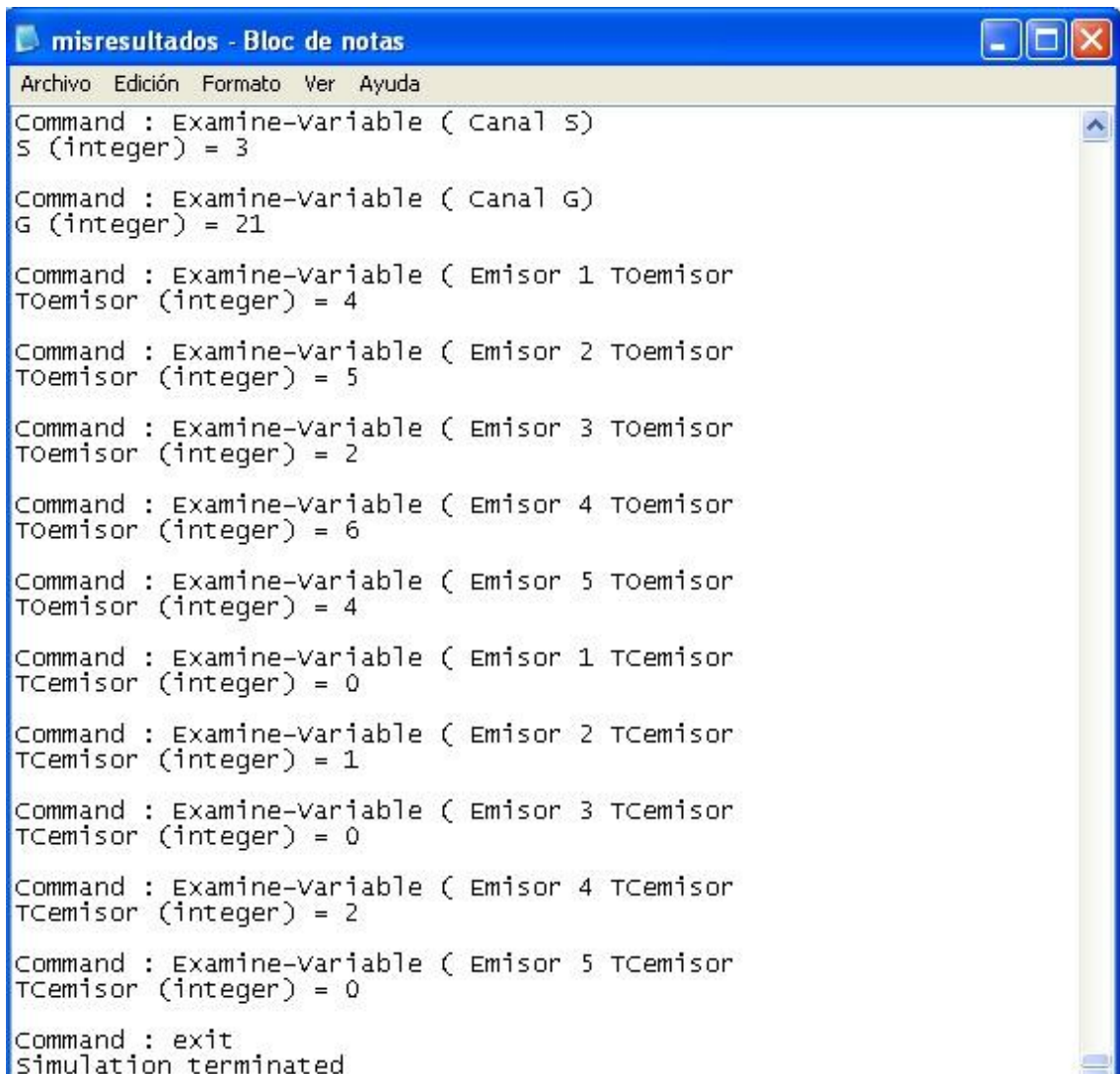
Cálculo de Variables

- Una vez creado el archivo `siminit.com`, ejecute de nuevo el simulador **aloha_smc.exe**. Como se puede observar el simulador ya no preguntará nada, tomará `N_nodos` y `PG` desde el archivo `siminit.syn` y ejecutará los comandos de `siminit.com`. El último comando, **exit**, sale del Simulador.

Los resultados pueden observarse en pantalla. Para guardarlos en un fichero para posterior análisis, puede usar la redirección de MS-DOS.

aloha_smc.exe > misresultados.txt

De esta forma, justo al final del archivo `misresultados.txt` podremos ver los valores de las variables que nos interesan. Con dichos valores ya se podría representar un punto de la curva S-G.



```
misresultados - Bloc de notas
Archivo Edición Formato Ver Ayuda
Command : Examine-Variable ( Canal S)
S (integer) = 3
Command : Examine-Variable ( Canal G)
G (integer) = 21
Command : Examine-Variable ( Emisor 1 TOemisor
TOemisor (integer) = 4
Command : Examine-Variable ( Emisor 2 TOemisor
TOemisor (integer) = 5
Command : Examine-Variable ( Emisor 3 TOemisor
TOemisor (integer) = 2
Command : Examine-Variable ( Emisor 4 TOemisor
TOemisor (integer) = 6
Command : Examine-Variable ( Emisor 5 TOemisor
TOemisor (integer) = 4
Command : Examine-Variable ( Emisor 1 TCemisor
TCemisor (integer) = 0
Command : Examine-Variable ( Emisor 2 TCemisor
TCemisor (integer) = 1
Command : Examine-Variable ( Emisor 3 TCemisor
TCemisor (integer) = 0
Command : Examine-Variable ( Emisor 4 TCemisor
TCemisor (integer) = 2
Command : Examine-Variable ( Emisor 5 TCemisor
TCemisor (integer) = 0
Command : exit
Simulation terminated
```

Figura 126: Archivo `misresultados.txt`

7.7 Simulación para todos los valores de PG

Una vez aprendido el cálculo de variables para un valor determinado de PG, se debe repetir para todos los valores de PG. Para ello, se creará un script que automatizará el proceso.

1. Cree el archivo por lotes de MS-DOS **simular.bat** y edítelo con las siguientes instrucciones:

```
@echo off
set SDTEXTSYNFILE=siminit.syn

for %%i in (0 01 025 05 075 1 125 150 175 2 25 3
35 4 45 5 6 7 8 9
95) do (
echo PG=0.%%i en fichero: resultado_0_%%i.txt
echo N_nodos 5 > siminit.syn
echo PG 0.%%i >> siminit.syn
aloha_smc.exe > resultado_0_%%i.txt
)
```

Para evitar que el tamaño de los ficheros de resultados crezca mucho se pueden eliminar las trazas desde el simulador (set trace 0).

El fichero siminit.com finalmente será:

Set-Trace 0

Proceed-Until 10000

Examine-Variable (Canal S)
Examine-Variable (Canal G)
Examine-Variable (Emisor 1 TOemisor
Examine-Variable (Emisor 2 TOemisor
Examine-Variable (Emisor 3 TOemisor
Examine-Variable (Emisor 4 TOemisor
Examine-Variable (Emisor 5 TOemisor
Examine-Variable (Emisor 1 TCemisor
Examine-Variable (Emisor 2 TCemisor
Examine-Variable (Emisor 3 TCemisor
Examine-Variable (Emisor 4 TCemisor
Examine-Variable (Emisor 5 TCemisor
Exit

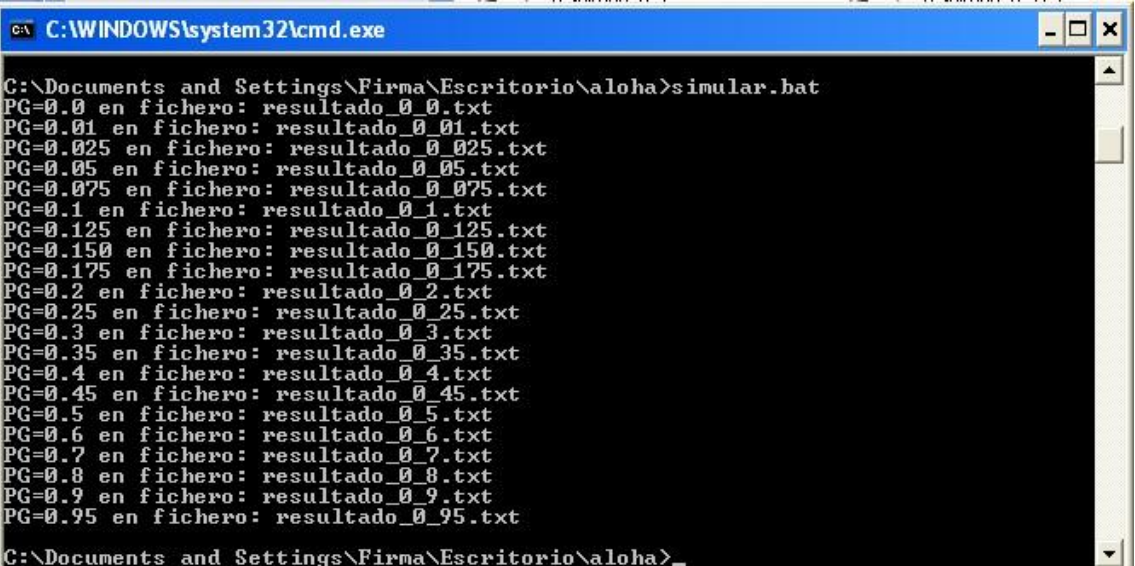
Cálculo de Variables

Proceed-Until 10000 indica que se va a realizar la simulación para 10000 ranuras.

- Una vez creados correctamente tanto el script `simular.bat` como el archivo `siminit.com`, proceda a la simulación de la siguiente manera:

C:\Documents and Settings\Firma\Escritorio\aloha\simular.bat

Esta simulación puede tardar bastante debido al alto número de ranuras ejecutadas, por lo que es recomendable que primero se realice con 10 ranuras para comprobar que todo funciona correctamente y que se crean sin ningún problema los archivos de texto, pertenecientes a cada PG, con los valores de las variables correspondientes.



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Firma\Escritorio\aloha>simular.bat
PG=0.0 en fichero: resultado_0_0.txt
PG=0.01 en fichero: resultado_0_01.txt
PG=0.025 en fichero: resultado_0_025.txt
PG=0.05 en fichero: resultado_0_05.txt
PG=0.075 en fichero: resultado_0_075.txt
PG=0.1 en fichero: resultado_0_1.txt
PG=0.125 en fichero: resultado_0_125.txt
PG=0.150 en fichero: resultado_0_150.txt
PG=0.175 en fichero: resultado_0_175.txt
PG=0.2 en fichero: resultado_0_2.txt
PG=0.25 en fichero: resultado_0_25.txt
PG=0.3 en fichero: resultado_0_3.txt
PG=0.35 en fichero: resultado_0_35.txt
PG=0.4 en fichero: resultado_0_4.txt
PG=0.45 en fichero: resultado_0_45.txt
PG=0.5 en fichero: resultado_0_5.txt
PG=0.6 en fichero: resultado_0_6.txt
PG=0.7 en fichero: resultado_0_7.txt
PG=0.8 en fichero: resultado_0_8.txt
PG=0.9 en fichero: resultado_0_9.txt
PG=0.95 en fichero: resultado_0_95.txt
C:\Documents and Settings\Firma\Escritorio\aloha>
```

Figura 127: simulación `simular.bat`

- Una vez obtenidos los archivos de texto, extrapole los datos obtenidos y se van a crear comparativas entre las variables para comprobar el comportamiento del sistema.

Cálculo de Variables

7.8 Comparativa entre variables

Una vez obtenidos los valores, se extrapolarán los datos a un documento Excel, donde se podrán apreciar los valores de **TOemisor**, **TCemisor**, **TPemisor** y **TStotal** para todos los puntos de PG. La última columna de cada tabla será la media de los valores de cada Emisor dividido entre el número de ranuras (10000).

| TOemisor | | | | | | |
|-----------------|----------------|----------------|----------------|----------------|----------------|-----------|
| PG | Emisor1 | Emisor2 | Emisor3 | Emisor4 | Emisor5 | TQ |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0,01 | 115 | 89 | 94 | 93 | 99 | 0,049 |
| 0,025 | 284 | 290 | 279 | 299 | 256 | 0,1408 |
| 0,05 | 679 | 682 | 622 | 703 | 697 | 0,3383 |
| 0,075 | 1225 | 1206 | 1241 | 1254 | 1195 | 0,6121 |
| 0,1 | 1766 | 1742 | 1801 | 1799 | 1738 | 0,8846 |
| 0,125 | 2174 | 2207 | 2303 | 2171 | 2215 | 1,107 |
| 0,15 | 2591 | 2559 | 2648 | 2632 | 2602 | 1,3032 |
| 0,175 | 2944 | 2853 | 2924 | 2908 | 2823 | 1,4452 |
| 0,2 | 3101 | 3072 | 3090 | 3016 | 3045 | 1,5324 |
| 0,25 | 3319 | 3337 | 3267 | 3336 | 3312 | 1,6571 |
| 0,3 | 3393 | 3411 | 3523 | 3399 | 3517 | 1,7243 |
| 0,35 | 3519 | 3535 | 3553 | 3482 | 3552 | 1,7641 |
| 0,4 | 3577 | 3602 | 3611 | 3574 | 3600 | 1,7964 |
| 0,45 | 3577 | 3657 | 3659 | 3651 | 3658 | 1,8202 |
| 0,5 | 3674 | 3673 | 3726 | 3630 | 3658 | 1,8361 |
| 0,6 | 3765 | 3753 | 3727 | 3681 | 3725 | 1,8651 |
| 0,7 | 3742 | 3778 | 3812 | 3696 | 3687 | 1,8715 |
| 0,8 | 3805 | 3777 | 3790 | 3788 | 3733 | 1,8893 |
| 0,9 | 3814 | 3795 | 3783 | 3804 | 3817 | 1,9013 |
| 0,95 | 3820 | 3811 | 3815 | 3781 | 3824 | 1,9051 |

Figura 128: Valores obtenidos del Tráfico Ofrecido

Cálculo de Variables

| TCemisor | | | | | | | |
|----------|---------|---------|---------|---------|---------|--------|---|
| PG | Emisor1 | Emisor2 | Emisor3 | Emisor4 | Emisor5 | TC | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0,01 | 111 | 88 | 90 | 91 | 98 | 0,0478 | |
| 0,025 | 251 | 261 | 257 | 259 | 230 | 0,1258 | |
| 0,05 | 503 | 500 | 454 | 516 | 501 | 0,2474 | |
| 0,075 | 647 | 655 | 673 | 691 | 681 | 0,3347 | |
| 0,1 | 747 | 756 | 795 | 780 | 738 | 0,3816 | |
| 0,125 | 778 | 800 | 838 | 795 | 753 | 0,3964 | |
| 0,15 | 776 | 790 | 768 | 740 | 778 | 0,3852 | |
| 0,175 | 764 | 763 | 699 | 751 | 709 | 0,3686 | |
| 0,2 | 740 | 712 | 743 | 722 | 702 | 0,3619 | |
| 0,25 | 680 | 677 | 601 | 646 | 647 | 0,3251 | |
| 0,3 | 597 | 627 | 676 | 599 | 666 | 0,3165 | |
| 0,35 | 597 | 633 | 647 | 595 | 610 | 0,3082 | |
| 0,4 | 573 | 598 | 623 | 612 | 619 | 0,3025 | |
| 0,45 | 577 | 575 | 612 | 605 | 561 | 0,293 | |
| 0,5 | 588 | 626 | 595 | 541 | 562 | 0,2912 | |
| 0,6 | 601 | 576 | 584 | 546 | 575 | 0,2882 | |
| 0,7 | 571 | 585 | 563 | 496 | 541 | 0,2756 | |
| 0,8 | 563 | 556 | 585 | 571 | 562 | 0,2837 | |
| 0,9 | 558 | 557 | 573 | 561 | 555 | 0,2804 | |
| 0,95 | 561 | 567 | 585 | 525 | 522 | 0,276 | |
| | | | | | | | |

Figura 129: Valores obtenidos del Tráfico Cursado

Cálculo de Variables

| TPemisor | | | | | | | |
|----------|---------|---------|---------|---------|---------|--------|---|
| PG | Emisor1 | Emisor2 | Emisor3 | Emisor4 | Emisor5 | TP | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0,01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0,025 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0,05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0,075 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0,1 | 0 | 0 | 1 | 0 | 1 | 0,0002 | |
| 0,125 | 1 | 2 | 0 | 2 | 3 | 0,0008 | |
| 0,15 | 3 | 1 | 4 | 5 | 4 | 0,0017 | |
| 0,175 | 3 | 6 | 12 | 9 | 12 | 0,0042 | |
| 0,2 | 8 | 12 | 7 | 13 | 14 | 0,0054 | |
| 0,25 | 19 | 16 | 20 | 16 | 20 | 0,0091 | |
| 0,3 | 26 | 24 | 23 | 23 | 18 | 0,0114 | |
| 0,35 | 21 | 26 | 22 | 26 | 26 | 0,0121 | |
| 0,4 | 31 | 30 | 31 | 26 | 31 | 0,0149 | |
| 0,45 | 33 | 36 | 32 | 36 | 37 | 0,0174 | |
| 0,5 | 37 | 30 | 35 | 47 | 37 | 0,0186 | |
| 0,6 | 42 | 37 | 40 | 53 | 38 | 0,021 | |
| 0,7 | 36 | 36 | 50 | 54 | 45 | 0,0221 | |
| 0,8 | 43 | 38 | 39 | 42 | 41 | 0,0203 | |
| 0,9 | 46 | 43 | 48 | 46 | 49 | 0,0232 | |
| 0,95 | 45 | 34 | 34 | 49 | 53 | 0,0215 | |
| | | | | | | | |

Figura 130: Valores obtenidos del Tráfico Perdido

Cálculo de Variables

| Tstotal | | | | | | |
|---------|---------|---------|---------|---------|---------|--------|
| PG | Emisor1 | Emisor2 | Emisor3 | Emisor4 | Emisor5 | TS |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0,01 | 125 | 92 | 96 | 94 | 100 | 0,0507 |
| 0,025 | 340 | 348 | 308 | 375 | 306 | 0,1677 |
| 0,05 | 961 | 989 | 909 | 1014 | 1063 | 0,4936 |
| 0,075 | 2246 | 2222 | 2223 | 2327 | 2092 | 1,111 |
| 0,1 | 3573 | 3564 | 3565 | 3668 | 3559 | 1,7929 |
| 0,125 | 4694 | 4755 | 4910 | 4605 | 4844 | 2,3808 |
| 0,15 | 5760 | 5795 | 5862 | 5902 | 5906 | 2,9225 |
| 0,175 | 6698 | 6399 | 6516 | 6531 | 6231 | 3,2375 |
| 0,2 | 7026 | 6967 | 7030 | 6671 | 6780 | 3,4474 |
| 0,25 | 7430 | 7540 | 7386 | 7607 | 7462 | 3,7425 |
| 0,3 | 7495 | 7557 | 7658 | 7672 | 7959 | 3,8341 |
| 0,35 | 8115 | 7817 | 8008 | 7809 | 7909 | 3,9658 |
| 0,4 | 7879 | 7908 | 7832 | 8007 | 7803 | 3,9429 |
| 0,45 | 7813 | 7738 | 7897 | 7743 | 7779 | 3,897 |
| 0,5 | 7867 | 8054 | 7954 | 7368 | 7847 | 3,909 |
| 0,6 | 7736 | 7972 | 7825 | 7232 | 7891 | 3,8656 |
| 0,7 | 8130 | 8148 | 7512 | 7390 | 7711 | 3,8891 |
| 0,8 | 7872 | 8079 | 8113 | 7954 | 7946 | 3,9964 |
| 0,9 | 7829 | 7994 | 7693 | 7736 | 7681 | 3,8933 |
| 0,95 | 7910 | 8349 | 8407 | 7682 | 7548 | 3,9896 |

Figura 131: Valores obtenidos del Tiempo de Servicio

Con los datos de las tablas, y con la ayuda del mismo programa Excel, pueden obtenerse las siguientes gráficas:

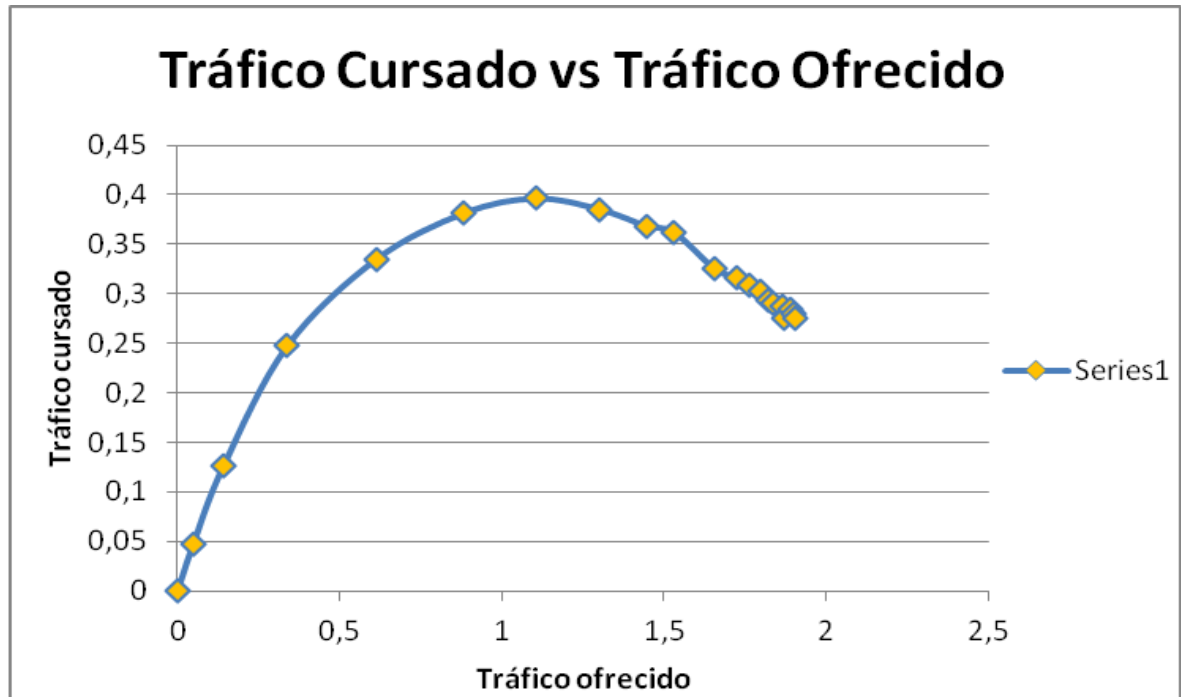


Figura 132: Tráfico Cursado vs Tráfico Ofrecido

En esta primera gráfica, se ha realizado una comparativa entre el Tráfico Cursado y el Tráfico Ofrecido, que nos muestra **la curva de eficiencia del canal**.

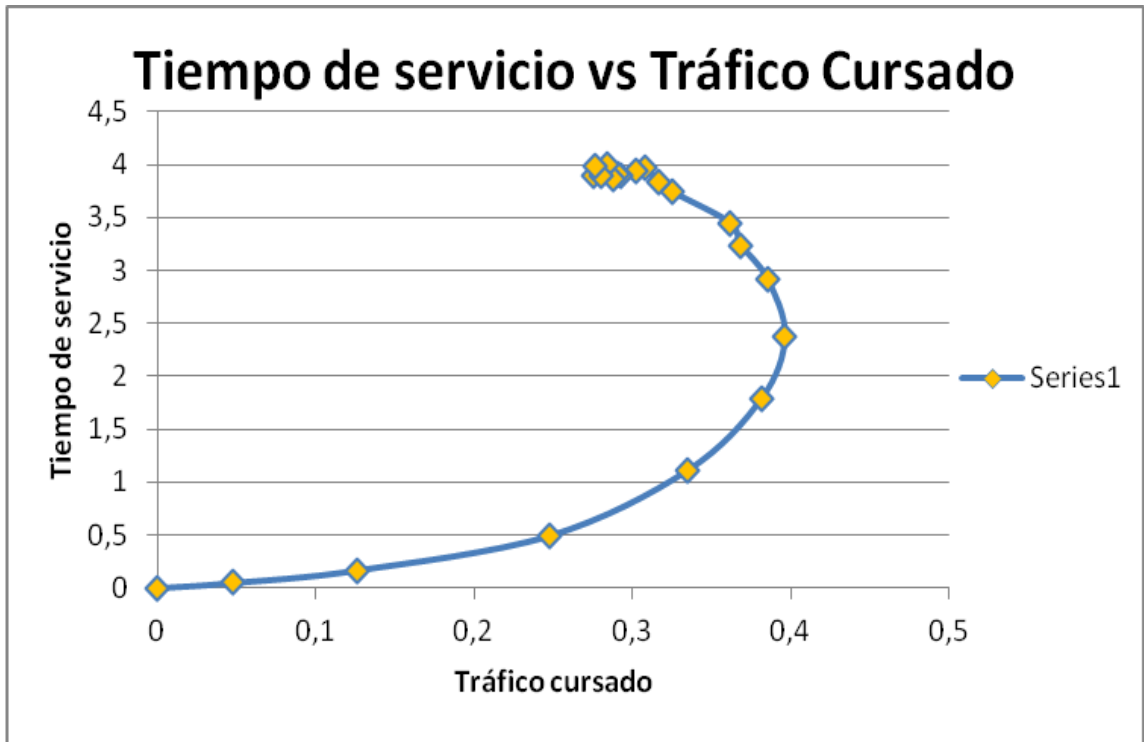


Figura 133: Tiempo de Servicio vs Tráfico Cursado

En esta segunda gráfica, se ha realizado una comparativa entre el Tiempo de servicio y el Tráfico Cursado, que nos muestra el **tiempo medio de Servicio**. Se puede observar que a partir de un valor máximo de Tráfico Cursado, el tiempo de servicio aumenta de forma exponencial.

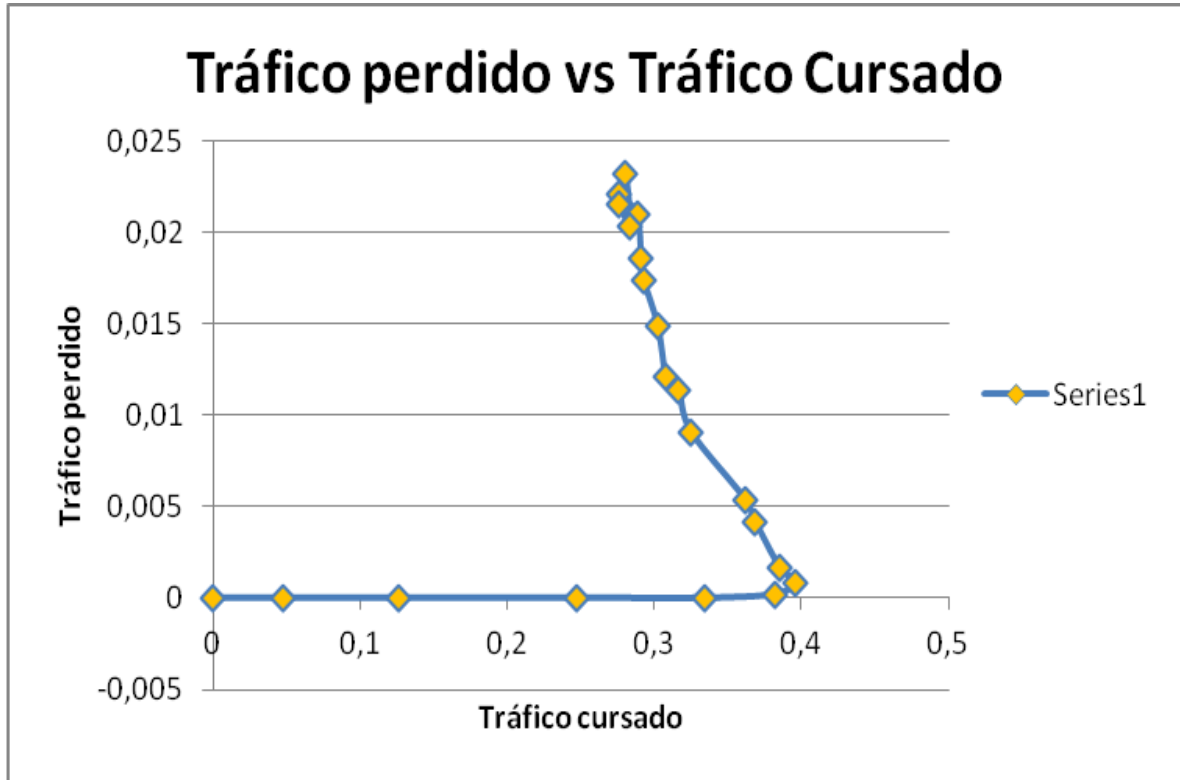


Figura 134: Tráfico Perdido vs Tráfico Cursado

En esta tercera gráfica, se ha realizado una comparativa entre el Tráfico Perdido y el Tráfico Cursado, que nos muestra el **número de paquetes perdidos**. Se puede observar, de igual manera que en la gráfica del tiempo medio de Servicio, que a partir de un valor máximo de Tráfico Cursado, las pérdidas aumentan exponencialmente.

8. CONCLUSIONES

Las conclusiones a las que se han llegado mediante la realización de este proyecto son las siguientes:

- Se han proporcionado los conocimientos requeridos para la utilización de la herramienta Telelogic Tau, así como para poder llevar a cabo la realización paso a paso de un protocolo de comunicaciones
- Se ha proporcionado una iniciación al lenguaje SDL, imprescindible para la creación de protocolos de comunicaciones.
- Se ha realizado una introducción a la Suite SDL, gracias a la cual se ha podido crear el protocolo de comunicación usado como ejemplo, Aloha Ranurado.

9.BIBLIOGRAFÍA

Para la realización de este proyecto, nos hemos apoyado sobre todo en la información que se puede encontrar en la opción de ayuda dentro del programa Telelogic Tau, así como en material proporcionado por diferentes asignaturas y algunas búsquedas web.

- › Dentro de Telelogic Tau, Manual de ayuda > SDL Suite Getting Started.
- › Tutorial sobre SDL (3º Ingeniería Técnica de Telecomunicaciones). José Juan Sánchez Manzanares.
- › Prácticas de Ingeniería de Protocolos y Servicios (3º Ingeniería Técnica de Telecomunicaciones). Juan Carlos Sánchez Aarnoutse.
- › www.telelogic.com
- › Laboratorio Ingeniería de Protocolos y Servicios (3º Ingeniería Técnica de Telecomunicaciones). Francesc Burrull i Mestres.