



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA
DEPARTAMENTO DE SISTEMAS Y AUTOMÁTICA



**DISEÑO DE UN SOFTWARE DIDÁCTICO PARA LA ENSEÑANZA-
APRENDIZAJE DE TÉCNICAS DE BOND GRAPH CON APLICACIONES EN
MODELACIÓN Y SIMULACIÓN DE SISTEMAS LINEALES EN AMBIENTE
DE PROGRAMACIÓN 20-SIM**

**KAIRABA ALEJANDRA ESTEVES SOLIS
WILLIAMS DANIEL ZAPIAIN CASTILLO**

Tutor: Ph.D. Francisco Javier Arteaga Bravo



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA
DEPARTAMENTO DE SISTEMAS Y AUTOMÁTICA



**DISEÑO DE UN SOFTWARE DIDÁCTICO PARA LA ENSEÑANZA-
APRENDIZAJE DE TÉCNICAS DE BOND GRAPH CON APLICACIONES EN
MODELACIÓN Y SIMULACIÓN DE SISTEMAS LINEALES EN AMBIENTE
DE PROGRAMACIÓN 20-SIM**

**TRABAJO ESPECIAL DE GRADO PRESENTADO ANTE LA ILUSTRE
UNIVERSIDAD DE CARABOBO PARA OPTAR POR EL TÍTULO DE
INGENIERO ELECTRICISTA**

**KAIRABA ALEJANDRA ESTEVES SOLIS
WILLIAMS DANIEL ZAPIAIN CASTILLO**

Tutor: Ph.D. Francisco Javier Arteaga Bravo

DEDICATORIA

A Dios,
a mis Padres,
a mis hermanos,
a mis abuelos.

Kairaba A. Esteves S.

DEDICATORIA

A DIOS, a Él le pertenece este trabajo, por Él pude terminarlo, con Él pude resolver las dificultades y celebrar las victorias.

A la Virgen María, por su amor maternal y testimonio de constancia y fidelidad a Dios, su ejemplo me ha guiado por mis caminos.

A mis Padres, co-creadores con Dios de mi vida, ellos con sacrificio y amor abnegado me han alentado, apoyado, formado, alimentado, etc.; y para ellos es este fruto. ¡Lo hicimos!.

A mis hermanas, que con amor fraternal me han apoyado en todo; por encima de disgustos y alegrías, allí están.

A mis sobrinos y ahijados, que estas líneas sirvan de aliento a culminar los proyectos que se propongan.

A mis familiares, abuelos, primos, tías (en especial Ana, Armelia, Cristina) que me han ayudado a forjarme como ser humano y social.

“¿Cómo pagaré al Señor todo el bien que me ha hecho? Alzaré la copa de la bendición, e invocaré el nombre del Señor”.

Sal 116(115), 12-13.

Willians Daniel

AGRADECIMIENTOS

A DIOS primeramente, por darnos la fuerza, la inteligencia, la vida para llevar a cabo la realización del presente trabajo especial de grado, y por acompañarnos y ayudarnos en todo momento durante nuestros estudios.

A nuestros Padres: Victor Esteves, Solvi Solis de Esteves, Willians Zapiain y Milagros Castillo de Zapiain por el apoyo incondicional, la formación y valores recibidos de ellos, y por darnos la vida sin la cual hoy no podríamos cosechar este fruto exitoso.

A nuestro tutor, Ph.D. Francisco Arteaga por darnos la oportunidad de trabajar con él, y por la colaboración y el apoyo brindado durante el desarrollo de este trabajo especial de grado.

A nuestros profesores jurados: Ing. Whendy García e Ing. Teddy Rojas, por su atenta colaboración constructiva para la revisión y aprobación del presente trabajo.

A la Ing. Oriana Barrios y al profesor Alexis Guarate por sus aportes en cuanto al diseño del software didáctico.

A nuestros amigos Daniel Chávez y Christopher Motta por su valiosa colaboración en la digitalización del logo principal del software didáctico.

Al Ing. Carlos Quiroz por su apoyo incondicional.

A nuestros familiares y amigos que siempre han estado presentes tanto en los momentos buenos como en las adversidades, brindándonos su apoyo hasta el último momento de alcanzar una de nuestras importantes metas. Y a todas aquellas personas que de una u otra manera colocaron su granito de arena.

A todos muchas gracias.

KAIRABA A. ESTEVES S.

WILLIANS D. ZAPIAIN C.

INTRODUCCIÓN

La técnica de Bond Graph (grafos de enlace) es un enfoque gráfico para la modelación de sistemas físicos en los cuales los puertos de energía de los componentes del sistema se conectan por enlaces que transfieren la energía y hacen posible la dinámica del sistema. En la actualidad, la modelación y simulación de los sistemas son técnicas indispensables, más aun, cuando hay que enfrentarse a sistemas multidisciplinarios con cierto grado de complejidad; es aquí donde resalta la importancia del uso de la técnica de bond graph ya que incorpora en un mismo modelo sistemas provenientes de diferentes dominios físicos a través de la generalización de las variables involucradas en la dinámica de la energía de los sistemas.

Una excelente combinación resulta cuando el talento del diseñador se complementa con un buen software, el cual hace posible establecer una valoración final antes de que los sistemas sean construidos. Por tal motivo, se dispone del software 20-Sim, el cual sirve para simular el comportamiento de sistemas dinámicos e incluye la modelación basada en la técnica de Bond Graph.

El presente trabajo de grado desea abrir una nueva ventana en el campo de la modelación y simulación de sistemas físicos lineales, utilizando la técnica de Bond Graph y el software 20-Sim como herramientas para el estudio y análisis de sistemas físicos.

Este trabajo de grado se distribuye en 8 capítulos, los cuales se reseñan a continuación:

Capítulo I, se describe el planteamiento del problema, los objetivos que se desean alcanzar y el alcance de la investigación.

Capítulo II, se mencionan los antecedentes del problema, los orígenes del Bond Graph y el marco conceptual el cual contiene una descripción detallada de la teoría de la técnica de Bond Graph, las variables que maneja, los elementos, las relaciones constitutivas, la causalidad en cada uno de los elementos, una breve descripción de los sistemas lineales y la importancia de la modelación y la simulación en el campo de la ingeniería.

Capítulo III, contiene la metodología a utilizar para la realización del trabajo de grado.

Capítulo IV, se presentan los procedimientos utilizados que se requieren para la obtención de modelos en Bond Graph de diferentes dominios físicos.

Capítulo V, se desarrolla un manual del software 20-Sim, el cual contiene ejemplos de modelación y simulación de sistemas lineales de diferentes dominios físicos desarrollados en el software 20-Sim.

Capítulo VI, se sintetizan las novedades de la versión de 20-Sim 4.3.0 (versión más reciente), y se desarrolla la modelación y simulación de un sistema eléctrico en esta versión.

Capítulo VII, se desarrolla un manual para el usuario del software didáctico el cual contiene la descripción de los botones y ventanas que contiene el software desarrollado.

Capítulo VIII, se establecen las conclusiones y recomendaciones del trabajo de grado.

ÍNDICE GENERAL

Resumen	VII
Índice de Figuras	XIV
Índice de Tablas	XXVII
Introducción	XXVIII

CAPÍTULO I: Formulación del Problema.

1.1.- Planteamiento del Problema.....	1
1.2.- Justificación de la Investigación.....	2
1.3.- Objetivos de la Investigación.....	3
1.3.1.- Objetivo General.....	3
1.3.2.- Objetivos Específicos.....	3
1.4.- Alcance.....	4

CAPÍTULO II: Marco Teórico.

2.1.- Antecedentes del Problema.....	7
2.2.- Marco Conceptual.....	9
2.2.1.- Introducción.....	9
2.2.2.- Orígenes del Bond Graph.....	9
2.2.3.- Conceptos y Definiciones.....	10
2.2.3.1.- Modelación Basada en Puertos.....	12
2.2.3.2.- Elementos de Bond Graph.....	14
2.2.3.2.1.- Elementos pasivos de un solo Puerto.....	14
• Elemento R.....	14
• Elemento C.....	15
• Elemento I.....	16

2.2.3.2.2.- Elementos activos de un solo Puerto.....	16
• Fuentes de Esfuerzo y Flujo.....	16
• Fuente de Esfuerzo.....	17
• Fuente de Flujo.....	17
2.2.3.2.3.- Elementos de Dos Puertos.....	17
• Transformador.....	17
• Girador.....	18
2.2.3.2.4.- Elementos de Tres Puertos o más.....	19
• Unión-0 o Vinculo-0.....	19
• Unión-1 o Vinculo-1.....	20
2.2.3.2.5.- Elementos Multipuertos.....	21
• Capacitancia Multipuerto.....	22
• Inductancia Multipuerto.....	23
• Multipuerto Combinado C e I.....	23
2.2.4.- Asignación Causal.....	24
2.2.4.1.- Causalidad.....	24
2.2.4.2.- Objetivos del Análisis Causal.....	26
2.2.4.3.- Causalidad en los Elementos Básicos de Bond Graph.....	27
• Causalidad en las Fuentes de Energía.....	27
• Causalidad en los Elementos Resistivos.....	27
• Causalidad en los Elementos Capacitivos e Inductivos.....	28
• Causalidad en Transformadores.....	29
• Causalidad en Giradores.....	30
• Causalidad en Unión-0 o Vinculo-0.....	31
• Causalidad en Unión-1 o Vinculo-1.....	31
2.2.5.- Otros Elementos de Bond Graph.....	32
• Fuentes Moduladas.....	32
a) Fuente Modulada de Esfuerzo.....	33
b) Fuente Modulada de Flujo.....	33
• Transformadores y Giradores Modulados.....	34
• Sensor de Esfuerzo.....	35

• Sensor de Flujo.....	36
• Divisor de Potencia.....	36
• Sensor de Energía.....	36
2.2.6.- Sistemas Lineales.....	37
2.3.- Importancia de la Modelación y Simulación de los Sistemas en Ingeniería y la Aplicación del Software 20-Sim y Bond Graph.....	38
CAPÍTULO III: Fases Metodológicas.	
3.1.- Metodología a Utilizar.....	43
3.2.- Técnicas de Recopilación de Datos.....	44
3.3.- Análisis de Datos y Resultados.....	44
3.4.- Organización del Trabajo por Partes.....	44
CAPÍTULO IV: Modelación en Bond Graph.	
4.1.- La Modelación en Bond Graph.....	49
4.1.1.- Procedimiento para la Construcción de Modelos en Bond Graph.....	49
• Reglas de Simplificación de Modelos en Bond Graph.....	50
4.1.2.- Procedimiento para la Obtención de Modelos en Bond Graph para Sistemas Eléctricos.....	52
4.1.2.1.- Circuito Eléctrico RLC.....	52
4.1.2.2.- Circuito Eléctrico Mixto.....	57
4.1.2.3.- Circuito Eléctrico con Transformador.....	62
4.1.3.- Procedimiento para la Obtención de Modelos en Bond Graph para Sistemas Mecánicos.....	64
4.1.3.1.- Sistema Masa Resorte Amortiguador.....	64
4.1.3.2.- Sistema Mecánico Mixto.....	69
4.1.4.- Procedimiento para la Obtención de Modelos en Bond Graph para Sistemas Hidráulicos.....	71
4.1.5.- Procedimiento para la Obtención del Modelo en Bond Graph de un	

Motor de Corriente Continua (CC).....	73
CAPÍTULO V: Manual 20-sim.	
5.1.- Introducción.....	81
5.2.- Conociendo 20-sim.....	81
5.3.- Modelación y Simulación con 20-sim.....	87
5.3.1.- Creación.....	87
5.3.2.- Compilación/Depuración.....	88
5.3.3.- Experimentación.....	88
5.3.4.- Simulación/Depuración.....	88
5.3.5.- Análisis.....	89
5.3.6.- Modos de Operación.....	89
5.4.- Jerarquía de los Modelos.....	89
5.5.- Tipos de Modelos.....	93
5.5.1.- Modelos de Ecuaciones.....	93
5.5.2.- Modelos de Diagramas de Bloques.....	95
5.5.3.- Modelos de Bond Graph.....	96
5.5.4.- Modelos de Diagramas Icónicos.....	97
5.6.- Biblioteca de Modelos.....	99
5.7.- Experimentos.....	103
• Valores de los Parámetros.....	106
• Propiedades de Ejecución.....	107
• Especificaciones de la Gráfica.....	108
5.8.- Modelación y Simulación de Sistemas Físicos Lineales Aplicando la Teoría de Bond Graph.....	111
5.8.1.- Sistema Masa-Resorte-Amortiguador.....	112
5.8.2.- Circuito RLC.....	130

5.8.3.- Circuito Eléctrico Mixto.....	145
5.8.4.- Motor de Corriente Continua (CC).....	163
CAPITULO VI: Versión 20-Sim 4.3.0.	
6.1.- Versión 20-Sim 4.3.0.....	173
6.1.1.- Introducción.....	173
6.1.2.- Versiones del 20-Sim 4.3.0.....	173
• Visor / versión demostración.....	173
• Estándar.....	173
• Profesional.....	173
6.2.- Requerimientos de Software y Hardware.....	174
6.3.- Características.....	174
6.3.1.- Modelación.....	174
6.3.2.- Simulación.....	174
6.3.3.- Análisis.....	174
6.3.4.- Exportación.....	175
6.4.- Toolboxes del Software 20-Sim.....	175
6.4.1.- Toolboxes Dominio del Tiempo.....	175
6.4.2.- Toolboxes dominio de la Frecuencia.....	175
6.4.3.- Cuadro de Controles.....	175
6.4.4.- Toolboxes Mecánica 3D.....	175
6.4.5.- Herramientas de Animación.....	176
6.4.6.- Toolboxes Mecatronica.....	176
6.5.- Novedades de la Versión 4.3.0.....	176
6.6.- Desarrollo de un Circuito RLC en el 20-Sim 4.3.0.....	177
CAPITULO VII: Manual para el usuario del Software Didáctico.	
7.1.- Introducción.....	195

7.2.- Requerimientos del Sistema para una Visualización Optima del Sitio Web.....	195
7.3.- Organización del sitio Web.....	195
• Página Inicio.....	196
• Página Menú.....	196
• Página Desarrollo.....	196
7.4.- Elementos de Navegación del Sitio Web.....	198
• Barra de Navegación.....	198
• Botones de Navegación.....	198
7.5.- Exploración de las Páginas del Menú.....	200
 CAPITULO VIII: Conclusiones y Recomendaciones.	
8.1.- Conclusiones.....	207
8.2.- Recomendaciones.....	209
Apéndice A. Métodos de Integración Numérica en 20-Sim.....	213
Apéndice B. Variables en 20-Sim.....	221
Apéndice C. Reglas de Simplificación para la Técnica de Bond Graph.....	227
Apéndice D. Procedimiento para la Asignación de la Causalidad.....	233
Apéndice E. Programas de Simulación Dinámica Basados en Bond Graph.....	241
Bibliografía.....	249

INDICE DE FIGURAS

Figura 2.1.- Circuito LC.....	12
Figura 2.2.- Modelo LC con Puerto.....	12
Figura 2.3.- Bond Modelo Basado en Puertos.....	13
Figura 2.4.- Modelo en Puertos del Circuito RLC.....	13
Figura 2.5.- Grafo del Elemento Resistivo.....	14
Figura 2.6.- Grafo del Elemento Capacitivo	15
Figura 2.7.- Grafo del Elemento Inercial	16
Figura 2.8.- Grafo Fuente de Esfuerzo.....	17
Figura 2.9.- Grafo Fuente de Flujo.....	17
Figura 2.10.- Grafo Transformador.....	18
Figura 2.11.- Grafo Girador.....	18
Figura 2.12.- Representación Unión 0.....	19
Figura 2.13.- Representación Unión 1.....	20
Figura 2.14.- Ejemplo Elemento Multipuerto: Inductancia mutua.....	21
Figura 2.15.- Representación Multipuerto.....	22
Figura 2.16.- Representación Multipuerto C.....	23
Figura 2.17.- Representación Multipuerto I.....	23
Figura 2.18.- Representación Multipuerto Combinado C e I.....	24
Figura 2.19.- Grafo con Causalidad Flujo.....	25
Figura 2.20.- Grafo con Causalidad Esfuerzo.....	25
Figura 2.21.- Causalidad para un Bond.....	25

Figura 2.22.- Enlace Activo.....	26
Figura 2.23.- Causalidad en la Fuente de Flujo.....	27
Figura 2.24.- Causalidad en la fuente de esfuerzo.....	27
Figura 2.25.- Relación causal en Elementos Resistivos (a) Causa es el Flujo (b) Causa es el Esfuerzo.....	28
Figura 2.26.- Causalidad Esfuerzo en elemento C	28
Figura 2.27.- Causalidad Flujo en elemento C.....	29
Figura 2.28.- Causalidad Flujo en elemento I.....	29
Figura 2.29.- Causalidad Esfuerzo en elemento I.....	29
Figura 2.30.- Causalidad en los Transformadores.....	30
Figura 2.31.- Causalidad en los Giradores.....	30
Figura 2.32.- Causalidad en Unión 0.....	31
Figura 2.33.- Causalidad en Unión 1.....	32
Figura 2.34.- Fuente Modulada de Esfuerzo.....	33
Figura 2.35.- Fuente Modulada de Flujo.....	33
Figura 2.36.- Transformador Modulado con Causalidad de Esfuerzo Entrante.....	34
Figura 2.37.- Transformador Modulado con Causalidad de Esfuerzo Saliente.....	34
Figura 2.38.- Girador Modulado con Causalidad de Esfuerzo Entrante.....	35
Figura 2.39.- Girador Modulado con Causalidad de Esfuerzo Saliente.....	35
Figura 2.40.- Sensor de Esfuerzo.....	35
Figura 2.41.- Sensor de Flujo.....	36
Figura 2.42.- Sensor de Energía.....	36
Figura 2.43.- Sistema Físico.....	37

Figura 2.44.- Entrada $x_1(t)$ y la salida $y_1(t)$	37
Figura 2.45.- Entrada $x_2(t)$ y la salida $y_2(t)$	37
Figura 2.46.- Ley que rige el principio de Superposición.....	38
Figura 4.1.- Regla Simplificación Uniones en Bond Graph	50
Figura 4.2.- Regla de Simplificación Bond Graph referida a las Uniones-0.....	50
Figura 4.3.- Regla de Simplificación Bond Graph referida a las Uniones-1.....	50
Figura 4.4.- Regla de Simplificación Bond Graph Referente a Diferencia de Esfuerzo o Flujo 1.....	51
Figura 4.5.- Regla de Simplificación Bond Graph Referente a Diferencia de Esfuerzo o Flujo 2	51
Figura 4.6.- Circuito RLC.....	52
Figura 4.7.- Uniones-0 Circuito RLC.....	52
Figura 4.8.- Inserción de Uniones -1 Circuito RLC.....	53
Figura 4.9.- Inserción de Elementos Circuito RLC.....	53
Figura 4.10.- Inserción de enlaces faltantes para completar modelo RLC.....	54
Figura 4.11.- Identificación Nodo de Referencia en Circuito RLC.....	54
Figura 4.12.- Simplificación de los Enlaces Conectados al Nodo de Referencia del Circuito RLC.....	55
Figura 4.13.- Identificación en Modelo RLC de Uniones y Enlaces a Eliminar.....	55
Figura 4.14.- 1ra Simplificación de Uniones y Enlaces en Circuito RLC.....	55
Figura 4.15.- 2da Simplificación de Uniones y Enlaces en Circuito RLC.....	56
Figura 4.16.- Obtención del Modelo en Bond Graph del Circuito RLC.....	56
Figura 4.17.- Asignación de Causalidad al Circuito RLC.....	56

Figura 4.18.- Circuito Eléctrico Mixto.....	57
Figura 4.19.- Identificación de Nodos en Circuito Mixto.....	57
Figura 4.20.- Ubicación Uniones-0 Para el Circuito RLC.....	58
Figura 4.21.- Ubicación Uniones-1 y Uniones- 0 Para el Circuito RLC.....	58
Figura 4.22.- Inserción de Enlaces Correspondientes a cada Elemento del Circuito RLC	59
Figura 4.23.- Inserción Enlaces Para Unir Elementos del Circuito RLC.....	59
Figura 4.24.- Ubicación Nodo de Referencia Para el Circuito RLC.....	60
Figura 4.25.- Simplificación de los Enlaces Conectados al Nodo de Referencia.....	60
Figura 4.26.- Simplificación del Circuito RLC Aplicando las Reglas.....	61
Figura 4.27.- Modelo en Bond Graph del Circuito RLC.....	61
Figura 4.28.- Circuito Eléctrico con Transformador.....	62
Figura 4.29.- Procedimiento para Obtener el Bond Graph del Sistema Eléctrico.....	63
Figura 4.30.- Bond Graph del Sistema Eléctrico.....	63
Figura 4.31.- Sistema Masa-Resorte-Amortiguador.....	64
Figura 4.32.- Ubicación de las Velocidades en el Sistema Masa-Resorte Amortiguador.	64
Figura 4.33.- Ubicación uniones-1 del modelo Masa – Resorte – Amortiguador.....	65
Figura 4.34.- Inserción de las Uniones-0 del Modelo M - R –A.....	65
Figura 4.35.- Inserción Enlace que Representa la Fuerza que da origen a las velocidades	66
Figura 4.36.- Inserción de cada elemento en el sistema Masa – Resorte –Amortiguador	66
Figura 4.37.- Inserción Uniones Para Completar el Sistema Masa – Resorte – Amortiguador.....	67

Figura 4.38.- Identificación Unión-1 de Referencia.....	67
Figura 4.39.- Simplificación de Enlaces Conectados a la Unión de Referencia del Sistema M-R-A	68
Figura 4.40.- Simplificación Uniones-0 del Sistema Masa – Resorte -Amortiguador	68
Figura 4.41.- Simplificación Uniones-1 del Sistema M – R – A.....	68
Figura 4.42.- Sistema Mecánico Mixto.....	69
Figura 4.43.- Procedimiento para Obtener el Bond Graph de un Sistema Mecánico...	70
Figura 4.44.- Bond Graph para el Sistema Mecánico.....	71
Figura 4.45.- Sistema Hidráulico.....	71
Figura 4.46.- Procedimiento Para Obtener Modelo en Bond Graph del Sistema Hidráulico.....	72
Figura 4.47.- Bond Graph Para el Sistema Hidráulico.....	73
Figura 4.48.- Esquema de un motor de CC.....	74
Figura 4.49.- Identificación de Dominios y Elementos.....	74
Figura 4.50.- Identificación de las referencias.....	75
Figura 4.51.- Identificación de los voltajes y velocidades relevantes.....	75
Figura 4.52.- Primer Bosquejo del Bond Graph. Voltajes y Velocidades con Uniones 0 y 1, Respectivamente.....	76
Figura 4.53.- Inclusión de las diferencias de voltajes del modelo.....	76
Figura 4.54.- Bond Graph del modelo Motor de CC completado.....	77
Figura 4.55.- Bond Graph del modelo Motor de CC simplificado.....	78
Figura 5.1.- Estructura organizativa de 20-sim, versión 3.6.....	82
Figura 5.2.- Ventana Principal de 20-SIM.....	83

Figura 5.3.- Explorador de Archivos.....	83
Figura 5.4.- Editor mostrando el modelo DiscreteController.em	84
Figura 5.5.- Ventana de chequeo del modelo.....	84
Figura 5.6.- Ventana del Simulador de 20-sim.....	85
Figura5.7.- Resultados de la ejecución del modelo DiscreteController.em en el Simulador de 20-sim.....	86
Figura 5.8.- Modelación y Simulación en 20-sim.....	87
Figura 5.9.- Esquema gráfico de la Jerarquía de un modelo.....	90
Figura 5.10.- Submodelo motor1	91
Figura 5.11.- Tipos de Submodelos.....	91
Figura 5.12.- Visualización de la jerarquía de modelos y submodelos.....	92
Figura 5.13.- Modelo de Ecuacion Principal.....	94
Figura 5.14.- Visualización de la Jerarquía de Modelos y Submodelos.....	94
Figura 5.15.- Modelo Principal de Diagrama de Bloques.....	95
Figura 5.16.- Submodelo de Diagrama de Bloques.....	96
Figura 5.17.- Modelo Principal de Bond Graph.....	96
Figura 5.18.- Submodelo de Bond Graph.....	97
Figura 5.19.- Modelo Principal de Diagramas Icónicos.....	98
Figura 5.20.- Submodelo de Diagramas Icónicos.....	98
Figura 5.21.- Navegador de la biblioteca de 20-sim.....	99
Figura 5.22.- Contenido de la carpeta Block Diagram	100
Figura 5.23.- Reorganización del contenido de la carpeta Block Diagram	101
Figura 5.24.- Reorganización del espacio de trabajo en 20-sim.....	102

Figura 5.25.- Añadiendo un submodelo al Editor desde la Biblioteca de 20-sim...	103
Figura 5.26.- Modelo Linearize.em	104
Figura 5.27.- Resultados de la simulación del modelo Linearize.em	105
Figura 5.28.- Editor de Parámetros.....	106
Figura 5.29.- Resultados de la primera y segunda simulación.....	107
Figura 5.30.- Propiedades de Ejecución.....	107
Figura 5.31.- Los resultados de la simulación después de cambiar el tiempo Finish a 50.....	108
Figura 5.32.- Editor de Propiedades de la Gráfica.....	109
Figura 5.33.- Seleccionador de Variables.....	109
Figura 5.34.- Editor de Propiedades de la Gráfica con una Variable.....	110
Figura 5.35.- Resultados de la simulación después de adicionar una variable.....	111
Figura 5.36.- Diagrama Iconico Sistema Masa-Resorte-Amortiguador.....	112
Figura 5.37.- Modelo en Grafo de Enlace.....	112
Figura 5.38.- Nuevo modelo en el Editor	113
Figura 5.39.- Insertando Unión 1.....	114
Figura 5.40.- Insertando elementos de Bond Graph	115
Figura 5.41.- Editor de Atributos.....	115
Figura 5.42.- Edición del nombre de los elementos.....	116
Figura 5.43.- Comandos para reposicionar el nombre de los elementos.....	117
Figura 5.44.- Reposicionando el nombre de los elementos.....	117
Figura 5.45.- Creación del primer enlace.....	118
Figura 5.46.- Creación del segundo enlace.....	119

Figura 5.47.- Modelo terminado.....	120
Figura 5.48.- Ventana de Resultados del Chequeo de un Modelo Correcto.....	120
Figura 5.49.- Ventana de Resultados del Chequeo de un Modelo Incorrecto...	120
Figura 5.50.- Ventana del simulador.....	121
Figura 5.51.- Valores de los elementos del modelo.....	123
Figura 5.52.- Edición de los valores de los parámetros de simulación del modelo.	123
Figura 5.53.- Edición del nombre de la gráfica.....	124
Figura 5.54.- Elección de la variable a simular.....	125
Figura 5.55.- Edición de apariencia de la simulación.....	126
Figura 5.56.- Resultado de la simulación del modelo M-R-A usando el Método de Euler con paso de avance igual a 0.2 en 20-sim.....	127
Figura 5.57.- Edición de los valores de los parámetros de simulación del modelo.....	128
Figura 5.58.- Resultado de la simulación del modelo Masa-Resorte-Amortiguador usando el Método de Runge-Kutta 4 con paso de avance igual a 0.1 en 20- sim.....	129
Figura 5.59.- Circuito RLC.....	130
Figura 5.60.- Circuito RLC.....	130
Figura 5.61.- Selección del espacio de trabajo.....	131
Figura 5.62.- Inserción de la Unión 1.....	131
Figura 5.63.- Navegador de la biblioteca de 20-sim.....	132
Figura 5.64.- Inserción del elemento Se.....	133
Figura 5.65.- Edición del nombre del modelo Se.em	134
Figura 5.66.- Edición de la posición del nombre del elemento.....	134

Figura 5.89.- Cambio de posición de los nombres de los elementos.....	151
Figura 5.90.- Creación de los enlaces (bonds) entre los elementos.....	151
Figura 5.91.- Comprobación del modelo creado.....	152
Figura 5.92.- Ventana de comprobación exitosa.....	153
Figura 5.93.- Ventana del Simulador.....	153
Figura 5.94.- Editor de valores de los parámetros del modelo.....	154
Figura 5.95.- Edición de los valores de los parámetros del modelo.....	154
Figura 5.96.- Edición de los valores de los parámetros de ejecución de la simulación.....	155
Figura 5.97.- Edición de las propiedades de la gráfica en la simulación.....	156
Figura 5.98.- Elección de la variable a simular.....	156
Figura 5.99.- Editor de propiedades de la gráfica.....	157
Figura 5.100.- Gráfica del voltaje del capacitor.....	158
Figura 5.101.- Gráfica de la corriente del capacitor.....	159
Figura 5.102.- Gráfica de la corriente del resistor “ R1 ”.....	159
Figura 5.103.- Gráfica del voltaje del resistor “ R1 ”.....	160
Figura 5.104.- Gráfica del voltaje del resistor “ R2 ”.....	160
Figura 5.105.- Gráfica de la corriente del resistor “ R2 ”.....	161
Figura 5.106.- Gráfica del voltaje del Inductor.....	161
Figura 5.107.- Gráfica de la corriente del inductor.....	162
Figura 5.108.- Gráfica del voltaje de la fuente.....	162
Figura 5.109.- Esquema electro-mecánico de un motor de CC.....	163
Figura 5.110.- Bond Graph del motor de CC.....	164

Figura 5.111.- Inserción del modelo bond graph del motor de CC en 20-sim.....	165
Figura 5.112.- Edición de los valores de los parámetros del modelo.....	166
Figura 5.113.- Edición de los valores de los parámetros de ejecución de la simulación.	167
Figura 5.114.- Edición de las propiedades de la gráfica en la simulación.....	167
Figura 5.115.- Elección de la variable a simular.....	168
Figura 5.116.- Editor de propiedades de la gráfica.....	169
Figura 5.117.- Gráfica de la Velocidad Angular del Motor.....	170
Figura 5.118.- Gráfica del Torque del Motor.....	170
Figura 6.1.- Circuito RLC.....	177
Figura 6.2.- Bond Graph del Circuito RLC Circuito RLC.....	177
Figura 6.3.- Selección del Espacio de Trabajo.....	178
Figura 6.4.- Selección del Botón Modo Depuración.....	178
Figura 6.5.- Selección de Librería de Bond Graph.....	179
Figura 6.6.- Selección de la Unión-1.....	180
Figura 6.7.- Inserción de la Unión-1.....	180
Figura 6.8.- Inserción del Elemento Se.....	181
Figura 6.9.- Inserción de Elementos del Modelo.....	181
Figura 6.10.- Enlace entre el elemento Se y la Union-1.....	182
Figura 6.11.- Modelo en Bond Graph del circuito RLC.....	183
Figura 6.12.- Subventana Process.....	184
Figura 6.13.- Ventana del simulador del 20-Sim 4.3.0.....	184
Figura 6.14.- Edición de los valores de los Elementos del Modelo.....	185

Figura 6.15.- Edición de los valores de los parámetros (tiempo) de simulación modelo.....	186
Figura 6.16.- Subventana <i>plot properties</i>	187
Figura 6.17.- Elección de la Variable a Simular.....	187
Figura 6.18.- Identificación de Variable a Simular y parámetros del Grafico.....	188
Figura 6.19.- Corriente del Circuito RLC.....	189
Figura 6.20.- Voltaje del Resistor.....	190
Figura 6.21.- Voltaje del Capacitor.....	191
Figura 6.22.- Voltaje del Inductor.....	191
Figura 6.23.- Voltaje de la Fuente.....	192
Figura 7.1.- Estructura del Software Didáctico.....	196
Figura 7.2.- Página Inicio.....	197
Figura 7.3.- Página Menú.....	197
Figura 7.4.- Página de desarrollo.....	198
Figura 7.5.- Barra de Navegación.....	199
Figura 7.6.- Botón Bond Graph.....	200
Figura 7.7.- Página de desarrollo Teoría de Bond Graph, Conceptos y Definiciones.	200
Figura 7.8.- Botón Aplicación Bond Graph.....	201
Figura 7.9.- Página de desarrollo Aplicaciones de la Técnica de Bond Graph.....	201
Figura 7.10.- Botón 20-sim.....	202
Figura 7.11.- Página de desarrollo Manual del Software 20-sim.....	202
Figura 7.12. Botón ayuda.....	202
Figura 7.13. Página de desarrollo. Ayuda al Usuario.....	203

Figura 7.14. Botón Tesis Pdf.....	203
Figura 7.15. Página de desarrollo Tesis.pdf.....	204
Figura A.1.- Descripción del Método de Euler.....	214
Figura A.2.- Grafica del Desarrollo del Método RK4.....	216
Figura B.1.- Ecuaciones que describen el submodelo Friction.em.....	220
Figura C.1.a.- Bond Graph sin simplificación.....	225
Figura C.1.b.- Bond Graph Simplificado.....	225
Figura D.1.- Descripción enlace fuerte.....	225
Figura D.2.- Descripción enlaces débiles.....	225

INDICE DE TABLAS

Tabla 2.1.- Generalización de variables en un dominio físico.....	11
Tabla 5.1.- Elementos para usar en el modelo.....	114
Tabla 5.2.- Valores de los elementos del modelo.....	122
Tabla 5.3.- Valores de los parámetros de simulación del modelo.....	122
Tabla 5.4.- Parámetros de la edición de apariencia de la simulación.....	125
Tabla 5.5.- Valores de los parámetros de simulación del modelo.....	128
Tabla 5.6.- Elementos para usar en el modelo.....	132
Tabla 5.7.- Valores de los elementos del modelo.....	138
Tabla 5.8.- Valores de los parámetros de simulación del modelo.....	139
Tabla 5.9.- Parámetros de la edición de apariencia de la simulación del modelo....	141
Tabla 5.10.- Nuevas variables a simular.....	142
Tabla 5.11.- Valores de los elementos del modelo.....	154
Tabla 5.12.- Valores de los parámetros de ejecución de la simulación.....	155
Tabla 5.13.- Valores de los parámetros las propiedades de la gráfica.....	157
Tabla 5.14.- Valores de los elementos del modelo.....	166
Tabla 5.15.- Valores de los parámetros de ejecución de la simulación.....	166
Tabla 5.16.- Valores de los parámetros las propiedades de la gráfica.	168
Tabla 6.1.- Parámetros circuito RLC.....	185
Tabla 6.2.- Valores de los Parámetros de Simulación del Modelo.....	186
Tabla 6.3.- Parámetros de la edición de apariencia de la simulación.....	188

Tabla 6.4.- Asignación de Variables Faltantes a Simular.....	189
Tabla 7.1.- Tabla de Botones de navegación.....	199



APÉNDICE A

MÉTODOS DE INTEGRACIÓN NUMÉRICA EN 20-SIM

El Software 20-sim posee una serie de métodos numéricos de integración para llevar a cabo la ejecución de las simulaciones [19]. La siguiente lista muestra los métodos de integración disponibles en las propiedades de ejecución (*Run Properties*) de la ventana del simulador de 20-sim.

- Euler
- Backward Euler
- Adams-Bashford 2
- Runge-Kutta 2
- Runge-Kutta 4
- Runge-Kutta Dormand Prince 8
- Runge-Kutta-Fehlberg
- Vode Adams
- Backward Differentiation Formula (BDF)
- Modified Backward Differentiation Formula (MBDF)

Todos los métodos anteriores son capaces de resolver numéricamente las ecuaciones diferenciales que describen y/o fundamentan los submodelos físicos modelados en el software 20-sim. A continuación se redacta una breve descripción de los métodos que se utilizan en el presente trabajo de grado y cuál, de la variedad disponible en 20-sim, se elige para el desarrollo de los ejercicios.

Método de Euler

Se tiene la siguiente Ecuación Diferencial Ordinaria (E.D.O.) a resolver [20]:

$$\frac{dy}{dx} = f(x, y) \quad (\text{A.1})$$



se conoce que la pendiente de una función en punto cualquiera (x_i, y_i) es la primera derivada de esa misma función evaluada en el mismo punto seleccionado, es decir:

$$m = f(x_i, y_i) \quad (\text{A.2})$$

Según el método de Euler, la pendiente m en el punto (x_i, y_i) es tomada como una aproximación de la pendiente promedio del intervalo $[x_i, x_{i+h}]$, definiendo a h como el paso de avance del método (escalar que se suma a la coordenada x del punto anterior). De esta forma se predice un nuevo valor de la función en x_{i+h} a través de la pendiente m en x_i , aplicando la siguiente expresión [20]:

$$y_{i+1} = y_i + m \cdot h = y_i + f(x_i, y_i) \cdot h \quad (\text{A.3})$$

Este método puede visualizarse en la siguiente Figura A.1:

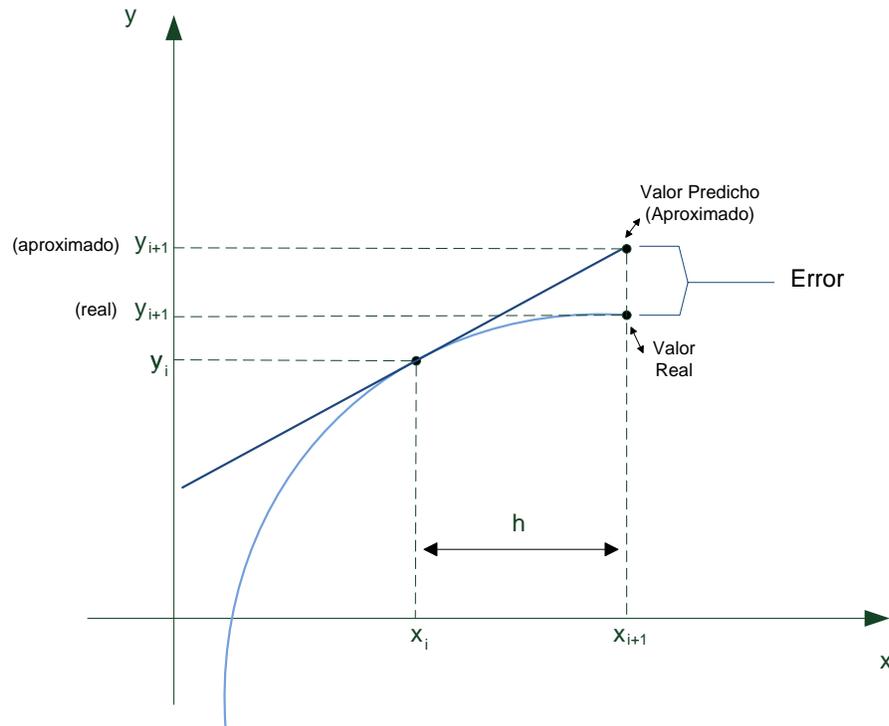


Figura A.1. Descripción del método de Euler.



El error absoluto queda definido por [20]:

$$\text{Error Absoluto} = |\text{Valor real} - \text{Valor predicho}| \quad (\text{A.4})$$

Y el error porcentual por:

$$\text{Error porcentual} = \frac{\text{Valor real} - \text{Valor predicho}}{\text{Valor real}} \times 100 \quad (\text{A.5})$$

En la Figura A.1 se muestra con más exactitud el intento de aproximación del método de Euler. Una desventaja importante del método de Euler es que el orden de exactitud es bajo. Si se quiere mantener un orden de exactitud elevado, se necesita un paso de avance h muy pequeño, lo que no sólo aumenta el tiempo de cómputo sino que también causa errores de redondeo. A continuación se presenta un método más exacto [25].

Método Runge-Kutta 4

Los métodos de Runge-Kutta se construyen a partir de un método de Taylor, suponiendo de orden N , de tal manera que se evite la evaluación de las derivadas parciales. Esto puede conseguirse a cambio de evaluar, en cada paso, la función en varios puntos. El método de Runge-Kutta de orden $N=4$, es el más popular entre la familia de métodos Runge-Kutta (de orden $N=2,3,\dots$, etc.) y es, en general, una buena elección ya que es bastante preciso, estable y fácil de programar; además, los expertos aseguran que trabajar con métodos de orden superior, que incluyen un mayor costo computacional, no compensa mayor exactitud [26]. En consecuencia, en el desarrollo y ejecución de los ejercicios planteados en este trabajo de grado se eligió el método de cuarto orden de Runge-Kutta (RK4).

El método RK4 utiliza las siguientes ecuaciones para llevar a cabo la resolución de las E.D.O. de un sistema [25].

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \quad (\text{A.6})$$



$$k_1 = hf(y_n, t_n) \quad (\text{A.7})$$

$$k_2 = hf(y_n + k_1/2, t_{n+1/2}) \quad (\text{A.8})$$

$$k_3 = hf(y_n + k_2/2, t_{n+1/2}) \quad (\text{A.9})$$

$$k_4 = hf(y_n + k_3, t_{n+1/2}) \quad (\text{A.10})$$

Considérese la gráfica $y = y(t)$ en el primer subintervalo $[t_0, t_1]$. Los valores k_1, k_2, k_3 y k_4 son aproximaciones a valores de la pendiente en algunos puntos de dicha gráfica. El valor k_1 es la derivada evaluada (pendiente) en el extremo izquierdo del subintervalo, los valores k_2 y k_3 son estimaciones de la derivada en el punto medio y k_4 es la derivada en el extremo derecho (ver Figura A.2) [26]. Estos cálculos resultan en una pendiente promedio la cual sigue siendo una estimación del valor de la pendiente real de la función solución de la E.D.O. que resuelve este método.

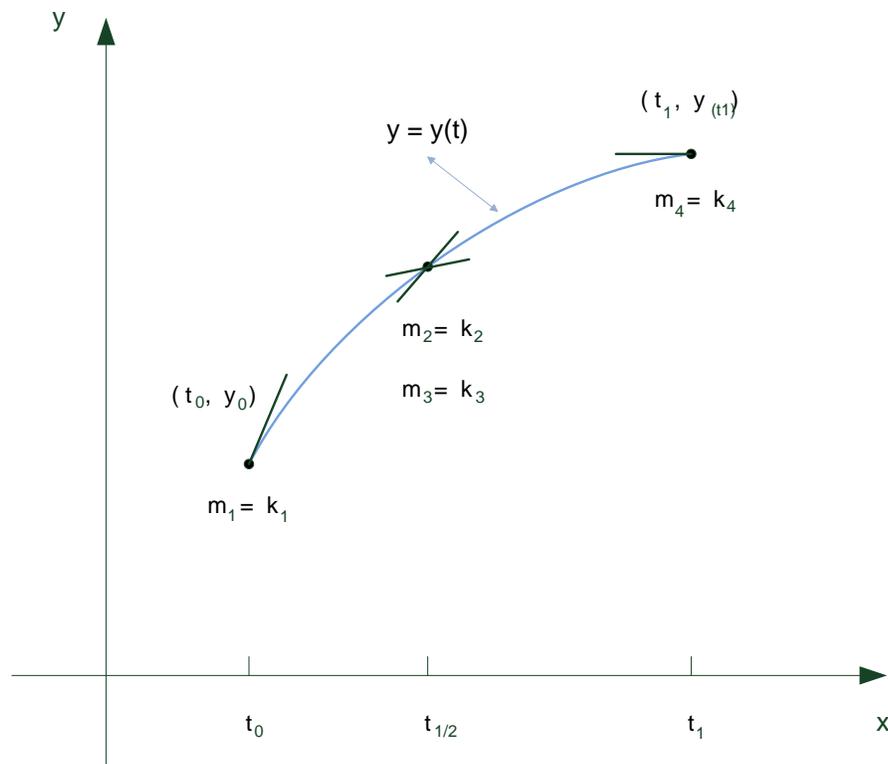


Figura A.2. Gráfica del desarrollo del método RK4.



Los métodos anteriormente desarrollados (Euler y Runge Kutta 4) concuerdan con la descripción breve que señala el archivo de ayuda de 20-sim “20-sim Help”. No obstante, el resumen descrito de cada método no es suficiente para tener una noción básica y comprensible, por esta razón se redacta este apéndice con el objetivo de mostrar al usuario cómo funcionan los métodos de integración numérica de 20-sim para solucionar los ejercicios planteados en el presente trabajo de grado.



APÉNDICE B

VARIABLES EN 20-SIM

En la sección 2.2.3.2 del Capítulo II, se describe la modelación de sistemas físicos basada en puertos. Estos puertos forman parte del submodelo (que representa un elemento del sistema físico) y es a través de ellos que los elementos, una vez conectados por medio de los enlaces (bonds), intercambian energía con el sistema. En el software 20-sim a estos puertos se les denomina **Puertos de Potencia** (*Power Ports*).

PUERTOS DE POTENCIA

Los puertos de potencia describen el flujo de la potencia y están caracterizados por dos variables. Para los grafos de enlace, se usan las extensiones ".*e*" (para designar el esfuerzo) y ".*f*" (para designar el flujo) para hacer referencia a estas variables, las cuales se pueden usar en ecuaciones (modelos de ecuaciones) escribiendo el nombre del puerto, seguido de un punto y la extensión [23], como también para elegir una variable específica en el seleccionador de variables del simulador (ver los pasos 15 y 16 de la sección 5.7 del Capítulo V).

Siempre se pueden conseguir las extensiones correctas usando el Editor de Caracteres (*Type Editor*) del Editor de 20-sim [23]. Esta ventana se abre posicionándose sobre el elemento al cual se desea conocerles las extensiones, clic derecho con el ratón y luego clic sobre *Edit Type*.

El ejemplo de la Figura B.1 muestra la ventana del Editor de Caracteres (*Type Editor*) del modelo *Friction.em* de la librería de 20-sim en la ruta de acceso: *C:\Archivos de programa\20-sim 3.6\library\Iconic Diagrams\Mechanical\Translation\Components* [23].



En el Editor de Caracteres, un puerto de potencia se define con el nombre “ p ”. Este nombre puede ser editado en el apartado *Name* del mismo Editor de Caracteres (ver Figura B.1), para esta explicación se permitirá el nombre por defecto “ p ”. Las notaciones para las variables fuerza “ F ” (esfuerzo, $.e$) y velocidad “ v ” (flujo, $.f$) de este submodelo mecánico, se denotan por las extensiones $p.F$ y $p.v$ respectivamente; en las cuales la “ p ” significa que es un puerto de potencia y las letras “ F ” y “ v ” designan el esfuerzo y el flujo respectivamente, especificados en el dominio mecánico. Esta última característica varía según el dominio físico con el que se trabaje o según la naturaleza física del submodelo.

Una vez abierto el Editor de Caracteres (*Type Editor*) presione sobre la casilla *Bond Graph* en el apartado *Type*, para obtener en el software 20-sim la Figura B.1:

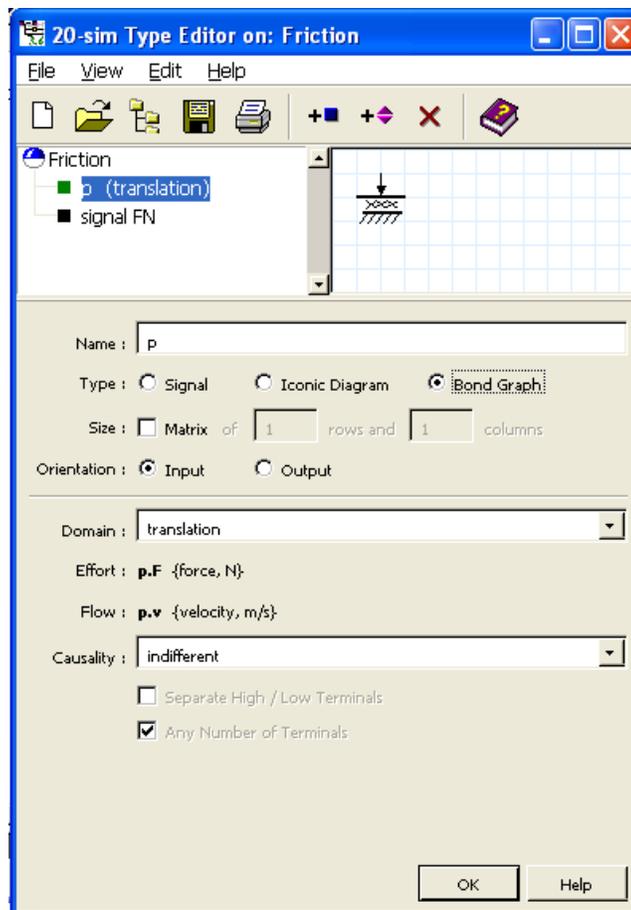


Figura B.1. Editor de Caracteres.

Las ecuaciones que describen el submodelo *Friction.em* muestran el uso de las variables $p.F$ y $p.v$, como se expresa en la Figura B.2 [23].

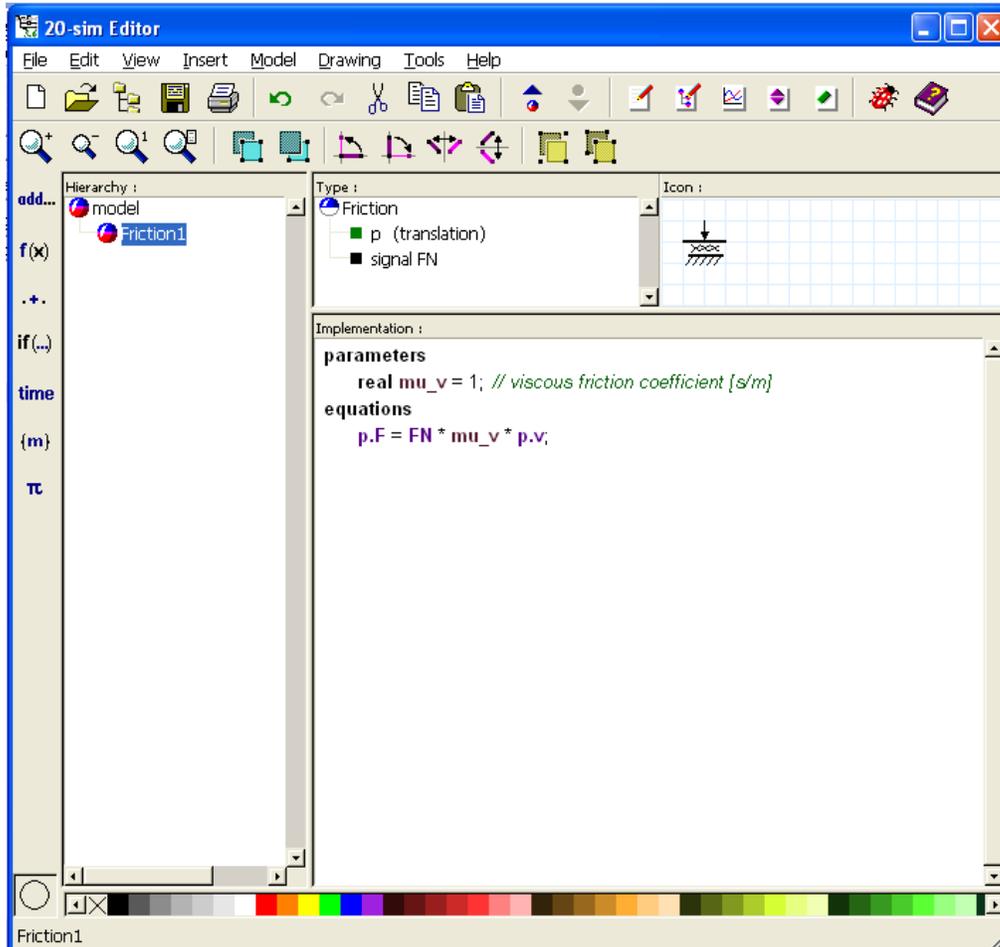


Figura B.2. Ecuaciones que describen el submodelo *Friction.em*.

APENDICE C

REGLAS DE SIMPLIFICACION PARA LA TÉCNICA DE BOND GRAPH

Este apéndice está dedicado a la tercera regla de simplificación referente a la diferencia de esfuerzo o flujo 1 para los Bond Graph cuya comprobación es más complicada de validar [6].

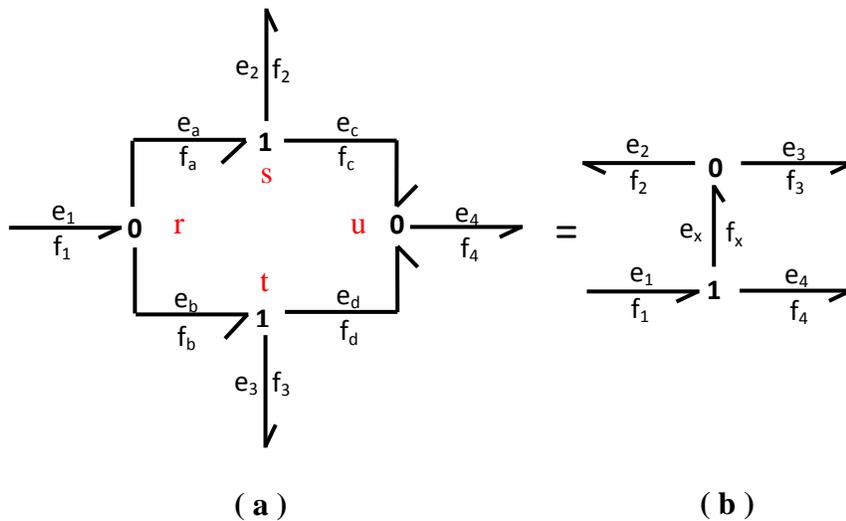


Figura C1. 3ra Regla de simplificación de los Bond Graph (a) Bond Graph sin simplificación. (b) Bond Graph simplificado.

Se procede a demostrar la igualdad mostrada en la figura c.1, comenzando con el planteamiento de las ecuaciones generadas por la Figura C1.a:

De la unión-0 (r) se tiene:

$$e_1 = e_a = e_b \tag{C.1}$$

$$f_1 = f_a + f_b \tag{C.2}$$



De la unión-1 (s) se tiene:

$$e_a = e_2 + e_c \quad (C.3)$$

$$f_a = f_2 = f_c \quad (C.4)$$

De la unión-1 (t) se tiene:

$$e_b = e_3 + e_d \quad (C.5)$$

$$f_b = f_3 = f_d \quad (C.6)$$

De la unión-0 (u) se tiene:

$$e_4 = e_c = e_d \quad (C.7)$$

$$f_4 = f_c + f_d \quad (C.8)$$

Se reordenan y reagrupan las ecuaciones de la C.1 a la C.8 para una mejor visualización.

Entonces,

$$\left\{ \begin{array}{l} e_1 = e_a = e_b \quad (C.1) \\ e_a = e_2 + e_c \quad (C.3) \\ e_b = e_3 + e_d \quad (C.5) \\ e_4 = e_c = e_d \quad (C.7) \end{array} \right.$$



Y,

$$\left\{ \begin{array}{l} f_1 = f_a + f_b \\ f_a = f_2 = f_c \\ f_b = f_3 = f_d \\ f_4 = f_c + f_d \end{array} \right. \quad \begin{array}{l} \text{(C.2)} \\ \text{(C.4)} \\ \text{(C.6)} \\ \text{(C.8)} \end{array}$$

Se sustituyen las ecuaciones C.1 y C.7 en las ecuaciones C.3 y C.5, cambiando los esfuerzos que tienen por subíndice letras por los esfuerzos con números, así:

$$e_1 = e_2 + e_4 \quad \therefore \quad e_2 = e_1 - e_4 = e_x \quad \text{(C.9)}$$

$$e_1 = e_3 + e_4 \quad \therefore \quad e_3 = e_1 - e_4 = e_x \quad \text{(C.10)}$$

De las ecuaciones C.9 y C.10 se deduce,

$$e_x = e_2 = e_3 \quad \text{(C.11)}$$

$$e_x = e_1 - e_4 \quad \text{(C.12)}$$

Ahora, se procede similarmente con las ecuaciones de flujo. Se sustituyen las ecuaciones C.4 y C.6 en las ecuaciones C.2 y C.8, cambiando los flujos que poseen subíndice de letras por los flujos con designación numérica, entonces:

$$f_1 = f_2 + f_3 = f_x \quad \text{(C.13)}$$

$$f_4 = f_2 + f_3 = f_x \quad \text{(C.14)}$$



De las ecuaciones C.13 y C.14 se deduce,

$$f_x = f_1 = f_4 \quad (\text{C.15})$$

$$f_x = f_2 + f_3 \quad (\text{C.16})$$

Ahora se procede a deducir las ecuaciones de la figura C.1.b; de la unión-0 se tienen las siguientes ecuaciones:

$$e_x = e_2 = e_3 \quad (\text{C.17})$$

$$f_x = f_2 + f_3 \quad (\text{C.18})$$

De la unión-1 se tienen las ecuaciones:

$$e_1 = e_x + e_4 \quad \therefore e_x = e_1 - e_4 \quad (\text{C.19})$$

$$f_x = f_1 = f_4 \quad (\text{C.20})$$

En consecuencia, se puede observar que las ecuaciones C.11, C.12, C.15 y C.16 corresponden (son iguales) con las ecuaciones C.17, C.19, C.20 y C.18 respectivamente, comprobando de esta forma la equivalencia entre el Bond Graph de la Figura C.1.a y el de la Figura C.1.b. Con un procedimiento similar se pueden comprobar las reglas 1, 2 y la regla 3 referente a diferencia de esfuerzo o flujo 2.

APÉNDICE D

PROCEDIMIENTO PARA LA ASIGNACIÓN DE LA CAUSALIDAD

Este apéndice tiene como objetivo principal ampliar y reforzar la asignación de la causalidad a los elementos de la técnica de Bond Graph mediante la exposición de un procedimiento elaborado por el profesor Samantaray [15]. Antes de la descripción del procedimiento es preciso definir algunos conceptos utilizados en la explicación de los pasos.

- **Enlace Fuerte y Enlaces Débiles**

En una unión 1, solamente un enlace puede traer la información de flujo; por ejemplo, solamente un enlace debe ser con extremo abierto y todos los demás deben ser con extremo cerrado por el trazo de causalidad como se muestra en la Figura D.1. Este enlace con causalidad única en la unión 1 se le denomina **enlace fuerte**. En ningún caso dos enlaces pueden ser causalmente asignados con el trazo en el extremo exterior del enlace respecto a la unión 1, ya que violaría las reglas de intercambio de información en la unión (dos enlaces no pueden comunicar flujos iguales) [15]. Similarmente en la unión 0, sólo un enlace (enlace fuerte) debe ser con extremo cerrado en el extremo interior del enlace, cerca de la unión 0 (ver Figura D.2). Este enlace fuerte determina el esfuerzo en la unión, cuyo valor los **enlaces débiles** (el resto de los enlaces que no son enlace fuerte) transportan [15].

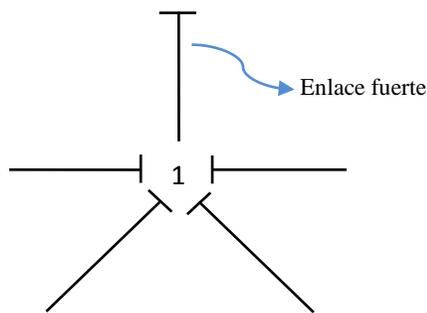


Figura D.1. Descripción enlace fuerte.

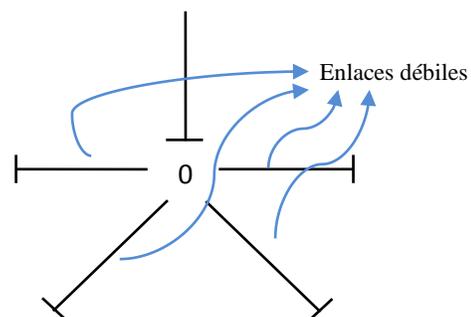


Figura D.2. Descripción enlaces débiles.



Las siguientes Tablas D.1 y D.2 sintetizan las causalidades en los elementos de la técnica de Bond Graph.

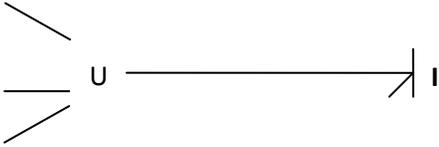
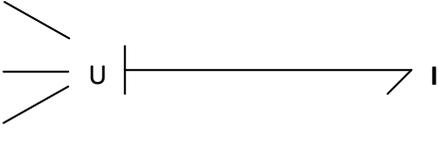
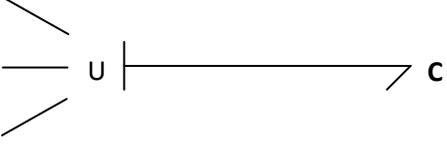
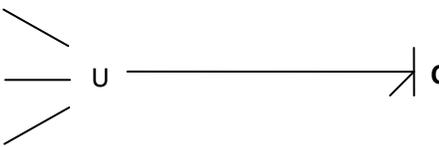
Elemento	Causalidad	Descripción
<p>S_e, S_f (fuentes)</p>		Causalidad en el elemento
		Causalidad en la unión
<p>I (inercia)</p>		Causalidad Integral
		Causalidad Diferencial
<p>C (capacitancia)</p>		Causalidad Integral
		Causalidad Diferencial

Tabla D.1. Causalidad en los elementos de Bond Graph.



Elemento	Causalidad	Descripción
Unión 1 (unión de flujo común)		
Unión 0 (unión de esfuerzo común)		
TF (transformador)		
GY (girador)		
R (resistencia)		Causalidad Resistiva Causalidad Conductiva

Tabla D.2. Continuación causalidad en los elementos de Bond Graph.



Nota: la variable U representada en la tabla D.1 y D.2 representa la Unión o Vinculo.

PROCEDIMIENTO PARA LA ASIGNACIÓN DE LA CAUSALIDAD EN MODELO DE BOND GRAPH

El siguiente procedimiento es una serie de 7 pasos que de manera organizada permiten al modelador de la técnica de Bond Graph asignar fácilmente la causalidad a los elementos de Bond Graph, según Samantaray el cual tiene una extensión de este procedimiento más detallada en su libro de texto “Bond Graph Modeling, Simulation and Fault Identification [15].

1. Asigne las causalidades fijas a las fuentes de esfuerzo y flujo existentes en el grafo de enlace (ver sección 2.2.4.3 del Capítulo II).
2. Expanda, de ser posible la asignación de la causalidad a través de las uniones. Por ejemplo, si a algún enlace se le ha asignado la causalidad de tal forma que se ha convertido en el enlace fuerte de una unión, la causalidad para los otros enlaces conectados a la misma unión (enlaces débiles) se determina por los conceptos de causalidad de las uniones (explicadas en la sección 2.2.4.3 del Capítulo II). Por el contrario, si a todos los otros enlaces de esa unión se les ha asignado la causalidad, entonces el último enlace debería ser el enlace fuerte. Similarmente, si a un puerto de algún elemento de dos puertos (TF o GY) se le asigna la causalidad, entonces la causalidad del otro puerto puede ser asignada.
3. Asigne la causalidad integral a alguno de los elementos almacenadores y extienda la asignación de la causalidad a través de las uniones. Continúe el procedimiento con el resto de los elementos almacenadores. Normalmente, al final de este paso, el resultado debería ser el grafo de enlace (bond graph) con la asignación de la causalidad completada.



4. Si el grafo de enlace aún le faltan asignaciones de causalidad, comience asignando la causalidad a un elemento **R** y extienda la asignación a los demás elementos posibles. En algunos casos, donde el modelo se determina por la asignación de la causalidad de los elementos **R**, pueden existir varias posibles asignaciones de la causalidad en el modelo. Siempre es recomendable maximizar las causalidades resistivas (asignar más) y minimizar las causalidades conductivas (asignar menos) en los elementos **R** (ver Tabla D.1. Causalidad de los elementos de Bond Graph).
5. Si el sistema desarrolla causalidad diferencial en algunos elementos almacenadores, intente minimizar su número de apariciones a través de la asignación inicial de causalidades integrales a otros elementos almacenadores que no sean los antes seleccionados.
6. Intente evitar la causalidad diferencial mediante cambios convenientes al modelo, como, por ejemplo, introducir capacitancias o resistencias, o ambas.
7. Descarte todos los modelos, que dan lugar a una estructura causal que viola las reglas de causalidad de las uniones.



APÉNDICE E

PROGRAMAS DE SIMULACIÓN DINÁMICA BASADOS EN BOND GRAPH

Este apéndice tiene como objetivo fundamental describir la evolución de los programas de simulación, las características, limitaciones, origen entre otras las cuales serán mostradas a continuación.-

THTSIM – TUTSIM

Fue desarrollado en la Universidad de Twente, Holanda a final de los años 70 siendo conocida como THTSIM en Europa y como TUTSIM en Estados Unidos.

Esta herramienta está basada en Bond Graph, y el modelo se debe introducir como una lista de números y nombres, que son transformados en el programa a una forma estado-espacio. Acepta diagrama de bloques, bonds y elementos.

El modelador tiene contacto directo con el modelo, con la simulación y con los resultados, lo que le permite interrumpir la simulación, cambiar algún parámetro, el tiempo de integración o los rangos de representación y continuar.

Las debilidades más relevantes de este paquete de software:

- No resuelve sistemas estructuralmente singulares.
- No admite ecuaciones algebraicas, por lo que es necesario introducir elementos stiff para desacoplar las coordenadas dependientes.
- No asigna causalidad automática para el modelo de Bond Graph por lo que es necesario que el modelador especifique.
- La simulación es pobre en relación a otros simuladores.

CAMAS-20-SIM

CAMAS (Computer Aided Modeling, Analysis and Simulation) fue desarrollado por el mismo grupo de investigación de THTSIM-TUTSIM a principio de los años 90.



Está compuesto de dos partes, En la primera se realiza la edición del modelo a través de un intérprete, bien sea una entrada interactiva de Bond Graphs, o bien un texto generado en SIDOPS, y la segunda para la simulación, la grabación de la descripción del modelo en una base de datos y la preparación de los modelos para su simulación. Analiza la causalidad del modelo por medio de un algoritmo automático.

En la segunda parte, se realiza la simulación y el procesamiento opcional de sus resultados. Admite ecuaciones de restricción y determina las coordenadas dependientes en función de la asignación causal. Viene a ser el sucesor del programa CAMAS, sigue utilizando SIDOPS como lenguaje de programación y opera en Windows, permite la simulación de sistemas dinámicos pertenecientes a múltiples dominios físicos mediante el empleo de iconos, diagramas de bloques, grafos y ecuaciones.

En 20-Sim es necesario compilar el modelo para llevar a cabo la simulación, el proceso de compilación se ejecuta mediante una herramienta la cual lleva incorporada el software. Además contiene un modulo mediante el cual se pueden construir y simular mecanismos de forma tridimensional.

CAMP-G

CAMP-G, (Computer Aided Modeling Program with Graphical input) fue desarrollado en la Universidad de California a principio de los 90, con la finalidad de convertir un modelo físico representado mediante Bond Graph en una serie de datos validos para lenguajes de simulación digitales (DSL), tales como IBM DSL (Digital Simulation Language), CSMP (Continuous System Modeling Program), ACSL (Advanced Continuous Simulation Language), CSSL (Continuous System Simulation Language), con aplicaciones externas como MATLAB. Una vez que el Bond Graph ha sido creado, el programa genera una descripción del sistema, la cual puede ser aceptada por un lenguaje de simulación como los descritos anteriormente.

Esta herramienta de modelización genera un código fuente que contiene la representación matemática del sistema mediante las ecuaciones diferenciales de primer



orden, las condiciones iniciales, las condiciones de integración y los ficheros necesarios para las salidas graficas y numéricas.

Permite simular sistemas lineales y no lineales, aunque en los no lineales los elementos deben ser editados manualmente en el programa ACSL generado, lo cual resulta complejo.

SYMBOLS 2000

SYMBOLS (System Modeling by BONDgraph Language and Simulation), fue desarrollado por el Indian Institute technology a finales de los 90, la primera versión permitía trabajar bajo windows y MS-DOS. En la actualidad trabaja únicamente bajo windows.

Está conformado por cuatro módulos:

- El Editor, permite crear el diseño del modelo mediante el diagrama de grafos, iconos, bloques y conjuntos de ecuaciones y en Segundo lugar, genera el código necesario para su posterior simulación y control.
- El Simulador, el cual permite simular las ecuaciones obtenidas y procesar la información.
- El Control, permite realizar el análisis y síntesis de los sistemas de control.
- El de Animación, por medio del cual se representan los resultados obtenidos.

Cuenta con Microsoft Developer Studio para compilar el modelo antes de simularlo, creando una librería dinámica con el código necesario para llevarlo a cabo, asigna la causalidad a los modelos y permite trabajar con bucles algebraicos de forma simbólica.

BONDYN

Fue desarrollado en la Universidad Politécnica de Madrid a principio de los años 90. Consta de dos módulos, el primero permite definir el modelo de forma grafica a través de menús y ventanas, y el segundo mediante el cual se genera un fichero de código FORTRAN, el cual será compilado antes de llevar a cabo la simulación.



Su principal contribución con respecto a otros programas que trabajan en MBS es el de uso de multiplicadores de Lagrange, los multiplicadores consisten en muelles rígidos e inercias nulas. Los primeros simulan muelles con rigidez infinita, sin deformación y con velocidades relativas nulas en sus extremos. Por otra parte, las inercias nulas presentan momento lineal nulo y suponen una restricción adicional. Ambas restricciones dan lugar a ecuaciones algebraicas entre variables de estado.

El modelo en Bond Graph se construye ensamblando subsistemas, en el proceso de ensamblado se requiere de un conjunto de bonds comunes en los submodelos a unir. Este proceso permite ensamblar submodelos de diferentes dominios energéticos (mecánico, eléctrico, hidráulico, etc.). Una vez ensamblado el sistema, el programa construye el sistema de ecuaciones de estado de primer orden y tras ser compilado, corre de forma conjunta con una serie de librerías, las cuales contienen algoritmos numéricos necesarios para la integración de las ecuaciones diferenciales.

DYMOLA

DYMOLA (DYnamic MOdeling LABoratory) fue desarrollado en Suecia a mediados de los 90, por el Research Park Ideon. Consiste en un programa de modelado y simulación de propósito general desarrollado en el lenguaje orientado a objetos MODELICA para describir sistemas, subsistemas y componentes de un modelo.

Consta de varios módulos:

- DYMODRAW es una interfaz gráfica basada en editores de objetos, crea y edita gráficamente los modelos, con sus submodelos y conexiones.
- DYMOVIEW es una interfaz gráfica, que permite visualizar con animación el comportamiento dinámico del sistema, se basa en objetos gráficos predeterminados tales como cilindros, cajas, conos, esferas, tubos y vigas.
- DYMOSIM, permite resolver sistemas tipo DAE o ODE.

La construcción de modelos se hace de forma jerarquizada, a partir de librerías desarrolladas en dicho lenguaje. Descomponiendo el sistema en distintos submodelos y posteriormente acoplándolos mediante conexiones.



Luego de ser descrito el modelo, DYMOLA genera un sistema de ecuaciones en distintos formatos para ser resuelto por diferentes solvers como ACSL, Simulink o Desire. Es capaz de generar subrutinas directamente en código FORTRAN.

CAMBAS

CAMBAS (Computer Aided Model Building Automation System) fue creado en la Universidad de Michigan a principio de los años 90, por el Automotive Research Center.

Consiste en un generador de modelos a base de bloques que permiten por medio de una interface grafica basada en menús desplegables, describir un sistema como conjunto de distintos componentes interconectados entre sí, pudiendo definirlos, colocarlos, editarlos, borrarlos e interconectarlos como se desee. Dichos elementos son modelados mediante Bond Graph y guardados en librerías. Adicionalmente, se puede representar el Bond Graph correspondiente a todo el modelo. Es una herramienta limitada, con posibilidades de simulación detallada demasiado pobres.



FORMULACIÓN DEL PROBLEMA

1.1 PLANTEAMIENTO DEL PROBLEMA

Como consecuencia de los nuevos avances de la tecnología y las exigencias de la industria en reducir tiempos y costos, se hace necesario optimizar y controlar los procesos que involucran la dinámica de la energía a través de la modelación y simulación, las cuales son herramientas que tienen como objetivo la generación de un conjunto de ecuaciones y gráficos que permiten conocer y describir el comportamiento dinámico de diversos sistemas físicos, tornándose de gran utilidad cuando se requiere realizar el análisis y diseño de sistemas multidisciplinarios que involucran el desarrollo de un proceso de transformación de la energía. Algunas de estas herramientas para la modelación y simulación son: redes neuronales, redes de Petri, lógica difusa, memorias asociativas, agentes cooperativos, modelos algorítmicos, modelos lingüísticos y bond graphs (grafos de enlace) [1].

La primera versión de una técnica conocida como Bond Graph se desarrolla a mediados del siglo XX. Esta herramienta permite la modelación y simulación de la dinámica de sistemas, independientemente del dominio al cual pertenezcan (eléctrico, mecánico, electromecánico, térmico, etc.), permitiendo estudiar las propiedades energéticas de un sistema en los diversos dominios físicos a través del análisis en la potencia y la energía del sistema. Los grafos de enlaces se han convertido en un importante tema de investigación, especialmente en una época en la cual las aplicaciones mecatrónicas y de control avanzado se realizan con la integración de subsistemas provenientes de diferentes dominios [1].

Académicamente los alumnos de la Escuela de Ingeniería Eléctrica de la Universidad de Carabobo, no cuentan con alguna asignatura ofrecida en esta carrera que permita aprender Bond Graph a pesar de ser una herramienta relativamente sencilla y que proporcionaría un gran aporte para el estudio, modelación y simulación de los sistemas



físicos que involucran dinámica de la energía, ya que es una técnica que grafica los subsistemas, siendo éstos representados por nodos y los flujos de energía en el sistema por ramas [2]. Con esta técnica de Bond Graph se pueden modelar sistemas de iguales dominios físicos y también de distintos dominios físicos involucrados en el mismo sistema.

Por lo tanto, se propone este trabajo especial de grado, el cual tiene como finalidad crear una herramienta (o software) eficiente y didáctica para la enseñanza – aprendizaje de la técnica de Bond Graph con aplicaciones en modelación y simulación de sistemas lineales, desarrollando las simulaciones en ambiente de programación del software 20-SIM [3]. La utilización de este software didáctico contribuirá y apoyará las nuevas y actuales estrategias educativas de nuestra institución, facilitando el estudio de las materias relacionadas con este tema, tales como: Teoría de Control I, Teoría de Control II, Teoría de Control III, Simulación, entre otras, ofrecidas por el Departamento de Sistemas y Automática de la Escuela de Ingeniería Eléctrica de la Facultad de Ingeniería de la Universidad de Carabobo.

1.2 JUSTIFICACIÓN DE LA INVESTIGACIÓN

Debido al continuo incremento y la velocidad en la que avanza el desarrollo tecnológico en áreas tales como: Procesos Industriales de Manufactura, Mecatrónica, Control Avanzado, Electrónica de Potencia, Biología, Ambiente, Sistemas de Potencia, entre otros; y teniendo en cuenta la alta competitividad de las industrias y el costo en tiempo y dinero que resulta del desarrollo de prototipos, entonces conceptos tales como modelación de un sistema, adquieren gran importancia hoy en día, debido a que la modelación es una “técnica para el análisis y el diseño de un sistema basada en idealizaciones matemáticas o físicas de todo el sistema o una parte del mismo” [4]. Considerando que los sistemas en ingeniería requieren un análisis en su comportamiento dinámico (o en operación) y que la mayoría de sus componentes están interconectados y pertenecen a diferentes dominios de la física, se hacen muy complejos los modelos que si bien nos proporcionan mayor exactitud, también nos proveen una gran dificultad de



manipulación, medición y evaluación práctica de sus parámetros. En consecuencia, se han desarrollado diferentes representaciones gráficas de modelos en la búsqueda de herramientas que permitan establecer una relación práctica entre exactitud y sencillez tales como: representaciones esquemáticas, diagramas de bloques, diagramas de flujos de señales y grafos de enlace; siendo este último una de las mejores alternativas debido a que la técnica conserva la topología original del sistema modelado y cuenta con simplicidad visual. La técnica de Bond Graph está relacionada con la noción de energía, no depende del tipo del sistema modelado, permite la inmediata obtención de las ecuaciones de estado del sistema, conduce directamente a la simulación del sistema, se utiliza en el estudio de la controlabilidad y observabilidad de sistemas lineales, etc. [5].

Por último, es conveniente recalcar que es necesario, para todo Ingeniero Electricista que desee estar actualizado con los nuevos avances tecnológicos contar con material teórico y práctico que lo motive al aprendizaje de tan importante técnica, aún más cuando en muchas Universidades del mundo, los *Bond Graphs* han sido incorporados a los pensum de estudios como en la Universidad de Twente, Holanda.

1.3 OBJETIVOS DE LA INVESTIGACIÓN

1.3.1 OBJETIVO GENERAL

Diseñar un software didáctico para la enseñanza-aprendizaje de métodos y técnicas de BOND GRAPH con aplicaciones en modelación y simulación de sistemas lineales en ambiente de programación 20-SIM

1.3.2 OBJETIVOS ESPECIFICOS

- ✓ Describir la teoría de Bond Graph para la modelación de sistemas en diferentes dominios (eléctrico, térmico, mecánico, hidráulico, etc.)



- ✓ Aplicar la técnica de Bond Graph para la resolución de problemas en modelación y simulación de sistemas lineales de primer orden, en diferentes dominios físicos.
- ✓ Desarrollar el Software Didáctico Multimedia en Ambiente Dreamweaver, con un manual para el usuario explicando las diferentes secciones del trabajo.
- ✓ Desarrollar un Manual/Tutorial, sobre el Software 20-SIM explicando diferentes ejemplos de Programación donde se realiza la simulación de diversos sistemas aplicando las Técnicas de Bond Graph.

1.4 ALCANCE

Se presenta el siguiente trabajo especial de grado mediante un software didáctico para la enseñanza aprendizaje de la técnica de Bond Graph, como una herramienta para la modelación y simulación de diferentes sistemas físicos.

El contenido de este trabajo se desarrolla en tres partes que se describen con esquemas, dibujos y animaciones; la primera parte ofrecerá una orientación teórica presentando la metodología y los elementos fundamentales para el óptimo desarrollo de la técnica de Bond Graph; en la segunda parte se explican detalladamente casos de modelación de sistemas con diferentes dominios físicos, partiendo de un nivel de fácil comprensión hasta modelos de mayor dificultad, y en la tercera parte se realizan las simulaciones de los problemas antes planteados con el software 20 SIM.



2.1 ANTECEDENTES DEL PROBLEMA

Debido a que la mayoría de los componentes de los sistemas en la ingeniería están interconectados, y pertenecen a diferentes dominios físicos (mecánico, eléctrico, hidráulico, térmico, etc.), se hace necesario integrar un proceso de diseño que conlleve a la solución. Por lo tanto, la técnica de BOND GRAPH es particularmente útil en la modelación y simulación de la dinámica de sistemas multidisciplinarios debido a que sus ventajas radican en que se puede pasar fácilmente de un dominio a otro, a través de sus diversos elementos Transformador, Girador, vínculo 0, vínculo 1, etc. [6]. Además se puede modificar fácilmente el modelo de un sistema añadiendo y/o quitando elementos de Bond Graph parcialmente sin la necesidad de deshacer el modelo completo. Estos elementos a su vez permiten establecer y visualizar de manera gráfica las relaciones matemáticas subyacentes en los modelos de sistemas utilizados en ingeniería [7].

La técnica de los grafos de enlace se desarrolla a nivel mundial desde 1959 cuando H. M. Paynter [1] crea la primera versión de sus esquemas de Bond Graph, logrando un avance en el estudio académico en técnicas de modelación. A continuación se señalan los recursos bibliográficos (trabajos de grado, tesis doctorales, publicaciones de revistas, etc.) que son fuentes de información en la enseñanza de esta rama innovadora, y además sirven como apoyo para el desarrollo del presente trabajo de grado.

- Trabajo realizado por José Gregorio Díaz; César Peraza y Francisco Arteaga en el año 2005, publicado por el Centro de Investigación y Tecnología en Automatización Electrónica y Control (CITAEC). UNIVERSIDAD DE CARABOBO, [8], donde los autores presentan los fundamentos de la técnica de una manera sencilla y realizan la modelación y simulación de algunos casos de sistemas lineales eléctricos, mecánicos y electromecánicos, dándole una introducción al software de simulación 20 SIM, apoyados del trabajo de F. Cellier titulado Modelado Orientado a Objetos de Sistemas Físicos [9].



- Trabajo de grado: Modelación y simulación de convertidores DC-DC y redes de atraso de fase usando técnicas de Bond Graph y Software 20-SIM, [3], donde sus autoras Johana Marques y Deyanira Vargas [10] analizan la técnica de Bond Graph para el modelado de sistemas eléctricos y realizan la simulación de los convertidores DC-DC y del lazo de control realimentado en el software 20-SIM y en el software MATLAB a fin de verificar las respuestas obtenidas en los modelos.
- Trabajo de grado: Obtención y simplificación de ecuaciones de estado definitivas a partir de un Bond Graph, realizado por Antonio Bosnjak Seminario [5], donde el autor propone una forma sencilla de obtener las ecuaciones de estado para diferentes sistemas.
- Tesis doctoral: Procedimientos optimizados utilizando métodos simbólicos para la simulación de sistemas dinámicos mediante Bond Graph, donde su autor Gregorio Romero Rey [4], fundamenta su trabajo en la descripción de la capacidad que tiene la causalidad (aspecto fundamental para la aplicación de la técnica) para generar los modelos matemáticos de cualquier tipo de sistema en la ingeniería.
- Trabajo publicado por Laboratorio de Sistemas Oleohidráulicos y Pneumáticos, Universidad Politécnica de Cataluña: Técnicas de modelación y simulación, donde se presentan los aspectos más importantes de la modelación y los diferentes softwares que acompañan a cada técnica para verificar los resultados de manera práctica [11].
- Libro de texto: Bond Graph in Modeling, Simulation and Fault Identification, en el cual sus autores A. Mukherjee, R. Karmakar y A.K. Samantaray exponen de forma extensa la trayectoria de sus últimos 15 años en la enseñanza de la modelación, la dinámica y el control de sistemas con grafos de enlace [2].



2.2 MARCO CONCEPTUAL

2.2.1 INTRODUCCIÓN

Para la realización de este trabajo de grado es importante comenzar presentando un desarrollo teórico que defina claramente en qué consisten los grafos de enlace y lo beneficioso que pudiera ser tener una herramienta que nos permita realizar la modelación de cualquier sistema físico mediante la interconexión de un conjunto de elementos simples.

La técnica de Bond Graph, es una herramienta gráfica que permite el análisis y la modelación de sistemas físicos independientemente de su naturaleza (mecánico, térmico, eléctrico, hidráulico, etc.), también es una herramienta que puede ser utilizada para el diagnóstico de fallas en sistemas de control de procesos en el área de Ingeniería [12].

De una manera más simplificada, se puede decir que los grafos de enlace constituyen una representación gráfica de un sistema físico, igual que lo son los diagramas de bloques o los diagramas de flujo, pero con la diferencia que los Bond Graph representan la información (cambios de energía) de manera bi-dimensional [13].

2.2.2 ORÍGENES DEL BOND GRAPH

La idea de los Bond Graph, surge debido a la necesidad de tener una herramienta, con la cual se pueda estudiar y analizar sistemas de diferentes dominios, por lo tanto el concepto de los grafos de enlace o Bond Graph se origina en el año de 1959 por H.M. Paynter (1923-2002) [1], quien luego fuera profesor en el Departamento de Ingeniería Mecánica del Instituto de Tecnología de Massachusetts MIT, al observar las analogías entre los sistemas eléctricos y mecánicos de una central hidroeléctrica. Posteriormente, la idea es desarrollada por sus estudiantes, C. D. Karnopp (más tarde profesor en UC Davis, California) y R. C. Rosenberg (más tarde profesor de la Universidad Estatal de Michigan, Michigan) quienes añaden una publicación la cual es muy completa, en esta además de los conceptos básicos incorporan la aplicación de esta técnica a distintos dominios físicos y su relación con la teoría clásica de vibraciones y de control. Por su parte P. Breedveld y F. Lorenz realizan un importante trabajo tratando de estandarizar la nomenclatura y formulación multibond (multi-enlace) tanto para Bond Graph unidimensionales como



multibond [4]. Estos pioneros y sus estudiantes han estado difundiendo sus ideas en todo el mundo.

2.2.3 CONCEPTOS Y DEFINICIONES

Teniendo en cuenta que un esquema de Bond Graph puede ser representado en diferentes dominios físicos, dos componentes son modelados por el mismo elemento *Bond Graph* si procesan la energía de la misma forma. Este hecho motiva a la generalización de las variables físicas: *esfuerzo e* y *flujo f* (*variables de potencia*, su producto da potencia), e *impulso p* y *desplazamiento q* (*variables de energía*, son las respectivas integrales de las dos anteriores) [24].

Variables de Potencia: Se conocen como variables de potencia, ya que la Potencia se expresa como el producto de la variable de esfuerzo y la variable de flujo, representado por:

$$\mathbf{Potencia} = \mathbf{e}(t) \cdot \mathbf{f}(t) \quad (2.1)$$

Variables de energía o dinámicas: Desplazamiento e integral del esfuerzo en el tiempo, el desplazamiento se define mediante la ecuación (2.2) que expresa la integral del flujo con respecto del tiempo. Y la integral del esfuerzo en el tiempo, que en mecánica se expresa como la cantidad de movimiento se expresa mediante la ecuación (2.3) [24].

$$\mathbf{q}(t) = \int_{t_0}^t \mathbf{f}(t) dt = \mathbf{q}_0 + \int_{t_0}^t \mathbf{f}(t) dt \quad (2.2)$$

$$\mathbf{P}(t) = \int_{t_0}^t \mathbf{e}(t) dt = \mathbf{p}_0 + \int_{t_0}^t \mathbf{e}(t) dt \quad (2.3)$$

Tanto la ecuación (2.2) como la ecuación (2.3), ambas permiten expresar las variables de energía de forma diferencial, dando como resultado las ecuaciones (2.4) y (2.5)



$$\frac{dp(t)}{dt} = e(t) \tag{2.4}$$

$$\frac{dq(t)}{dt} = f(t) \tag{2.5}$$

Y la ecuación (2.6) muestra que la energía transmitida por el bond (enlace), $E(t)$ es la integral de la potencia en el tiempo:

$$E(t) = \int^t Potencia dt = \int^t (e(t) \cdot f(t))dt \tag{2.6}$$

En la Tabla 2.1 se ilustra una generalización posible, con las correspondientes variables en cada dominio físico.

Tabla 2.1. Generalización de variables en un dominio físico

Dominio Físico	Variables de Potencia		Variables de energía	
	Esfuerzo e	Flujo f	Momento p	Desplaz. q
Eléctrico	Voltaje	Corriente	Flujo concatenado	Carga
Traslación	Fuerza	Velocidad	Impulso	Desplaz.
Rotación	Momento	Velocidad angular	Momento angular	Angulo
Fluido dinámico	Presión	Caudal	Impulso del fluido	Volumen
Electromagnetismo	Tensión	Corriente	Flujo magnético	Carga eléctrica
Química	Potencial químico	Flujo molar		Número de moles
Termodinámica	Temperatura absoluta	Flujo de entropía		Entropía
Hidráulico	Presión	Caudal volumétrico	Momento	Volumen

Fuente: [10] [14].



En la técnica de Bond Graph se necesitan reconocer cuatro grupos de símbolos básicos, a saber, tres elementos pasivos de un puerto (elementos R, C, I), dos elementos activos (SE y SF), dos elementos de dos puertos (TF y GY) y dos nodos (unión 0 y unión 1) [15].

2.2.3.1 MODELACIÓN BASADA EN PUERTOS

Fundamentalmente hay que tener en cuenta que para la modelación basada en puertos debemos conocer dos conceptos básicos como lo son el puerto (port) y el enlace (Bond), para introducirlos mejor en el tema consideraremos un ejemplo sencillo de un circuito como el mostrado en la Figura 2.1 [8].

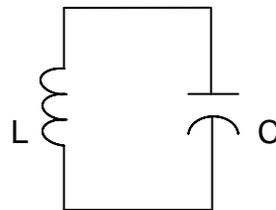


Figura 2.1. Circuito LC.

Este es un circuito LC en el cual la salida para la representación de estado en el inductor es el voltaje y en el capacitor es la corriente, haciendo un análisis de este sistema y llevándolo a su representación en puerto tenemos un modelo como el mostrado en la Figura 2.2.

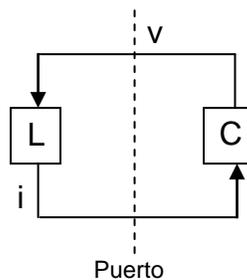


Figura 2.2. Modelo LC con puerto.



En la Figura 2.2, los elementos interactúan a través del puerto por medio del cual comunican señales de corriente y tensión, en el inductor la entrada es la tensión y la salida es la corriente mientras que para el capacitor la entrada es la corriente y la salida la tensión, el producto de estas señales es la potencia de intercambio en el puerto.

Una simplificación de la Figura 2.2 nos lleva a obtener el bond de este modelo en el cual se aprecian las señales que participan en la interacción entre el inductor y el capacitor, el bond no es más que una línea la cual une los subsistemas como podemos apreciar en la Figura 2.3 [8].

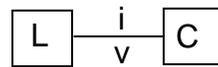
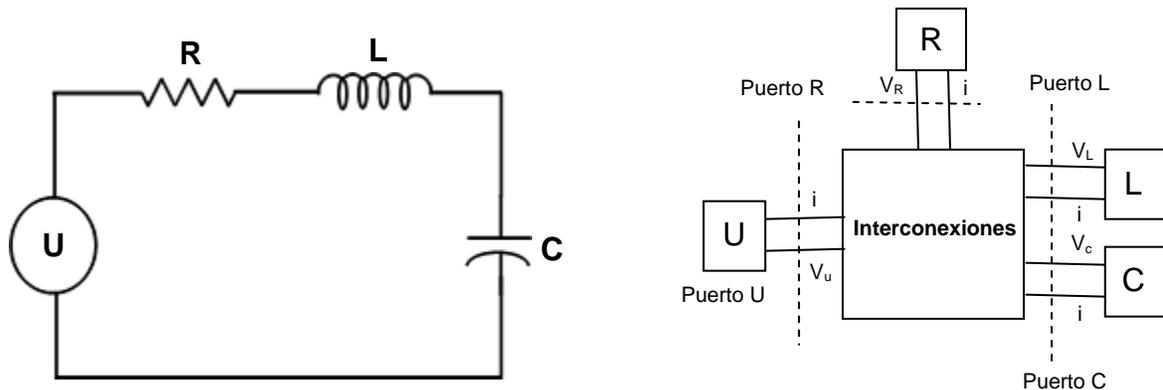


Figura 2.3 Bond modelo basado en puertos.

Cuando es el caso de un sistema más complejo el diagrama equivalente debe incluir más puertos, en donde los puertos intercambian energía a través de interconexiones más complejas un ejemplo de esto puede ser mostrado en la Figura 2.4 [8].



Circuito RLC serie.

Circuito RLC Serie Modelo en Puertos.

Figura 2.4. Modelo en Puertos del Circuito RLC.



La modelación basada en puertos puede hacerse, independientemente del dominio al cual pertenezca el sistema, ya que se utilizan términos más globales como el esfuerzo, flujo, desplazamiento y cantidad de movimiento [8].

2.2.3.2 ELEMENTOS DE BOND GRAPH

Para los grafos de enlaces, se requieren de cuatro grupos de elementos básicos tales como los elementos pasivos de un puerto, elementos activos de un puerto, elementos activos de dos puertos y elementos de tres puertos o más representados por dos nodos (uniones) o vínculos. Las variables básicas son el esfuerzo (e) y el flujo (f), la integral en el tiempo del esfuerzo (P) y la integral en el tiempo del flujo (Q) [15].

2.2.3.2.1 ELEMENTOS PASIVOS DE UN SOLO PUERTO:

- **Elemento R:**

El resistor de un puerto es un elemento en el cual las variables flujo y esfuerzo en el puerto único se relacionan por una función estática. Estos consumen energía del sistema y la convierten en calor.

Su representación en Bond Graph es la siguiente, tal como se muestra en la Figura 2.5:

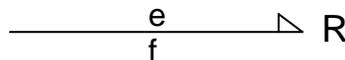


Figura 2.5. Grafo del Elemento Resistivo.

El esfuerzo en un elemento R se puede calcular a partir de (2.7) mientras que el flujo a partir de (2.8).

$$e(t) = R \cdot f(t) \quad (2.7)$$



$$f(t) = \frac{1}{R} \cdot e(t) \quad (2.8)$$

Este elemento es utilizado para modelar los efectos de los resistores en circuitos eléctricos, amortiguadores en sistemas mecánicos y tapones porosos en líneas de fluido.

▪ **Elemento C:**

Es un dispositivo de un puerto en el cual existe una relación elemental entre un esfuerzo y un desplazamiento, tal dispositivo ideal almacena y entrega energía sin pérdidas.

Su representación en Bond Graph es la mostrada en la Figura 2.6:

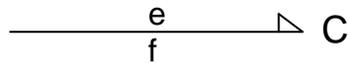


Figura 2.6. Grafo del elemento capacitivo.

La relación entre esfuerzo y flujo viene dada por la ecuación (2.9) y ecuación (2.10).

$$q = \int f dt \quad (2.9)$$

$$e = \varphi(p) \quad (2.10)$$

Este elemento es utilizado para modelar los efectos de las capacitancias en circuitos eléctricos, resortes en sistemas mecánicos, barras de torsión, acumuladores y tanques de gravedad y otros elementos pasivos análogos [15].



- **Elemento I:**

Es un elemento de almacenamiento de energía de un puerto que relaciona por una ley estática elemental el momento p con el flujo f , en estos la entrada es un flujo y la salida un esfuerzo [15].

La representación en Bond Graph es mostrada en la Figura 2.7:

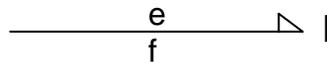


Figura 2.7. Grafo del Elemento Inercial.

El unipuerto I es un elemento en el cual la relación entre flujo y esfuerzo viene dada por la ecuación (2.11) y ecuación (2.12) [8].

$$p = \int e dt \quad (2.11)$$

$$f = \varphi(p) \quad (2.12)$$

El elemento inercial se utiliza para modelar los efectos de la inductancia en circuitos eléctricos y los efectos de la masa o inercia en sistemas mecánicos o de fluidos.

2.2.3.2.2 ELEMENTOS ACTIVOS DE UN SOLO PUERTO:

- **Fuentes de Esfuerzo y Flujo:**

Las fuentes son elementos capaces de suministrar energía al sistema, en los grafos de enlaces son también llamadas puertos activos.



Se consideran dos tipos de fuentes ideales:

1. **Fuente de Esfuerzo, *Source of Effort* (SE):** son aquellas que provocan una entrada de esfuerzo (voltaje, fuerza o presión) al sistema su representación en Bond Graph es mostrada en la Figura 2.8 [10].

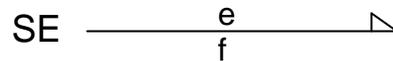


Figura 2.8. Grafo fuente de esfuerzo.

2. **Fuente de Flujo, *Source of Flow* (SF):** Son aquellas que suministran flujo (corriente, velocidad o caudal) al sistema su representación en Bond Graph es la mostrada en la Figura 2.9 [10].

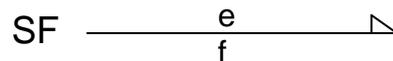


Figura 2.9. Grafo fuente de flujo.

2.2.3.2.3 ELEMENTOS DE DOS PUERTOS:

- **Transformador (TF):**

El transformador de Bond Graph conserva potencia y transmite los factores de potencia con la escala apropiada y definida por el módulo (r) del transformador. Estos poseen dos enlaces o puertos para poder representar a los dispositivos que transforman energía y sirven para representar de una manera ideal una palanca rígida, un par de



engranajes, un transformador eléctrico, etc. Su representación en bond graph es mostrada en la Figura 2.10 [10].

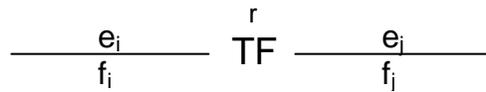


Figura 2.10. Grafo transformador.

Las relaciones entre los esfuerzos y flujos de entrada y salida vienen dadas por las ecuaciones:

$$e_i = r e_j \tag{2.13}$$

$$r \cdot f_i = f_j \tag{2.14}$$

Donde la “r” sobre el transformador denota el modulo del transformador, la cual puede ser una constante o cualquier expresión.

- **Girador (GY):**

Este elemento establece la relación de flujo a esfuerzo y de esfuerzo a flujo, manteniendo la potencia igual en los puertos, donde μ representa la ley de relación entre esfuerzo y flujo, su representación en bond graph es mostrada en la Figura 2.11 [15].

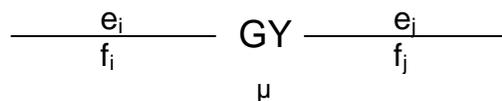


Figura 2.11. Grafo girador.



En los giradores las relaciones entre esfuerzo y flujo están dadas por las ecuaciones:

$$e_i = \mu \cdot f_j \quad (2.15)$$

$$\mu \cdot f_i = e_j \quad (2.16)$$

2.2.3.2.4 ELEMENTOS DE TRES PUERTOS O MÁS:

- **Unión-0 o Vínculo-0:**

En este tipo de nodo los grafos que entran y salen del nodo llevan asociado el mismo esfuerzo. Es también llamado vínculo 0 de esfuerzo común para conexión en paralelo su representación viene dada por la Figura 2.12 [10].

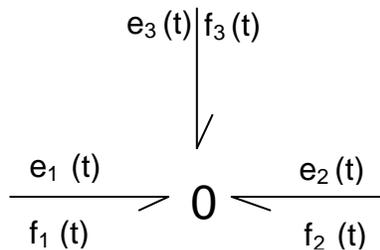


Figura 2.12. Representación Unión-0 o Vínculo-0.

Las relaciones de interconexión que describen una unión de flujo están dadas por:

$$e_1(t) = e_2(t) = e_3(t) \quad (2.17)$$

$$f_1(t) + f_2(t) + f_3(t) = 0 \quad (2.18)$$



Estas ecuaciones especifican que en una unión-0, los esfuerzos en todos los bonds son siempre iguales y la suma algebraica de los flujos es cero.

▪ **Unión-1 o Vínculo-1:**

En este tipo de nodo se mantiene constante el flujo de los grafos que entran y salen de la unión, también llamado vínculo 1 o de flujo común para la conexión en serie. [10].

Esta unión representa un punto de masa común en un sistema mecánico, una conexión en serie (con la misma corriente circulando en todos los elementos) en una red eléctrica y una tubería hidráulica que representa la continuidad del flujo, etc [15]. Su representación en Bond Graph es mostrada en la Figura 2.13.

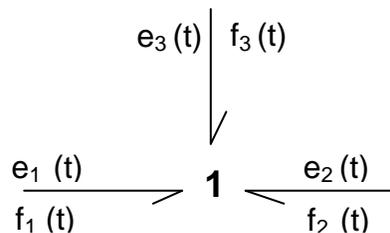


Figura 2.13. Representación Unión-1 o Vínculo-1.

Las relaciones de interconexión para este elemento son:

$$f_1(t) = f_2(t) = f_3(t) \tag{ 2.19 }$$

$$e_1(t) + e_2(t) + e_3(t) = 0 \tag{ 2.20 }$$

A diferencia de la unión esfuerzo, la unión de flujo tiene un flujo común en los enlaces que inciden en la unión y la suma de las variables de esfuerzo en los enlaces es cero.

Los puertos de conexión de esfuerzo y flujo pueden extenderse a puertos de 3 o más puertos.

2.2.3.2.5 ELEMENTOS MULTIPUERTOS:

Los elementos multipuertos no son más que la unión de los diferentes elementos que tenemos ya en Bond Graph los cuales se agrupan para representar un fenómeno el cual no es posible representar con un elemento unipuerto, como ejemplo de este se tiene el caso de la inductancia mutua mostrada en la Figura 2.14.

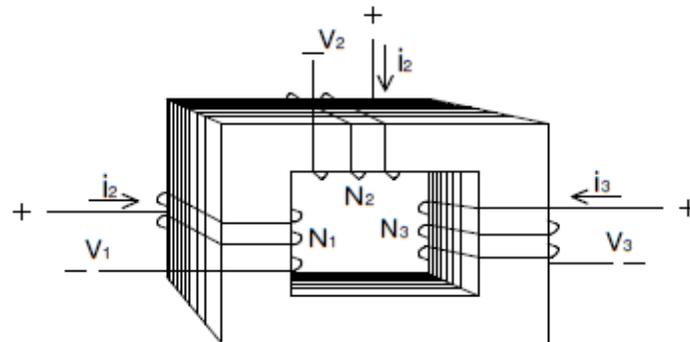


Figura 2.14. Ejemplo Elemento Multipuerto: Inductancia mutua.

Cada devanado posee una inductancia propia L_i pero interactúa con los otros dos devanados por inductancia mutua, a esta representación se le asocia una matriz que vincula el vector de flujos magnéticos mostrado en la ecuación (2.21) con el vector de corrientes mostrado en la ecuación (2.22) bajo la forma representada por la matriz (2.23).

$$\Phi = [\Phi_1 \quad \Phi_2 \quad \Phi_3]^T \quad (2.21)$$

$$i = [i_1 \quad i_2 \quad i_3]^T \quad (2.22)$$

$$\Phi = Li = \begin{bmatrix} L_1 & M_{12} & M_{13} \\ M_{21} & L_2 & M_{23} \\ M_{31} & M_{32} & L_3 \end{bmatrix} i \quad (2.23)$$

La energía almacenada en este sistema se expresa mediante la ecuación (2.24),



$$E(\phi) = \frac{1}{2} \phi^T L^{-1} \phi \quad (2.24)$$

La representación gráfica se muestra en la Figura 2.15

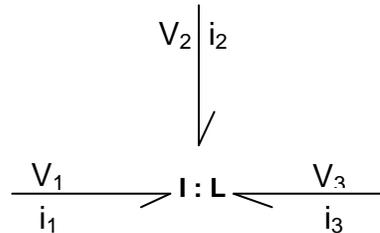


Figura 2.15. Representación Multipuerto.

Las variables de esfuerzo y flujo se agrupan en los vectores (2.25) y (2.26) respectivamente, y la potencia es expresada por la ecuación (2.27),

$$v = [v_1 \ v_2 \ v_3]^T \quad (2.25)$$

$$i = [i_1 \ i_2 \ i_3]^T \quad (2.26)$$

$$P = v^t \cdot i \quad (2.27)$$

Los elementos multipuertos no necesariamente son inductancias, también pueden ser elementos tipo R o resistivos y elementos tipo C o capacitivos como se muestra a continuación:

- **Capacitancia Multipuerto:** Este elemento se caracteriza por almacenamiento de energía múltiple, con dos flujos como entrada su representación en Bond Graph es mostrada en la Figura 2.16.

$$\left| \begin{array}{c} e_1(t) \\ f_1(t) \end{array} \right| \mathbf{C} \left| \begin{array}{c} e_2(t) \\ f_2(t) \end{array} \right|$$

Figura 2.16. Representación Multipuerto C.

$$\begin{bmatrix} e_1(t) \\ e_2(t) \end{bmatrix} = f(t) \begin{bmatrix} q_1(t) \\ q_2(t) \end{bmatrix} \quad (2.28)$$

- **Inductancia Multipuerto:** Este elemento al igual que la capacitancia multipuerto, se caracteriza por almacenamiento de energía múltiple, con dos flujos como entrada su representación en Bond Graph es mostrada en la Figura 2.17.

$$\left| \begin{array}{c} e_1(t) \\ e_2(t) \end{array} \right| \mathbf{I} \left| \begin{array}{c} f_1(t) \\ f_2(t) \end{array} \right|$$

Figura 2.17. Representación Multipuerto I.

$$\begin{bmatrix} f_1(t) \\ f_2(t) \end{bmatrix} = f(t) \begin{bmatrix} p_1(t) \\ p_2(t) \end{bmatrix} \quad (2.29)$$

- **Multipuerto Combinado C e I:** Almacenamiento de energía múltiple con un flujo y un esfuerzo como entrada, su representación en Bond Graph es mostrada en la Figura 2.18.

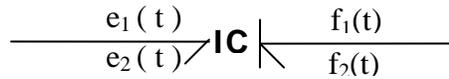


Figura 2.18. Representación Multipuerto combinado C e I.

$$\begin{bmatrix} e_1(t) \\ f_2(t) \end{bmatrix} = f(t) \begin{bmatrix} p_1(t) \\ q_2(t) \end{bmatrix} \quad (2.30)$$

2.2.4 ASIGNACIÓN CAUSAL

Uno de los objetivos que se persigue al hacer el análisis de un grafo de enlace es la asignación de la causalidad. Esta es utilizada como una herramienta de análisis previo para la obtención de las distintas ecuaciones tanto algebraicas como diferenciales que posee el modelo.

Un segundo objetivo es la obtención de los flujos y esfuerzos existentes sobre cada uno de los distintos grafos, de tal forma que se pueda dar lugar a un conjunto ordenado de ecuaciones algebraicas y diferenciales que puedan ser resueltas por un método de integración numérica [4].

2.2.4.1 CAUSALIDAD

El concepto de causalidad es muy importante ya que mediante este podemos establecer las relaciones causa – efecto entre los diferentes elementos de un sistema, ya que en cada enlace de un multipuerto existe tanto una variable de esfuerzo como una variable de flujo y ambas tienen sentido contrario, solo es posible que una de ellas sea la entrada mientras que la otra sea la salida, en otras palabras, una variable impone mientras que la otra es la respuesta [10].



La manera de representar la causalidad es mediante un trazo perpendicular al enlace, situado en uno de sus extremos, cuando la causa de un bond es el flujo se coloca un trazo perpendicular al bond en la parte posterior de este como se muestra en la Figura 2.19.

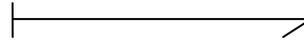


Figura 2.19. Grafo con Causalidad flujo.

Mientras que si la causa es el esfuerzo, el trazo se coloca en la parte anterior del bond como se muestra en la Figura 2.20.

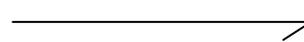


Figura 2.20. Grafo con Causalidad esfuerzo.

De esta manera el trazo causal indica un sentido de entrada-salida para $e(t)$ y $f(t)$ en cada puerto de los dos componentes juntos. El esfuerzo entra en un bond en el mismo sentido del trazo causal.

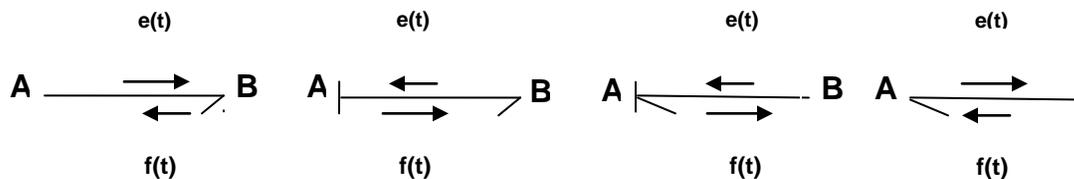


Figura 2.21. Causalidad para un enlace.



Cuando los enlaces poseen solamente una de las dos variables (esfuerzo o flujo), en cualquiera de las dos direcciones, bien sea debido a que la segunda variable asociada al enlace siempre tiene un valor cero o porque su valor es despreciable con respecto a la otra, en este caso se dice que el enlace está activado [10].

Los sub-sistemas unidos por un enlace que se encuentra activo poseen una relación particular donde solo una señal es transmitida de uno de ellos hacia el otro, sin producir una respuesta del segundo hacia el primero [10].

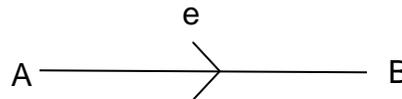


Figura 2.22. Enlace activo.

La Figura 2.22 muestra como la activación del enlace es representada mediante una flecha completa dibujada sobre el enlace activado, en donde el esfuerzo e está determinado por el sub-sistema A y es entrada al sub-sistema B y que el flujo f tiene un efecto despreciable sobre el sub-sistema A.

2.2.4.2 OBJETIVOS DEL ANÁLISIS CAUSAL

Los principales objetivos del análisis causal se clasifican en:

- Detección de las relaciones de dependencia entre las variables del modelo.
- Formulación de las ecuaciones.
- Asignación de flujo y esfuerzo a lo largo de una ruta causal [10].



2.2.4.3 CAUSALIDAD EN LOS ELEMENTOS BÁSICOS DEL BOND GRAPH

- **Causalidad en las Fuentes de Energía:**

La causalidad en las fuentes depende de la naturaleza de la misma ya que estas imponen un esfuerzo o un flujo pero no reciben respuestas [10].

Cuando la fuente es de flujo, la causa es el flujo y su representación gráfica es como se muestra en la Figura 2.23.

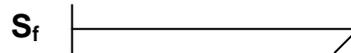


Figura 2.23. Causalidad en la fuente de flujo.

Cuando la fuente es de esfuerzo, la causa es el esfuerzo y su representación gráfica es como se muestra en la Figura 2.24.

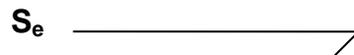


Figura 2.24 Causalidad en la fuente de esfuerzo.

- **Causalidad en los Elementos Resistivos:**

El elemento R tiene la característica de no imponer causalidad, lo que significa que esta se puede establecer arbitrariamente según sea la acción que la está produciendo. La representación en Bond Graph es la mostrada en la Figura 2.25.



Figura 2.25. Relación causal en elementos resistivos (a) Causa es el Flujo (b) Causa es el Esfuerzo.

▪ Causalidad en los Elementos Capacitivos e Inductivos:

La causalidad puede ser de tipo integral o derivativa, la derivativa se presenta cuando un sistema contiene elementos de almacenamiento de energía que no son dinámicamente independientes, en ésta, el número de elementos almacenadores de energía es igual al número de ecuaciones diferenciales linealmente dependientes. Mientras que en causalidad integral el número de elementos que almacenan energía es el número de ecuaciones diferenciales linealmente independientes.

La causalidad integral prevalece sobre la derivativa ya que las relaciones constitutivas de los elementos C e I son integrales y además de esto en Bond Graph se busca la independencia lineal del modelo, es decir, evitar que en el modelo matemático se tengan ecuaciones diferenciales linealmente dependientes, y con esto facilitar la solución numérica del modelo matemático [4].

La representación en Bond Graph para el elemento C es mostrada en la Figura 2.26 cuando la causalidad es producida por el esfuerzo, y en la Figura 2.27 cuando la causalidad es producida por el flujo [4].

$$esfuerzo = C \cdot \int flujo \cdot dt \quad (2.31)$$

Figura 2.26. Causalidad Esfuerzo en elemento C.



$$\text{-----} \nearrow \mathbf{C} \quad \text{flujo} = \frac{1}{c} \cdot d[\text{esfuerzo}]/dt \quad (2.32)$$

Figura 2.27. Causalidad flujo en elemento C.

La representación en Bond Graph para el elemento I cuando la causalidad es producida por el flujo es mostrada en la Figura 2.28 y cuando la causalidad es producida por el esfuerzo en la Figura 2.29 [4].

$$\text{-----} \nearrow \mathbf{I} \quad \text{flujo} = \frac{1}{m} \cdot \int \text{esfuerzo} dt \quad (2.33)$$

Figura 2.28. Causalidad flujo en elemento I.

$$\text{-----} \nearrow \mathbf{I} \quad \text{esfuerzo} = m \cdot d[\text{flujo}]/dt \quad (2.34)$$

Figura 2.29. Causalidad esfuerzo en elemento I.

▪ **Causalidad en Transformadores:**

Estos son de causalidad restringida, ya que el transformador no modifica la causalidad y si en el grafo de entrada la causalidad es flujo, en el de salida también lo es y viceversa para este elemento al igual que los anteriores se tendrán dos posibles casos. [10]



La Figura 2.30 muestra como la causalidad es igual tanto en los grafos entrantes como en los salientes.

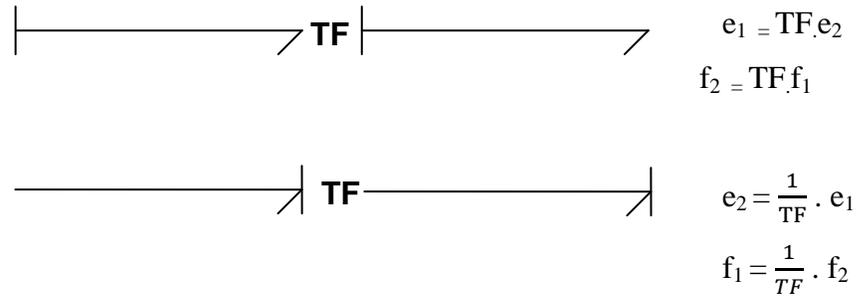


Figura 2.30. Causalidad en los transformadores.

▪ **Causalidad en Giradores:**

El elemento girador relaciona esfuerzos a la entrada con los flujos a la salida y viceversa. Como resultado, el girador cambia la causa en el grafo de salida respecto al de entrada [10].

Causalidad contraria en los grafos entrantes y salientes.

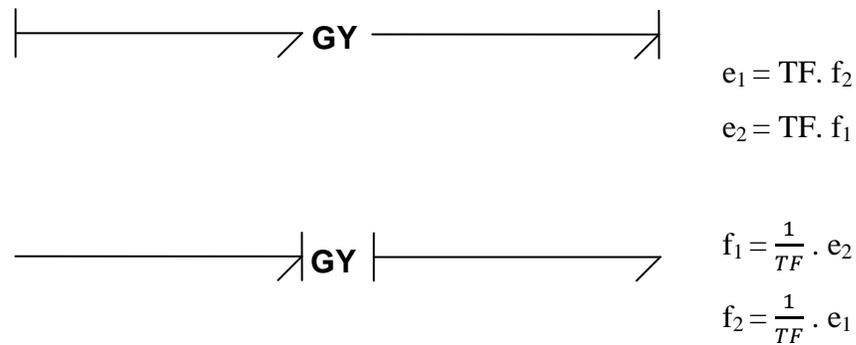


Figura 2.31. Causalidad en los giradores.

- **Causalidad en Unión-0 o Vínculo-0:**

Para la asignación de la causalidad en la unión cero, consideramos que todos los esfuerzos son iguales de manera que solo un enlace puede imponer esfuerzo y los demás flujo [10].

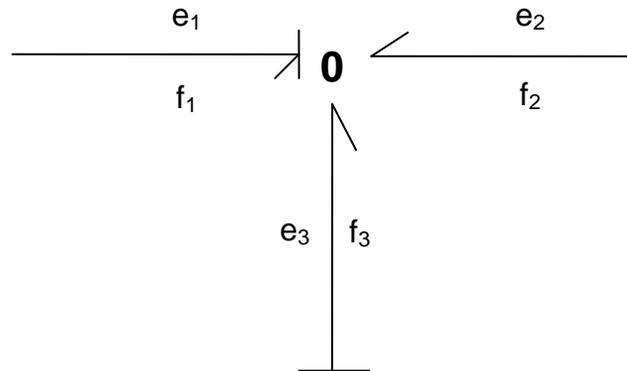


Figura 2.32. Causalidad en Unión-0 o Vínculo-0.

Tomando en cuenta el sentido del flujo de potencia, las relaciones constitutivas de este elemento se pueden escribir como:

$$e_1 = e_2 = e_3 \quad (2.35)$$

$$f_1 = - (f_2 + f_3) \quad (2.36)$$

- **Causalidad en Unión-1 o Vínculo-1:**

Para este tipo de unión los flujos todos son iguales por lo que, solo un enlace de la unión tipo 1 puede imponer flujo y los demás esfuerzo [10].

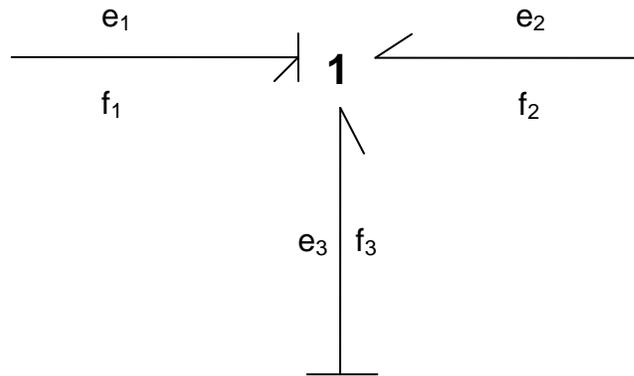


Figura 2.33. Causalidad en Unión-1 o Vínculo-1.

Utilizando la convención de signos dada por el sentido del flujo de potencia, las relaciones constitutivas del elemento 1 se pueden escribir como:

$$f_1 = f_2 = f_3 \quad (2.37)$$

$$e_1 = - (e_2 + e_3) \quad (2.38)$$

2.2.5 OTROS ELEMENTOS DE BOND GRAPH

Las relaciones constitutivas de los elementos descritos hasta ahora dependen de puertos y variables internas [23].

Cuando se desean modelar válvulas con restricciones hidráulicas, resistencias variables, condensadores, fuentes y otros componentes controlados en electrónica se requiere de otros elementos modulados los cuales serán descritos a continuación [23]:

- **Fuentes Moduladas:**

El modelado en Bond Graph permite el uso de las llamadas fuentes moduladas. En estas fuentes el esfuerzo y el flujo resultante son iguales a un valor fluctuante proporcionando una señal de entrada.



La dirección del esfuerzo y de flujo para estas fuentes están restringidas. En otras palabras: el MSE-elemento (esfuerzo-out) y MSF-elemento (esfuerzo-in) tienen una causalidad fija.

Existen dos tipos de fuentes moduladas a saber:

a) Fuente Modulada de Esfuerzo (MSe):

Este modelo representa una fuente ideal de esfuerzo modulada. Donde el esfuerzo puede ser ajustado a un valor (fluctuante) dado por una señal de entrada y el flujo es indiferente y conserva las características de causalidad de la fuente de esfuerzo. La representación en Bond Graph es mostrada en la Figura 2.34.

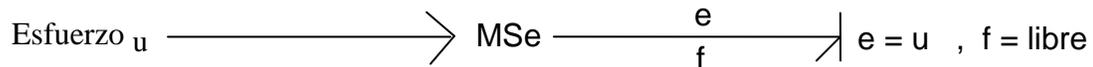


Figura 2.34. Fuente modulada de esfuerzo.

b) Fuente Modulada de Flujo (MSf):

Este modelo representa una fuente ideal de flujo modulada. El flujo se puede ajustar a un valor fluctuante dado por una señal de entrada y el esfuerzo es indiferente y conserva las características de causalidad de la fuente de flujo. La representación en Bond Graph es mostrada en la Figura 2.35.



Figura 2.35. Fuente modulada de flujo.



Transformadores y Giradores Modulados:

Estos son elementos no lineales y pueden estar en función de algún parámetro generado por una señal externa. En el transformador modulado las direcciones de esfuerzo y flujo están parcialmente fijadas y al igual que en la union-0 y la union-1 la causalidad es restringida, y la relación de transformación al igual que la relación de giro en el girador viene dada por una señal de entrada la cual le da un valor fluctuante.

Para el transformador una causalidad con esfuerzo entrante en el enlace de entrada produce una causalidad con esfuerzo saliente en el enlace de salida y viceversa, como es mostrado en las figuras respectivamente en las Figuras 2.36 y 2.37.

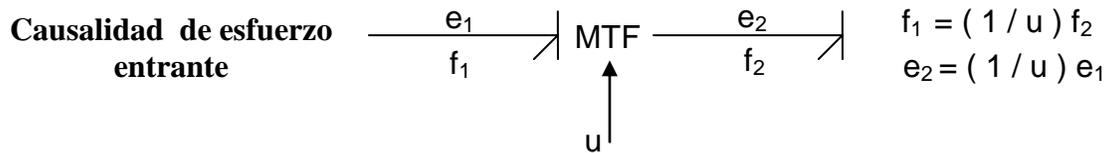


Figura 2.36. Transformador modulado con causalidad de esfuerzo entrante.

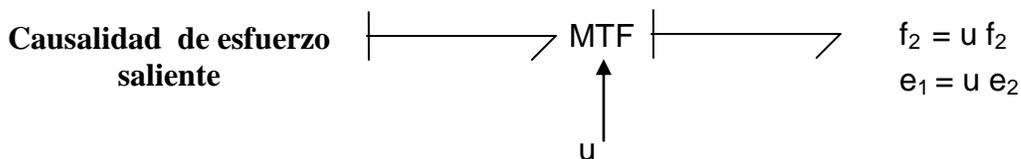


Figura 2.37. Transformador modulado con causalidad de esfuerzo saliente.

Mientras que para un Girador una causalidad con esfuerzo entrante en el enlace de entrada produce una causalidad con esfuerzo entrante en el grafo de salida y viceversa, la representación en Bond Graph es la mostrada en las Figuras 2.38 y 2.39.

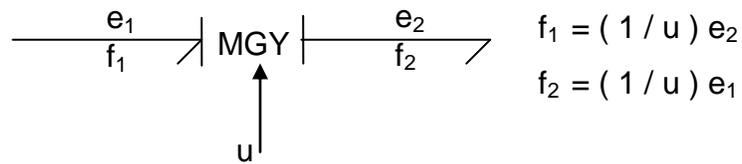


Figura 2.38. Girador modulado con causalidad de esfuerzo entrante.

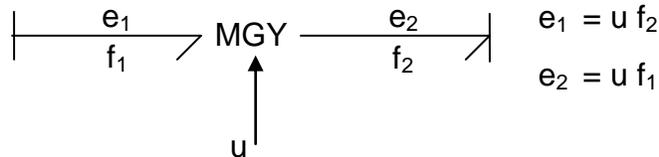


Figura 2.39. Girador modulado con causalidad de esfuerzo saliente.

- **Sensor de Esfuerzo:**

Este elemento se puede insertar en cualquier enlace para producir el esfuerzo de ese enlace como una señal de salida. Su representación en bond graph es la mostrada en la Figura 2.40.

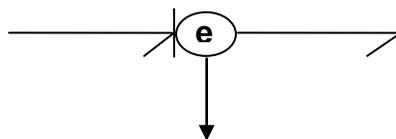


Figura 2.40. Sensor de esfuerzo.



- **Sensor de Flujo:**

Este elemento se puede insertar en cualquier enlace para producir el flujo de ese enlace como señal de salida.

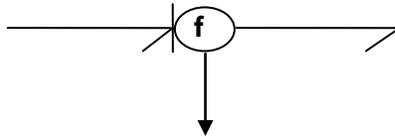


Figura 2.41. Sensor de flujo.

- **Divisor de Potencia:**

Divisor de potencia es un elemento que se puede utilizar para dividir un bono multi-en bonos individuales o viceversa.

- **Sensor de Energía:**

Este modelo se puede insertar en cualquier enlace para producir la energía que se realiza a través de ese enlace como señal de salida.

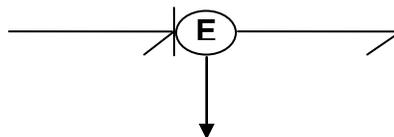


Figura 2.42. Sensor de energía.



2.2.6 SISTEMAS LINEALES

Considerando que los sistemas que se simulan en este Trabajo de Grado son todos de naturaleza lineales, a continuación se presenta una breve descripción de los sistemas lineales y el principio de superposición.

Un sistema puede llamarse lineal si cumple con el principio de superposición. De esta forma a continuación se explica de qué trata este principio.

Se considera un sistema físico el cual se muestra en la Figura 2.43 [22]. Si al aplicar una señal de entrada $x_1(t)$ al sistema de la Figura 1, se obtiene una salida $y_1(t)$, para una entrada diferente $x_2(t)$ se produce una salida $y_2(t)$, entonces el principio de superposición establece que para una entrada al sistema definida por $x(t) = x_1(t) + x_2(t)$, la salida del sistema debe ser: $y(t) = y_1(t) + y_2(t)$, esto se puede observar las Figuras 2.44 , 2.45 y 2.46.



Figura 2.43. Sistema físico.



Figura 2.44. Entrada $x_1(t)$ y la salida $y_1(t)$.



Figura 2.45. Entrada $x_2(t)$ y la salida $y_2(t)$.



Figura 2.46. Ley que rige el principio de Superposición.

Si no se cumple esta proposición, entonces el sistema es no-lineal.

2.3 IMPORTANCIA DE LA MODELACIÓN Y SIMULACIÓN DE LOS SISTEMAS EN INGENIERÍA Y LA APLICACIÓN DEL SOFTWARE 20-SIM Y BOND GRAPH:

Las técnicas de Bond Graph están enmarcadas dentro del área de modelación y simulación de sistemas lineales o no lineales. La simulación de sistemas se ha convertido en los últimos tiempos en una herramienta muy importante en la industria de procesos y el desarrollo de tecnología.

Actualmente la modelación y simulación de los sistemas son técnicas indispensables, más aún cuando se trata de sistemas multidisciplinarios con cierto grado de complejidad.

Un modelo simplifica un sistema de interés a través de la abstracción de un subconjunto de sus atributos observables, ya que este centra su atención en las características más relevantes y excluye otras que considera no son de relevancia directa al problema. El nivel de detalles incluidos depende del problema que se desea resolver y del solucionador, por tanto la solución del sistema es a partir de un modelo idealizado y



después de obtener una solución, los resultados se interpretan con respecto al contexto del mundo real del sistema original.

El enfoque de modelación utilizado en ingeniería es matemático, es decir los procesos físicos del mundo real son descritos por relaciones matemáticas que se resuelven mediante métodos numéricos o analíticos. Los sistemas de ingeniería son complejos, por tal motivo no es una tarea fácil crear un modelo válido y resolverlo de manera rápida y eficiente. Gracias a los avances de las tecnologías informáticas se han solventado problemas que antes eran intratables haciendo posible obtener resultados concretos válidos y exitosos en tiempos record [27].

Una excelente combinación resulta cuando el talento del diseñador se complementa con un buen software, el cual hace posible establecer una valoración final antes de que los sistemas sean construidos permitiendo predecir, comparar y optimizar el funcionamiento sin necesidad de llevarlos a cabo arriesgando altos costos, es decir, obviando experimentos costosos y dando soporte a las etapas de un proyecto desde el diseño conceptual pasando por el montaje hasta llegar al funcionamiento del mismo, lo cual a nivel de empresas e industrias resulta una herramienta hoy día indispensable [11].

El programa (o ambiente de programación) utilizado para realizar las simulaciones fue el 20-Sim el cual sirve para simular comportamientos de sistemas dinámicos tales como sistemas eléctricos, sistemas mecánicos, sistemas hidráulicos o cualquier combinación de estos sistemas. Este ambiente de programación proporciona un modelado, simulación y entorno de análisis para sistemas de ingeniería basado en la jerarquía de modelo. Este se divide en dos partes la primera es donde se introduce el modelo y se compila y con la segunda se realizan los experimentos sobre el modelo [11].

Las razones de por qué utilizar 20-sim y no algún otro programa son muchas, este fue el primer software basado en la técnica de Bond Graph y cuenta con una librería múltiple con un gran sistema de submodelos de dominio orientado, permite crear submodelos y combinarlos para formar modelos complejos, utiliza algoritmos de integración avanzada, permite una interacción cerrada con MATLAB Y SIMULINK.



FASES METODOLÓGICAS

3.1 METODOLOGÍA A UTILIZAR

Se plantea una metodología o procedimiento ordenado que se sigue para establecer lo significativo de los hechos y fenómenos hacia los cuáles está encaminado el interés de la investigación. Se adoptará el método análisis-síntesis que consiste en separar las partes de un todo para estudiarlas en forma individual (análisis), y la reunión racional de elementos dispersos para estudiarlos en su totalidad (síntesis) [16].

El Manual de Trabajos de Grado de Especialización, Maestría y Tesis Doctorales de la Universidad Pedagógica Experimental Libertador define: “La investigación, elaboración y desarrollo de una propuesta de un modelo operativo viable para solucionar requerimientos o necesidades de organizaciones o grupos sociales; puede referirse a la formulación de políticas, programas, tecnologías, métodos o procesos. El proyecto debe tener el apoyo en una investigación de tipo documental o de campo, o un diseño que cumpla con ambas modalidades” [29].

Por lo tanto, este capítulo plantea un proyecto factible con un diseño documental, definido por la naturaleza y características del problema planteado para desarrollar el trabajo especial de grado: Diseño de Software Didáctico para la Enseñanza-Aprendizaje de Técnicas de Bond Graph con Aplicaciones en Modelación y Simulación de Sistemas Lineales en Ambiente de Programación 20-SIM.



3.2. TÉCNICAS DE RECOPIACIÓN DE DATOS

Para la realización de este trabajo, se ha utilizado la recolección de información teórica a través de Consultas bibliográficas: Se han revisado libros, trabajos de grado, tesis doctorales, publicaciones de revistas, publicaciones y papers por Internet, relacionados con la teoría de Bond Graph, tomando en cuenta fundamentalmente todo lo referente con modelación y control, también se ha documentado con todo lo referente al software de programación 20-SIM.

3.3. ANALIS DE DATOS Y RESULTADOS

A pesar de que son muchas las aplicaciones desarrolladas en distintos lenguajes de programación para la simulación de Bond Graph, solo unas pocas están distribuidas comercialmente, lo que nos limita a la verificación de resultado a un solo Software.

Por lo tanto para la verificación y validación de los resultados se utilizará el software de programación 20-SIM, ya que nos permite simular el comportamiento de sistemas dinámicos, tales como: sistemas eléctricos, mecánicos e hidráulicos o cualquier combinación de estos sistemas. Además puede trabajar con Bond Graph multipuerto, con diagrama de bloques, diagrama icónicos y con ecuaciones.

3.4. ORGANIZACIÓN DEL TRABAJO POR PARTES

En este trabajo especial de grado, se plantea como objetivo principal el diseño de un software didáctico para la enseñanza-aprendizaje de métodos y técnicas de BOND GRAPH con aplicaciones en modelación y simulación de sistemas lineales en ambiente de



programación 20-SIM, y para poder alcanzarlo se desarrolla en cinco objetivos específicos de la siguiente manera:

1. Describir la teoría de Bond Graph para el modelado de diferentes sistemas tanto eléctrico, hidráulico, mecánico, etc.

Con ayuda de los diferentes recursos bibliográficos (trabajos de grado, tesis doctorales, publicaciones de revistas, etc.) se recopila toda la información referente a la teoría de Bond Graph y la información necesaria para la utilización del software 20-SIM como herramienta de simulación de los grafos de enlace.

En el capítulo II se expone todo el marco conceptual referente al tema, comenzando por una breve descripción del origen del Bond Graph y explicando todos los conceptos, definiciones y elementos que constituyen los grafos de enlace.

2. Aplicar la técnica de Bond Graph para la resolución de problemas en modelación y simulación de sistemas lineales en diferentes dominios físicos. Para alcanzar este objetivo, se desarrollan paso a paso diferentes problemas con distintos dominios físicos (mecánico, eléctrico, hidráulico, etc.). Dichos problemas han de ser familiares al lector que haya estado en contacto con estos sistemas físicos, especialmente los estudiantes de Ingeniería. Sin embargo, se han escogidos los problemas, desde una baja dificultad, pasando por una dificultad media hasta llegar a una dificultad alta, de manera tal que el uso del Software no esté restringido para ningún usuario.

3. Desarrollar un Manual/Tutorial sobre el Software 20-SIM explicando diferentes ejemplos de Programación donde se realiza la Simulación de diversos sistemas basada en las Técnicas de Bond Graph.

Se presenta una descripción del software 20-SIM. En el mismo orden de ideas, se muestra el editor que sirve para introducir y editar los modelos en forma de Bond Graph, ya que el software cuenta con la opción de poder introducir los modelos utilizando Ecuaciones, Diagramas de bloque, Bond Graph, Diagramas Icónicos o



una combinación de estas representaciones. También se muestra el simulador para describir y analizar los modelos, de forma tal que se puedan explicar diferentes ejemplos de programación de la simulación de diversos sistemas basada en las Técnicas de Bond Graph. Tanto el editor como el simulador, son componentes que están presentes dentro del 20-SIM, y son fundamentales para poder elaborar las simulaciones.

4. Desarrollar el Software Didáctico Multimedia en Ambiente Dreamweaver, con un manual para el usuario explicando las diferentes secciones del trabajo.

Con el apoyo de toda la información teórica, se desarrolla un manual didáctico, que describe las diferentes secciones del trabajo de grado, con énfasis en la aplicación de la técnica de Bond Graph en la resolución de sistemas físicos lineales, de manera que sea confortable para la revisión y estudio del usuario.



MODELACIÓN EN BOND GRAPH

4.1 LA MODELACIÓN EN BOND GRAPH

Cualquier sistema físico, el cual puede ser eléctrico, mecánico, hidráulico etc., puede ser modelado mediante la interconexión de un conjunto de elementos simples, obteniéndose un modelo el cual es la representación formal de la teoría de Bond Graph. Al modelar en Bond Graph pasamos de un dominio físico a otro usando variables generalizadas, sin importar si el sistema viene de uno o más dominios de energía. El siguiente procedimiento explica los pasos a seguir para obtener un modelo en Bond Graph [18]. Cabe destacar que existe una variedad de procedimientos para obtener un modelo en Bond Graph, se diferencian según el criterio del autor para desarrollar la técnica de Bond Graph [6], a continuación se exponen algunos de ellos.

4.1.1 PROCEDIMIENTO GENERAL PARA LA CONSTRUCCIÓN DE MODELOS EN BOND GRAPH [6]

1. Identificar que dominios físicos existen en el sistema e identificar cada uno de los elementos que lo conforman R, C, I, Se, Sf, TF y GY.
2. Dibujar las uniones-0 o uniones-1 de acuerdo al sistema correspondiente.
3. Dibujar una unión-0 o unión-1 para cada componente en una trayectoria serie y conectar el componente apropiado de Bond Graph usando un enlace en esa unión. La dirección asumida del flujo de potencia es a partir de las fuentes hacia los elementos almacenadores y/o disipadores de energía, la dirección se indica por el sentido de la semiflecha.
4. Identificar el nodo o punto de referencia.
5. Remover la unión-0 o unión-1 que representa el nodo o punto de referencia, así como también todos los Enlaces conectados a dicha unión.
6. Simplificar el modelo obtenido utilizando las Reglas de Simplificación (mostradas al final de este procedimiento).
7. Asignar la causalidad a los elementos empezando por las fuentes, seguido de los elementos almacenadores (causalidad integral) y finalmente los elementos disipadores (causalidad arbitraria).



• Reglas de Simplificación de Modelos en Bond Graph

La primera regla corresponde a la simplificación de uniones (regla 1), esta se cumple cuando en una unión hay un enlace entrante y uno saliente, lo cual permite obviar la unión y tomar en cuenta solo un enlace tal como es mostrado en la Figura 4.1 [6].



Figura 4.1. Regla 1, Simplificación Uniones en Bond Graph.

La segunda regla de simplificación corresponde a la de un enlace entre dos uniones iguales (regla 2), en esta regla puede ser eliminado o excluido el enlace y las uniones pueden unirse a una sola unión como es mostrada en la Figura 4.2 y Figura 4.3 [6].

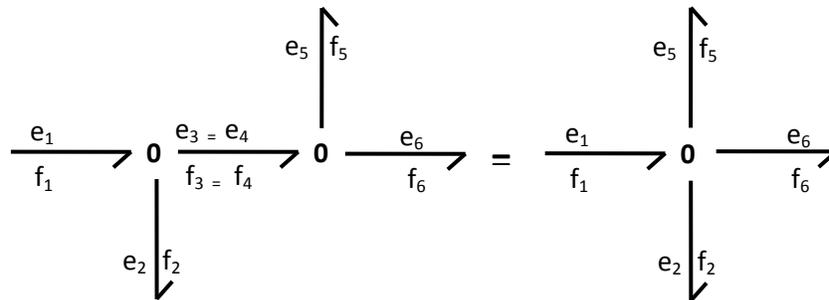


Figura 4.2. Regla 2, Simplificación Bond Graph referida a las Uniones-0.

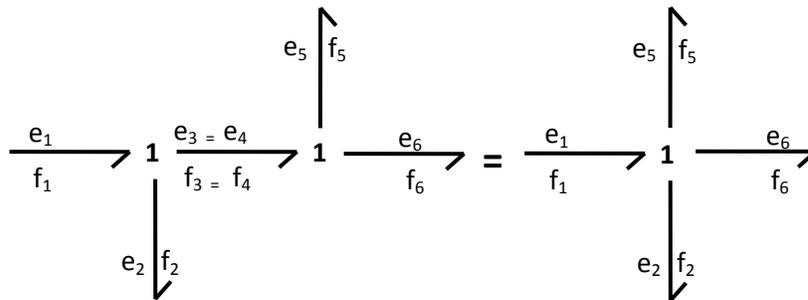


Figura 4.3. Regla 2, Simplificación Bond Graph referida a las Uniones-1.



La **tercera regla** corresponde a la de la diferencia de esfuerzo o flujo construidos idénticamente pero separadas entre sí, pueden transformarse en una sola diferencia de flujo o esfuerzo (regla 3), estas son mostradas en las Figuras 4.4 y 4.5 respectivamente, (ver apéndice C) [6].

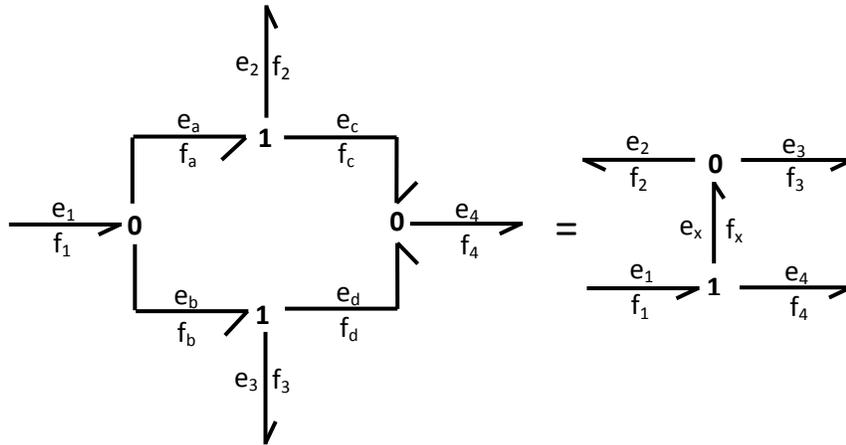


Figura 4.4. Regla 3, Simplificación Bond Graph referente a diferencia de esfuerzo o flujo 1.

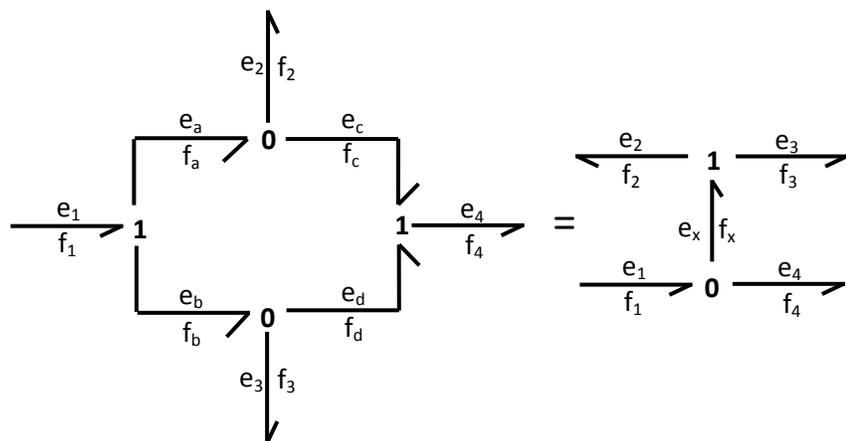


Figura 4.5. Regla 3, Simplificación Bond Graph referente a diferencia de esfuerzo o flujo 2.



4.1.2 PROCEDIMIENTO PARA LA OBTENCIÓN DE MODELOS EN BOND GRAPH PARA SISTEMAS ELÉCTRICOS

4.1.2.1 CIRCUITO RLC

Dado el circuito RLC mostrado en la Figura 4.6 obtener el modelo en Bond Graph aplicando el procedimiento descrito en los capítulos anteriores [8].

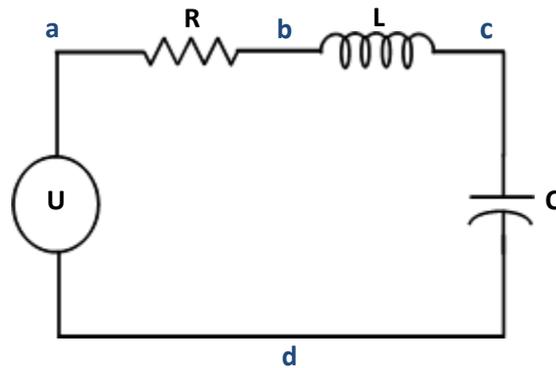


Figura 4.6. Circuito RLC.

1. Se identifica los nodos donde existen voltajes diferentes en el circuito estos corresponden a los puntos (a, b, c y d), estos nodos en Bond Graph se identifican como uniones-0 cada uno de ellos, tal como es mostrado en la Figura 4.7.

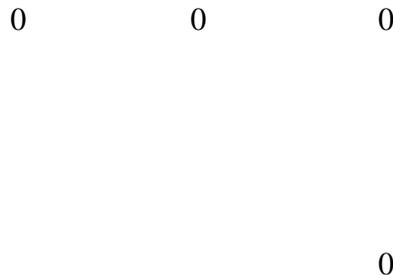


Figura 4.7. Uniones-0 Circuito RLC.



- 2. Se colocan uniones-1 las cuales unirán cada uno de los elementos involucrados en el modelo, es decir habrán tantas uniones-1 como elementos hay en el modelo esta representación se muestra gráficamente en la Figura 4.8.

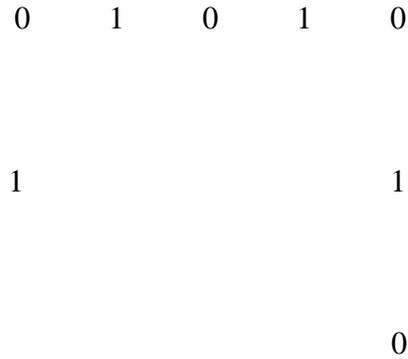


Figura 4.8. Inserción de Uniones -1 Circuito RLC.

- 3. Se colocan los elementos involucrados en el modelo en el lugar correspondiente (cada unión-1 representa un elemento) como es mostrado en la Figura 4.9.

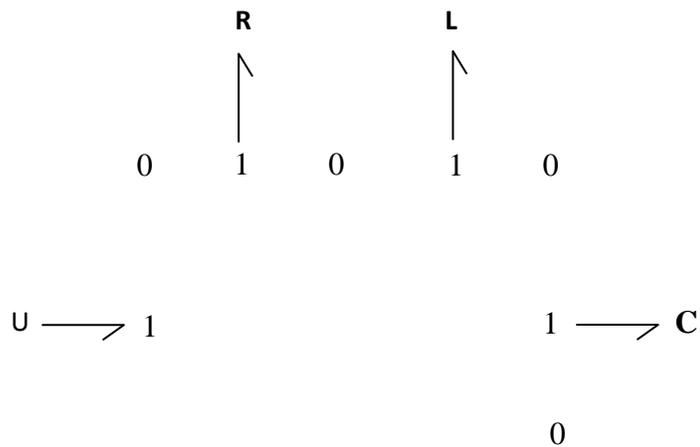


Figura 4.9. Inserción de elementos en Circuito RLC.



4. En el siguiente paso se agregan los enlaces faltantes para terminar de unir todos los componentes, considerando siempre que la flecha apuntara en sentido del flujo de la potencia, como es mostrado en la Figura 4.10.

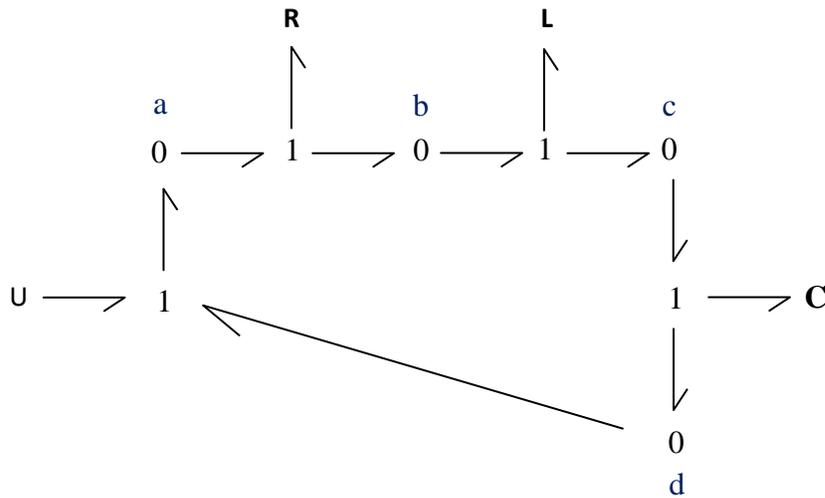


Figura 4.10. Inserción de enlaces faltantes para completar modelo RLC.

5. Se identifica el nodo de referencia (d), el cual se elimina junto con los enlaces que estén conectados al nodo de referencia, como es mostrado en la Figura 4.11.

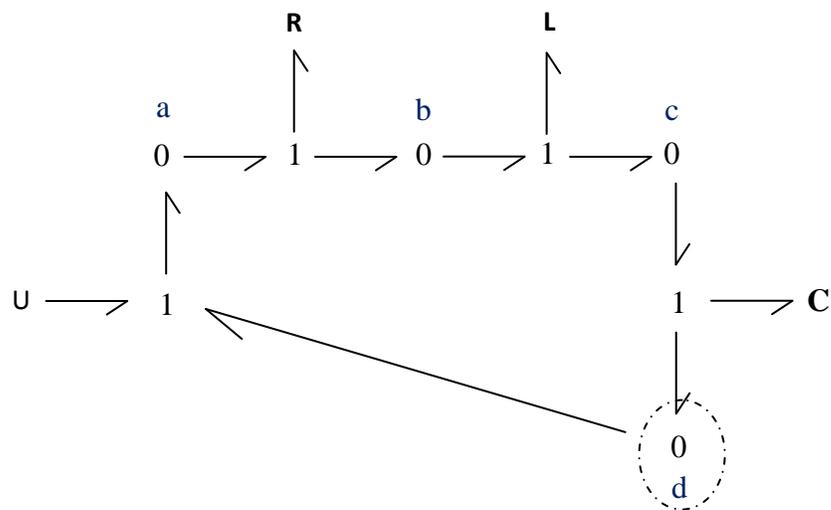


Figura 4.11. Identificación nodo de referencia en Circuito RLC.

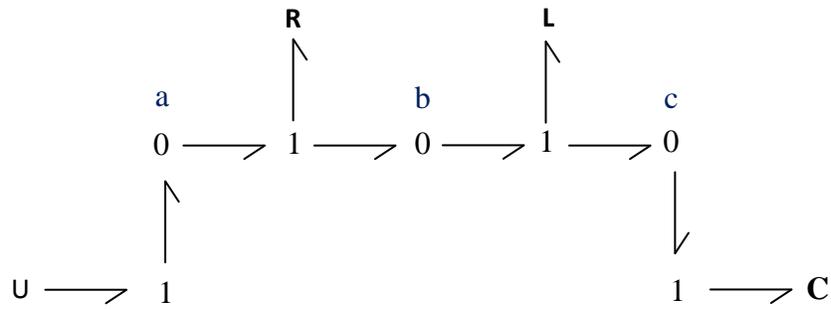


Figura 4.12. Simplificación de los enlaces conectados al nodo de referencia del Circuito RLC.

6. Se simplifica el modelo aplicando las reglas de simplificación de los Bond Graph.

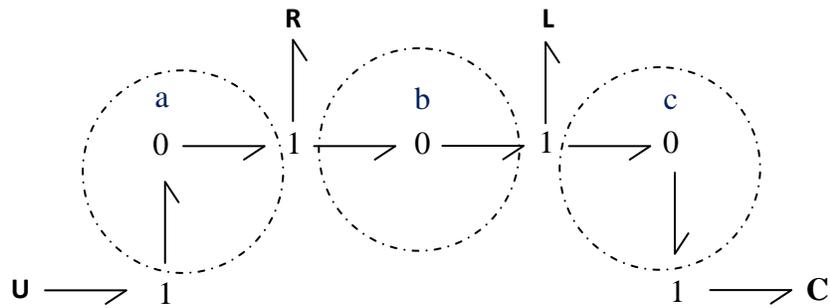


Figura 4.13. Identificación en modelo RLC de uniones y enlaces a eliminar aplicando la regla 1.

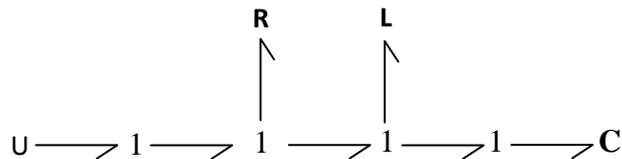


Figura 4.14. 1ra Simplificación de Uniones y enlaces en Circuito RLC aplicando regla 1.

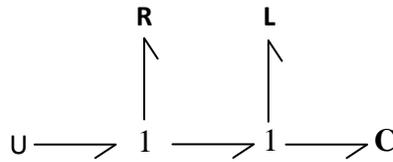


Figura 4.15. 2da Simplificación de Uniones y enlaces en circuito RLC aplicando regla 1.

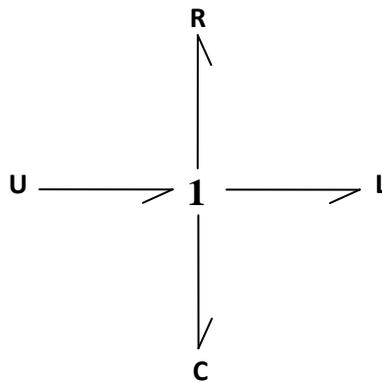


Figura 4.16. Obtención del modelo en Bond Graph del circuito RLC.

7. Se asigna la causalidad a los elementos empezando por las fuentes, seguido de los elementos almacenadores (causalidad integral) y finalmente los elementos disipadores (causalidad arbitraria). Finalmente el modelo quedara como el mostrado en la Figura 4.17.

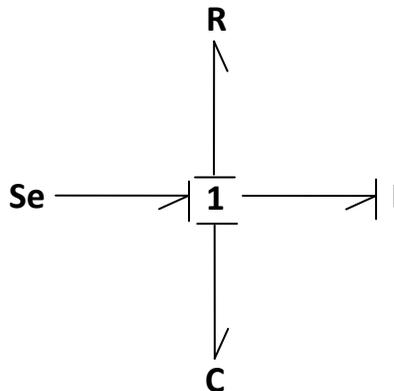


Figura 4.17. Asignación de causalidad al circuito RLC.



4.1.2.2 CIRCUITO ELECTRICO MIXTO

Dado el circuito eléctrico mostrado en la Figura 4.18 Obtener el modelo en Bond Graph [27].

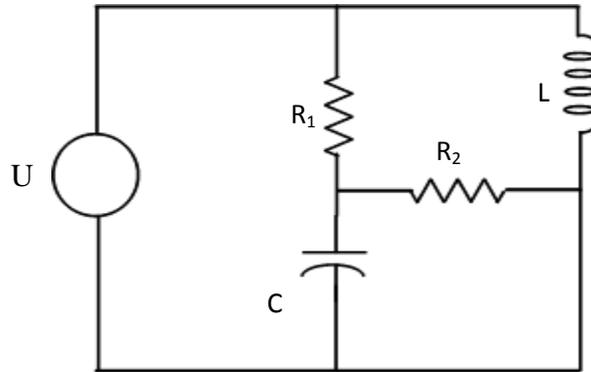


Figura 4.18. Circuito eléctrico mixto.

1. Identificamos los nodos donde existen voltajes diferentes en nuestro circuito estos corresponden a los puntos (a,b,c) en Bond Graph se identifican como uniones-0 cada uno de ellos.

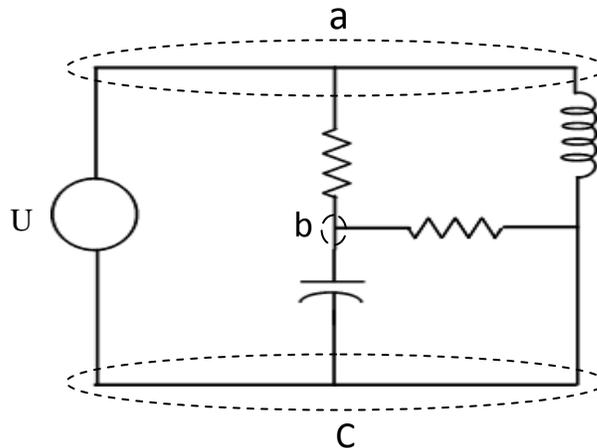


Figura 4.19. Identificación de nodos en circuito mixto.



0

0

0

Figura 4.20. Ubicación Uniones-0 para el Circuito RLC.

2. Se colocan uniones-1 las cuales unirán cada uno de los elementos involucrados en el modelo, es decir habrán tantas uniones-1 como elementos hay en el modelo.

1

0

1

1

0

1

1

0

Figura 4.21. Ubicación Uniones-1 y Uniones- 0 para el Circuito RLC.

3. Se colocan los elementos involucrados en el modelo en el lugar correspondiente (cada unión-1 representa un elemento).

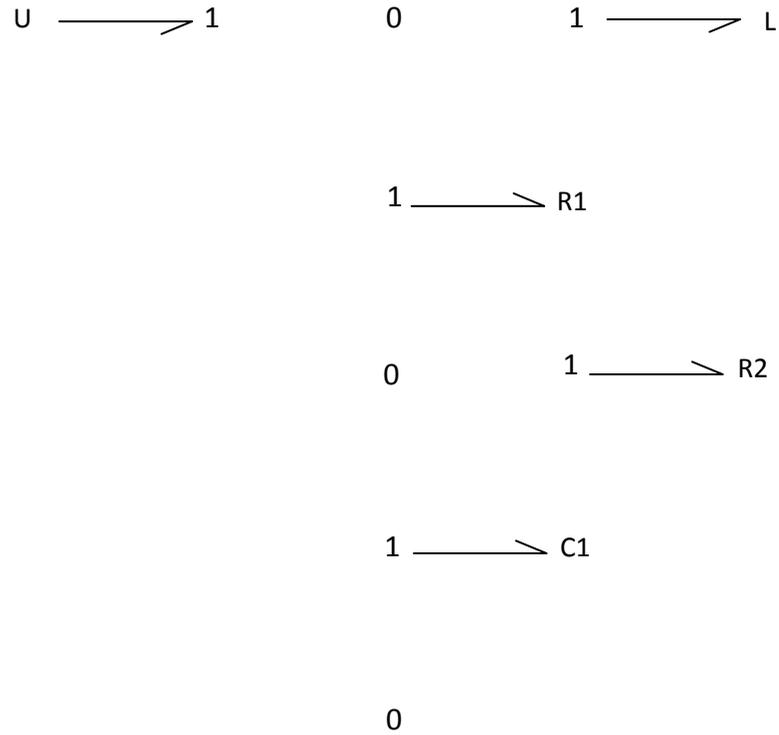


Figura 4.22. Inserción de enlaces correspondientes a cada elemento del circuito RLC.

- 4. En el siguiente paso se agregan los Bond faltantes para terminar de unir todos los componentes, considerando siempre que la flecha apuntara en sentido del flujo.

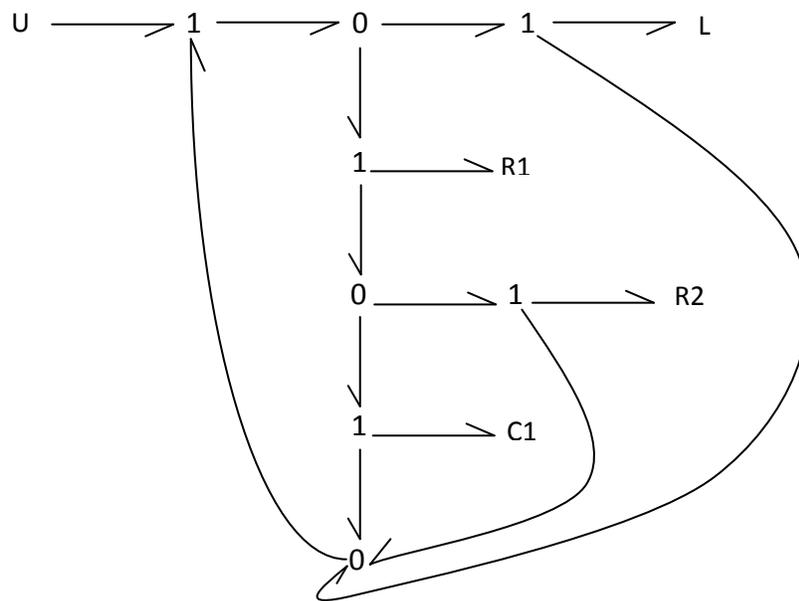


Figura 4.23. Inserción enlaces para unir elementos del circuito RLC.



- 5. Se identifica el nodo de referencia (d) el cual se eliminará junto con los Enlaces que estén conectados a dicho nodo.

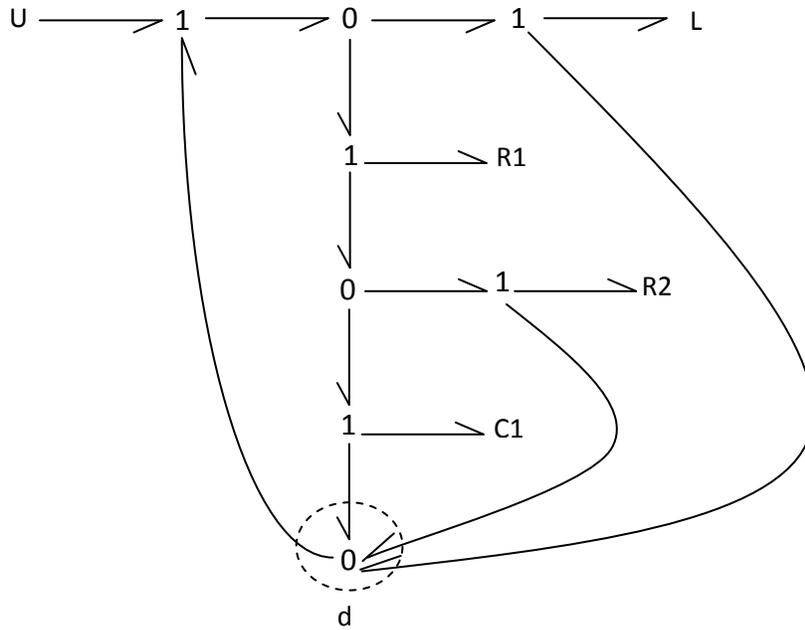


Figura 4.24. Ubicación de nodo de referencia para el circuito RLC.

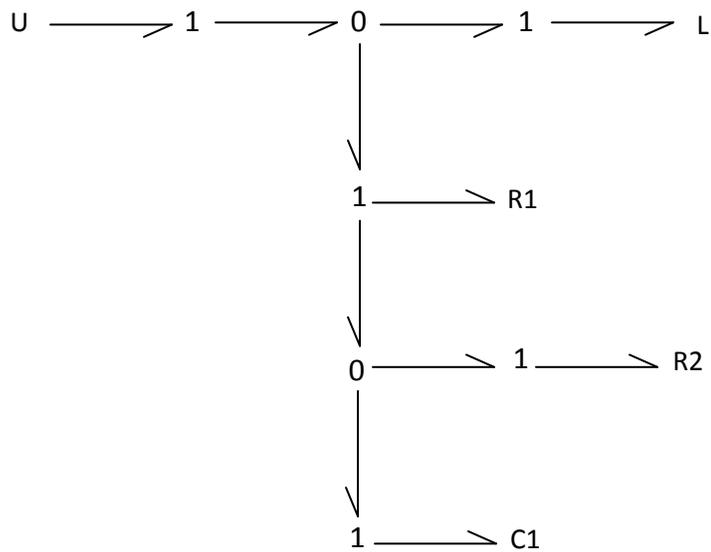


Figura 4.25. Simplificación de los enlaces conectados al nodo de referencia.



- 6. Se simplifica nuestro modelo aplicando las reglas de simplificación de los enlaces mostradas anteriormente, esta simplificación corresponde a la regla 1.

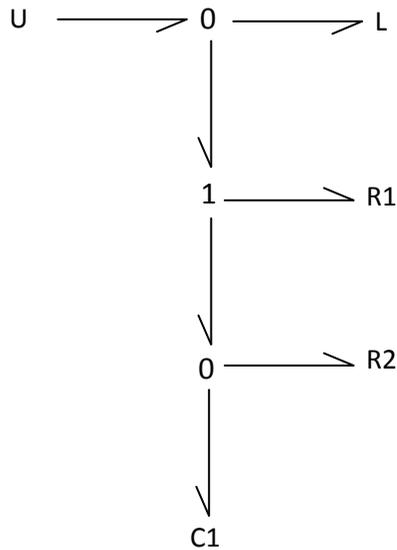


Figura 4.26 Simplificación del circuito RLC aplicando la regla 1.

Obtención del modelo en Bond Graph:

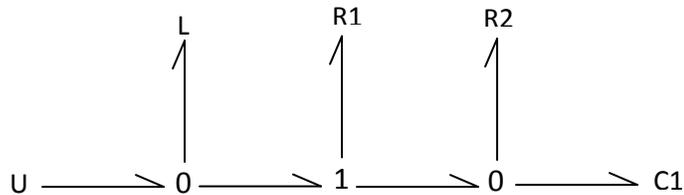


Figura 4.27 Modelo en Bond Graph del Circuito RLC.



4.1.2.3 CIRCUITO ELECTRICO CON TRANSFORMADOR

Dado el circuito mostrado en la Figura 4.28 obtener el modelo en Bond Graph:

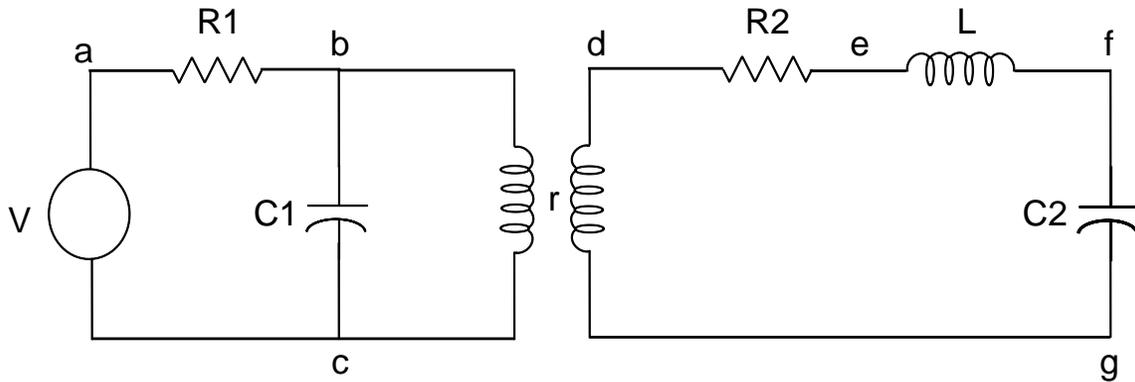


Figura 4.28 Circuito eléctrico con transformador.

1. Se asignan uniones-0 en los puntos donde se tiene voltaje distinto.
2. Se dibuja la unión-1 para cada componente en una trayectoria serie y se conecta el componente apropiado de Bond Graph usando un Enlace en esa unión. La dirección asumida del flujo de potencia es a partir de las fuentes hacia los elementos almacenadores y/o disipadores de energía, la dirección se indica por el sentido de la semiflecha.
3. Se dibujan los enlaces entre las uniones adyacentes indicando la dirección del flujo de potencia.
4. Se asigna como punto de referencia el nodo de voltaje cero, que es la tierra del circuito.
5. Simplificar el modelo obtenido utilizando las reglas mostradas anteriormente en las Figuras 4.1, 4.2, 4.3, 4.4 y 4.5.
6. Se asignar la causalidad a los elementos utilizados en el modelo [18].

El Circuito Eléctrico mostrado en la Figura 4.28 se toma como ejemplo para demostrar el procedimiento empleado en los sistemas eléctricos, primero se identifican los nodos donde existen voltajes diferentes (marcados como a, b, c, d, e y f), estos en Bond



Graph se representarán como uniones-0 cada uno de ellos, luego se procede a colocar las uniones-1 las cuales unirán cada uno de estos elementos utilizados en el modelo. En el siguiente paso se agregan los Enlace faltantes para terminar de unir todos los componentes y se identifican los nodos de referencia (c y g) los cuales se eliminarán junto con los Enlace que están conectados a ellos, esto se muestra en la Figura 4.29 [18].

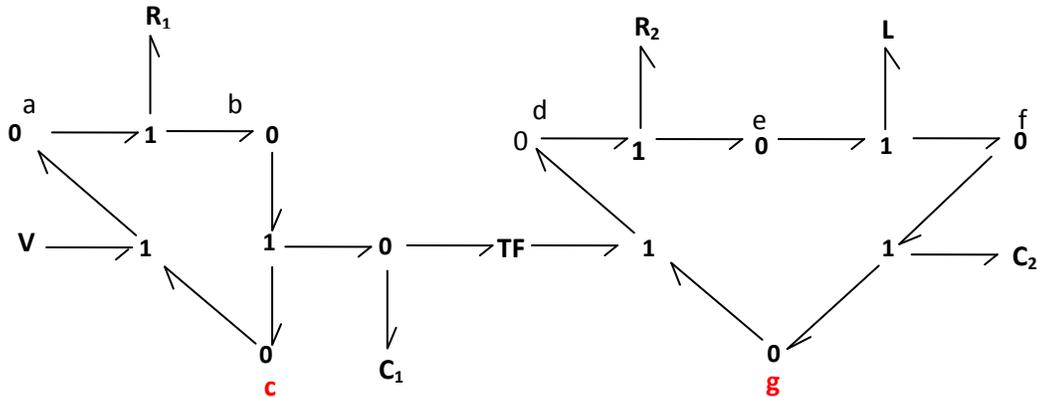


Figura 4.29. Procedimiento para obtener el Bond Graph del Sistema Eléctrico.

Simplificando de esta manera el modelo utilizando las reglas antes mencionadas para la reducción del Bond Graph, y asignándole la causalidad correspondiente a cada elemento podemos obtener finalmente nuestro modelo de Bond Graph representativo para el sistema eléctrico el cual mostramos en la Figura 4.30.

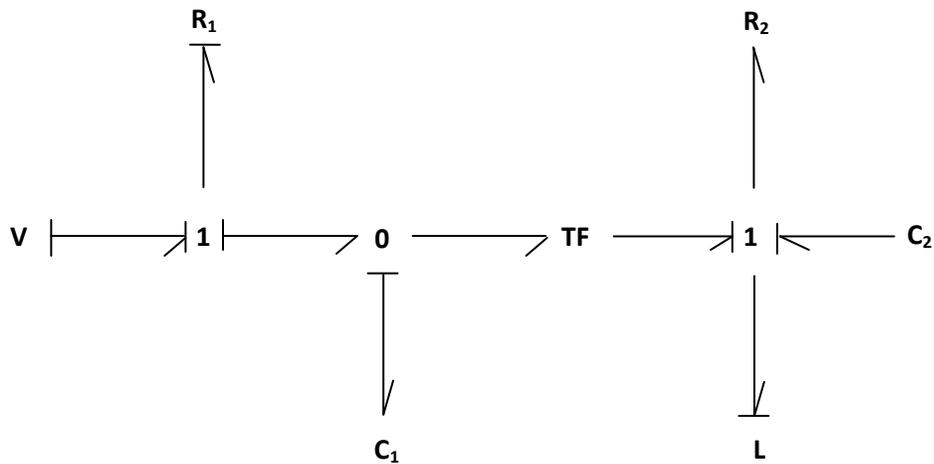


Figura 4.30. Bond Graph del Sistema Eléctrico.



4.1.3 PROCEDIMIENTO PARA LA OBTENCIÓN DE MODELOS EN BOND GRAPH PARA SISTEMAS MECÁNICOS

Para obtener el Bond Graph de un sistema mecánico se emplea el procedimiento general al igual que en el sistema eléctrico, pero considerando también las modificaciones correspondientes a un sistema mecánico.

4.1.3.1 SISTEMA MASA RESORTE AMORTIGUADOR

Dado el sistema masa resorte amortiguador mostrado en la Figura 4.31 obtener el modelo en Bond Graph.

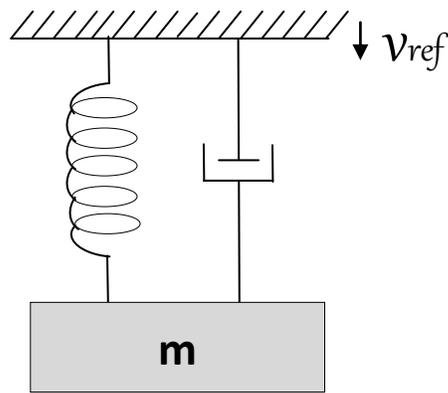


Figura 4.31. Sistema Masa-Resorte-Amortiguador.

1. Se identifica en el esquema del sistema mecánico, la dirección positiva de las distintas velocidades absolutas y relativas de los componentes y la de referencia [24].

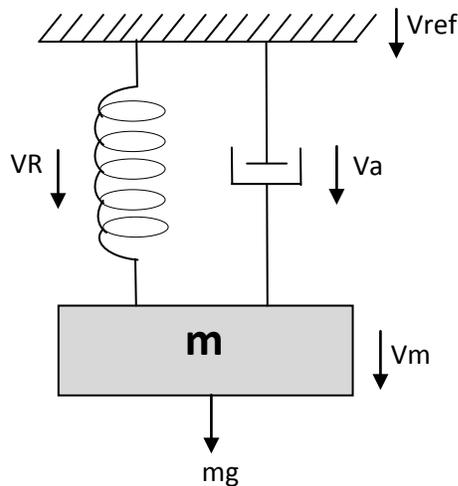


Figura 4.32. Ubicación de las velocidades en el sistema Masa-Resorte-Amortiguador.



- 2. Se asignan uniones-1 en los puntos donde se tienen velocidades absolutas, relativas distintas y en la velocidad de referencia [24].

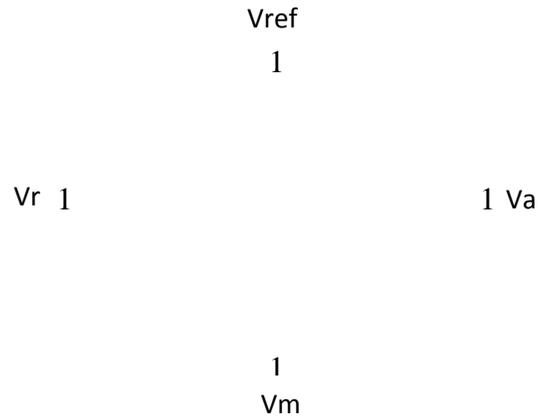


Figura 4.33. Ubicación Uniones-1 del modelo Masa – Resorte - Amortiguador.

- 3. Se insertan uniones-0 entre los puntos donde se encuentren las velocidades absolutas para establecer las velocidades relativas (las uniones-0 establecen las diferencias entre las velocidades absolutas que originan las velocidades relativas).

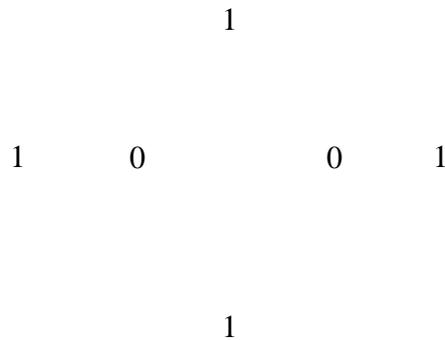


Figura 4.34. Inserción de las Uniones-0 del modelo Masa - Resorte - Amortiguador.

- 4. Se añaden las fuerzas que dan origen a las velocidades absolutas en sus uniones correspondientes.

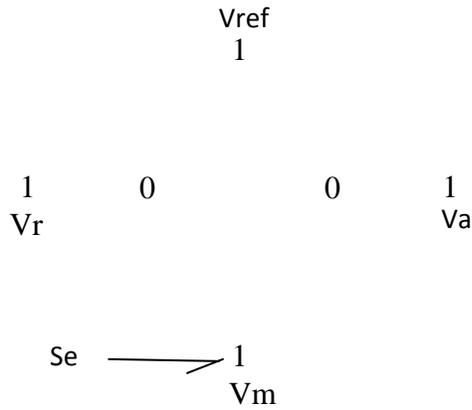


Figura 4.35. Inserción enlace que representa la fuerza que da origen a las velocidades.

5. Se añaden el elemento correspondiente a cada unión-1 excepto a la unión-1 correspondiente a la referencia.

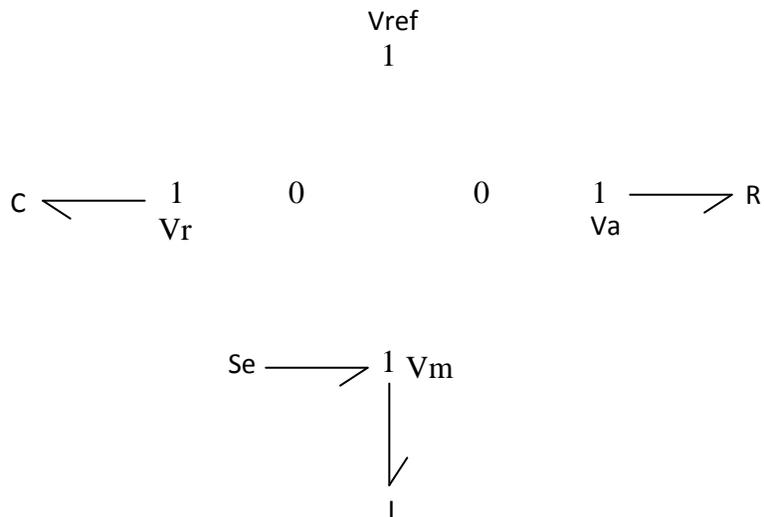


Figura 4.36. Inserción de cada elemento en el sistema Masa – Resorte - Amortiguador.



6. Se dibujan enlaces entre las uniones adyacentes indicando la dirección del flujo de potencia.

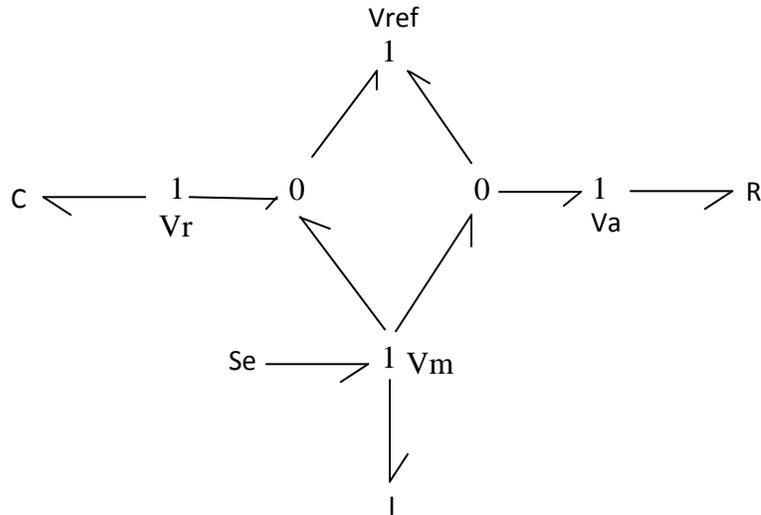


Figura 4.37. Inserción Uniones para completar el sistema Masa – Resorte - Amortiguador.

7. Se identifica la unión-1 de referencia y se elimina junto con los enlaces que estén conectados a dicha unión.

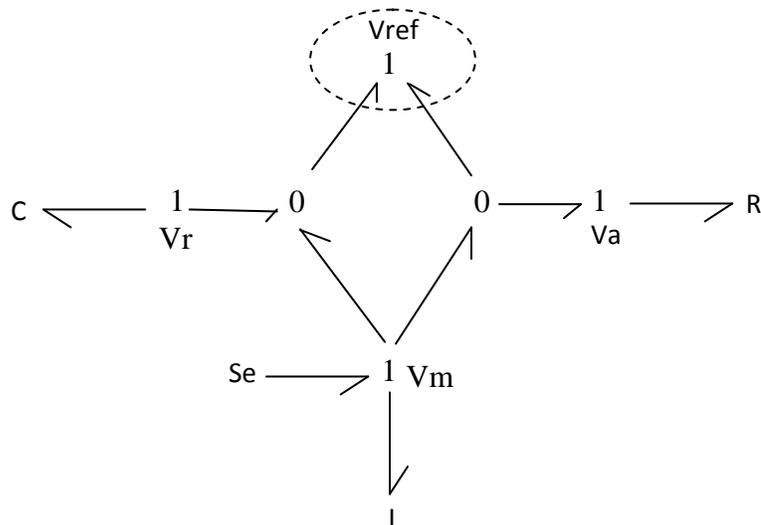


Figura 4.38. Identificación Unión-1 de referencia.

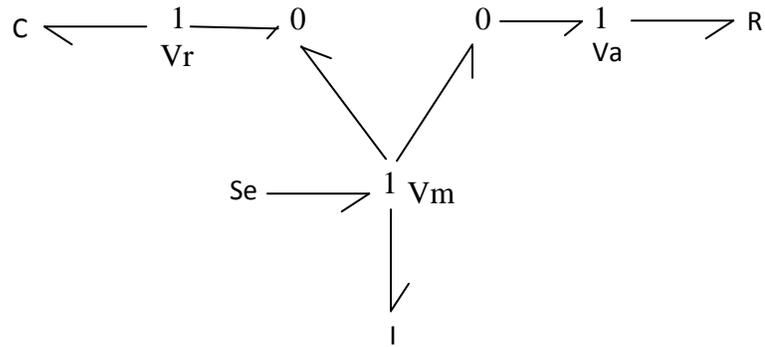


Figura 4.39. Simplificación enlaces conectados a Unión de referencia del Sistema M-R-A.

8. Se simplifica el modelo obtenido utilizando las reglas.

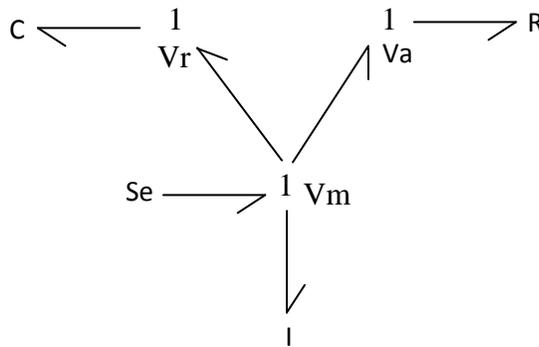


Figura 4.40. Simplificación Uniones-0 del sistema Masa – Resorte - Amortiguador.

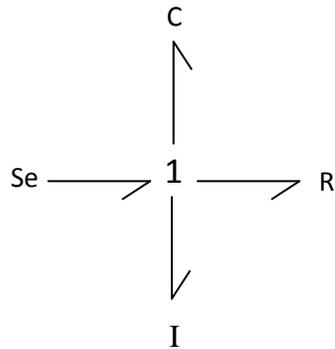


Figura 4.41. Simplificación Uniones-1 del sistema Masa – Resorte - Amortiguador.

4.1.3.2 SISTEMA MECANICO MIXTO

Se tiene el sistema mecánico mostrado en la Figura 4.42, obtener el modelo en Bond Graph, a partir del procedimiento que mostramos a continuación.

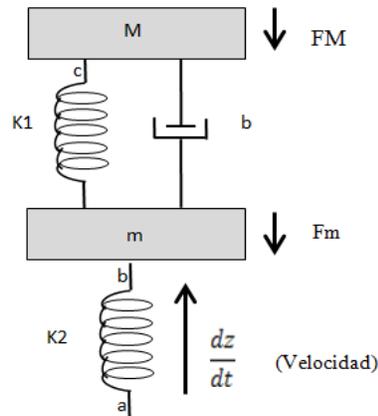


Figura 4.42. Sistema Mecánico mixto de traslación.

1. Se Identifica en el esquema del sistema mecánico, la dirección positiva de las distintas velocidades absolutas y relativas de los componentes y la de la velocidad de referencia [24].
2. Se asignan uniones-1 en los puntos donde se tienen velocidades absolutas y relativas distintas y en la velocidad de referencia [24].
3. Se insertan uniones-0 entre los puntos donde se encuentren las velocidades absolutas para establecer las velocidades relativas (las uniones-0 establecen las diferencias entre las velocidades absolutas que originan las velocidades relativas).
4. Se añade el elemento correspondiente a cada unión-1 excepto a la unión-1 correspondiente a la referencia.
5. Se añaden las fuerzas que dan origen a las velocidades absolutas en sus uniones correspondientes.
6. Se Dibujan enlaces entre las uniones adyacentes indicando la dirección del flujo de potencia.



7. Se identifica la unión-1 de referencia y eliminar junto con los enlaces que estén conectados a él.
8. Se simplifica el modelo obtenido utilizando las reglas.
9. Se asigna la causalidad a los elementos empezando por las fuentes, seguido de los elementos almacenadores (causalidad integral) y finalmente los elementos disipadores (causalidad arbitraria).

Para el sistema mecánico mostrado en la Figura 4.42 se identifican los puntos con diferentes velocidades como (a, b y c), los cuales serán representados en el diagrama de Bond Graph como uniones-1 cada uno de ellos, luego colocamos uniones-0 para unir los diferentes elementos al Bond Graph y se agregan los enlaces faltantes para terminar de unir todos los componentes del sistema, se identifica el enlace de referencia como el de la letra **a**, este se eliminará junto con los enlaces que están conectados a este enlace [18].

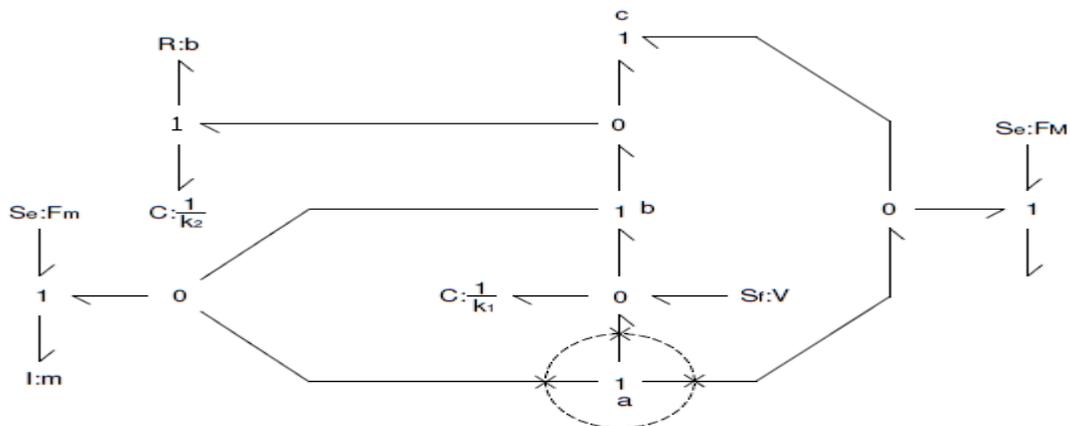


Figura 4.43. Procedimiento para obtener el Bond Graph del Sistema Mecánico mixto de traslación.

Una vez que se completa el modelo con cada uno de los elementos que lo conforman, se procede a identificar el nodo de referencia para así simplificar el modelo aplicando el paso 5, lo cual nos lleva a eliminar el nodo de referencia y los Enlaces que se encuentran conectados a este, finalmente se aplica el paso 6 con el cual se designa la



causalidad correspondiente a cada elemento para de esta manera obtener nuestro modelo Bond Graph el cual será como el mostrado en la Figura 4.44.

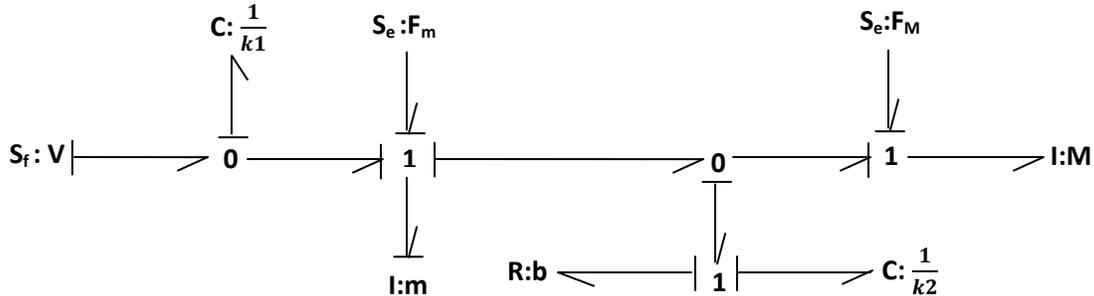


Figura 4.44. Bond Graph para el Sistema Mecánico de traslación.

4.1.4 PROCEDIMIENTO PARA LA OBTENCIÓN DE MODELOS EN BOND GRAPH PARA SISTEMAS HIDRÁULICOS

Para obtener el modelo en Bond Graph de un sistema hidráulico, hay que tener en cuenta que estos están caracterizados por alta presión y baja velocidad de fluido y los modelos se basan en la hipótesis de que toda la potencia del fluido de interés es el producto de la presión por el fluido [17].

Cuando el flujo de volumen es conservativo y el fluido es cercanamente incomprensible, los sistemas hidráulicos pueden ser manipulados como circuitos, en la Figura 4.45 se muestra un sistema hidráulico, y a partir de este sistema se desea obtener un modelo en Bond Graph.

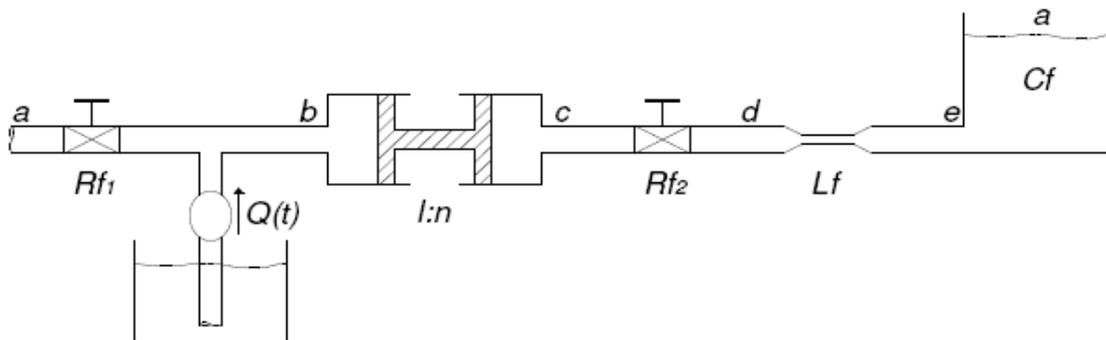


Figura 4.45. Sistema Hidráulico.



1. Se dibuja la unión-0 para cada presión distinta.
2. Se Inserta cada elemento, entre el uniones-0 apropiadas usando uniones-1, incorporar las fuentes de presión y de flujo.
3. Se define todas las presiones relativas a una presión de referencia (usualmente la referencia).
4. Se simplifica el modelo obtenido utilizando las reglas de simplificación mostradas anteriormente.
5. Se elimina la union-0 de referencia y los Enlaces que estén conectados a está.
6. Se asigna la causalidad a los elementos empezando por las fuentes, seguido de los elementos almacenadores (causalidad integral) y finalmente los elementos disipadores (causalidad arbitraria).

Aplicando el procedimiento descrito anteriormente obtendremos un modelo como el que es mostrado en la Figura 4.46.

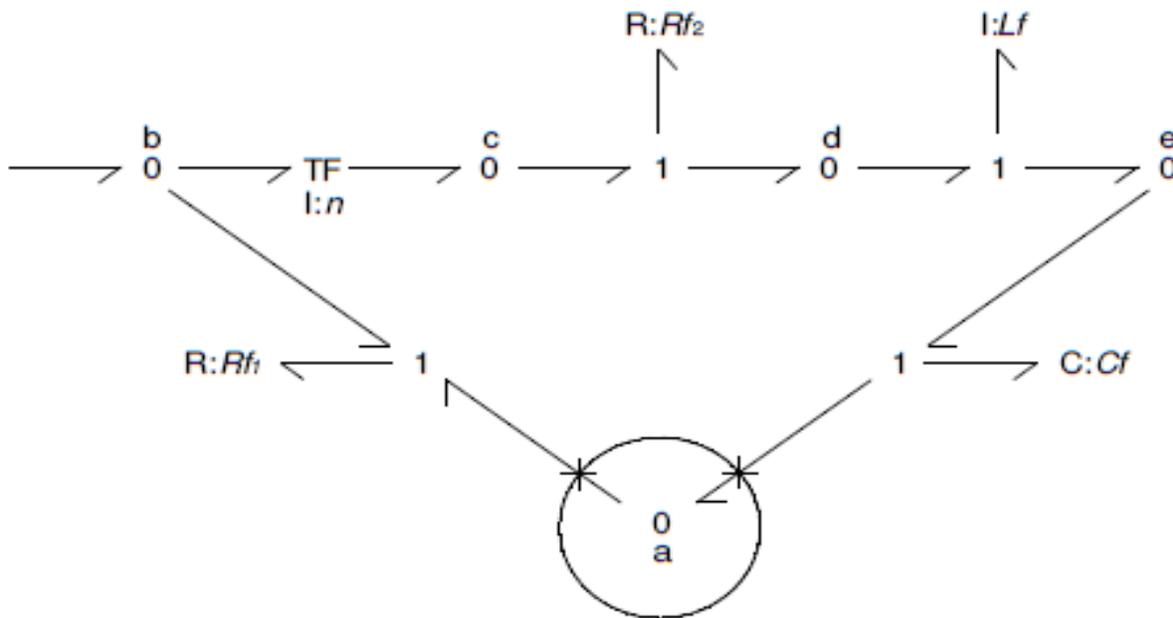


Figura 4.46. Procedimiento para obtener modelo en Bond Graph del Sistema Hidráulico.



Aplicando el paso 1 se identifican los nodos donde existe diferente presión (a,b,c) y se colocan uniones-0 por cada uno de ellos, en el paso 2 se colocan uniones-1 para unir cada elemento del Bond Graph, luego completan los enlaces faltantes para terminar de unir todos los elementos, una vez unidos todos los elementos se identifica la presión de referencia la cual en este ejemplo es la correspondiente al nodo **a**, esta junto con todos los enlaces que están conectados a ella se eliminaran y nuestro modelo de Bond Graph quedara como el mostrado en la Figura 4.47.

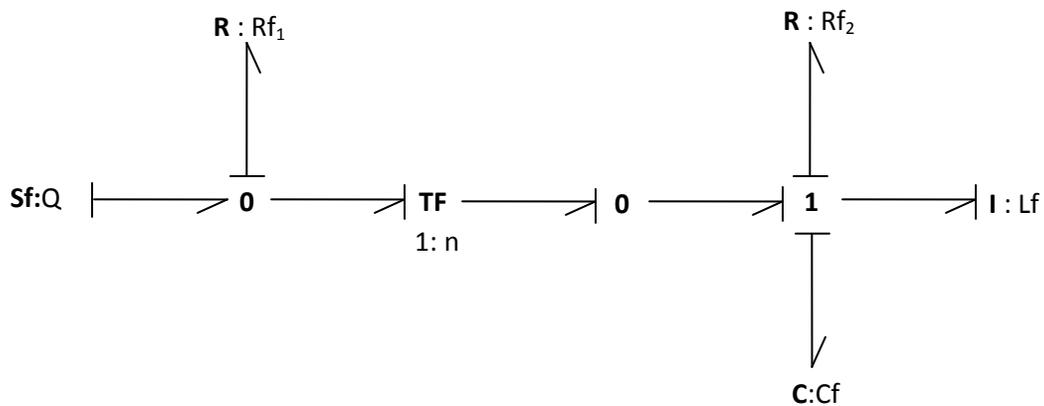


Figura 4.47. Bond Graph para el Sistema Hidráulico.

4.1.5 PROCEDIMIENTO PARA LA OBTENCIÓN DEL MODELO EN BOND GRAPH DE UN MOTOR DE CORRIENTE CONTINUA

En la Figura 4.48 se presenta un motor de Corriente Continua (CC) que se desea modelar utilizando la técnica de bond graph. El esquema del sistema físico mostrado incluye una inductancia, una resistencia eléctrica y una fuente de tensión del dominio eléctrico y del dominio mecánico la fricción en los rodamientos y la inercia rotacional. Se considera que el motor es de excitación separada y que la corriente de excitación i_f es fija, lo que hace que el flujo magnético Φ_f sea una constante, en consecuencia se desprecia el circuito de excitación en la modelación ya que no existe en el mismo ninguna variable. La siguiente serie de pasos para la modelación se rigen por el método desarrollado por el profesor Jan Broenink [6].

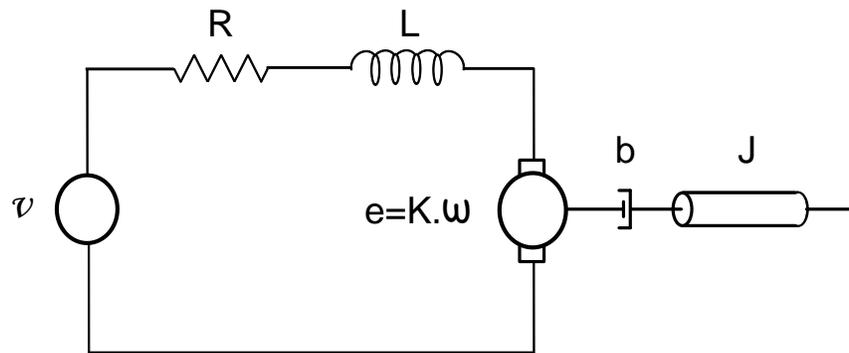


Figura 4.48. Esquema de un motor de CC.

1. Identificación de dominios y elementos:

- Una parte en el dominio eléctrico con una fuente de voltaje (S_e), un resistor (R), un inductor (I) y el puerto eléctrico del motor (un puerto del GY).
- Una parte en dominio mecánico rotacional con el puerto mecánico rotacional (un puerto del GY), la fricción de los rodamientos (R) y la inercia (I).

En la Figura 4.49, se indican los dos dominios y todos los elementos tienen un nombre único.

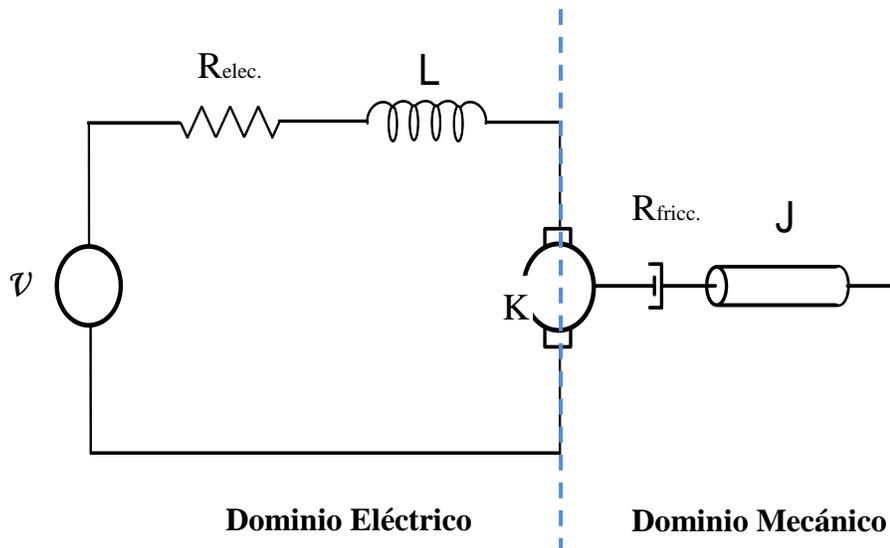


Figura 4.49. Identificación de dominios y elementos.



- Se indican las referencias en el modelo físico: el voltaje v_0 y la velocidad angular ω_0 . A la velocidad angular se le asigna una orientación positiva, obsérvese la Figura 4.50.

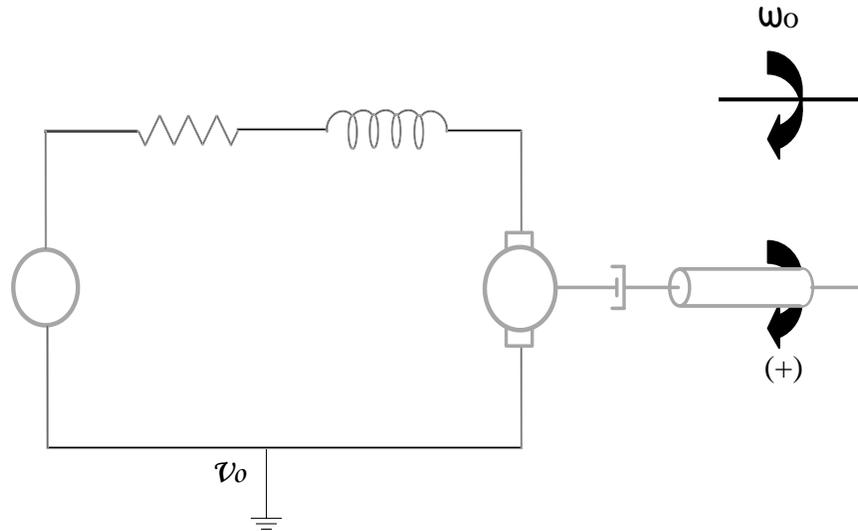


Figura 4.50. Identificación de las referencias.

- Se buscan e indican los otros voltajes y las otras velocidades relacionadas al modelo. Estas variables son v_1 , v_2 , v_3 , y ω_1 respectivamente. Observe la Figura 4.51.

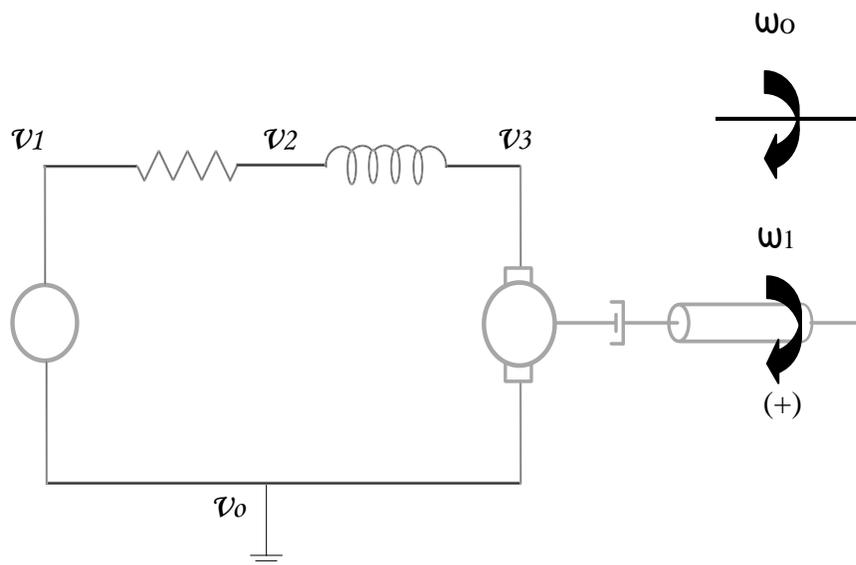


Figura 4.51. Identificación de los voltajes y velocidades relevantes.



- Las variables identificadas en el paso 3 se representan con uniones 0 y 1 respectivamente, como se muestra en la Figura 4.52, en un bosquejo compatible con la posición de las variables de la Figura 4.51. Las referencias no se dibujan ya que éstas serán eliminadas (las referencias valen 0 y no contribuyen en el comportamiento dinámico).

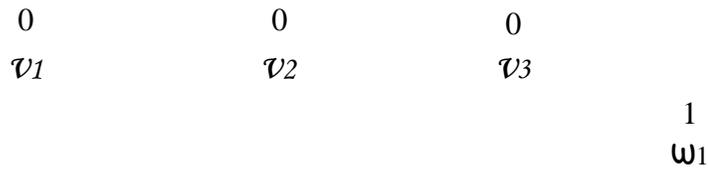


Figura 4.52. Primer bosquejo del Bond Graph. voltajes y velocidades con Uniones 0 y 1, respectivamente.

- Al chequear todos los puertos de los elementos identificados en el paso 1, en busca de diferencia de voltaje o velocidad, sólo se identificaron 2 diferencias de voltaje, v_{12} y v_{23} .
- Se dibujan las variables diferencia de voltajes en el Bond Graph, como se hace notar en la Figura 4.53. Luego de este Paso pueden conectarse los elementos a través de la estructura generada.

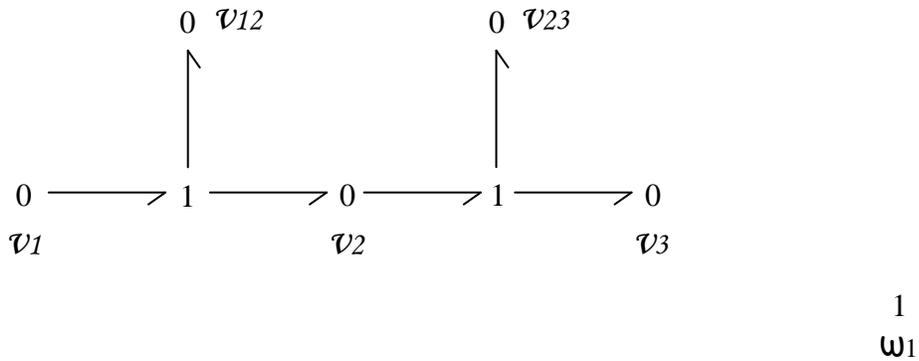


Figura 4.53. Inclusión de las diferencias de voltajes del modelo.



- 7. Todos los elementos se conectan a la respectiva unión como se muestra en la Figura 4.54. Nótese que los elementos de dominio no-mecánicos siempre se conectan a uniones 0 (esfuerzos o diferencias de esfuerzo) y los elementos del dominio mecánico están siempre conectados a uniones 1. Esta argumentación caracteriza y diferencia este procedimiento de otros anteriormente explicados.
- 8. Por último, el bond graph necesita ser simplificado para eliminar las uniones y enlaces que no son de relevancia, a pesar de la ayuda que aportan en la construcción del bond graph (ver Figura 4.55). Para simplificar, se aplican las reglas de simplificación explicadas en el paso 6 de la sección 4.1.1 de este capítulo.

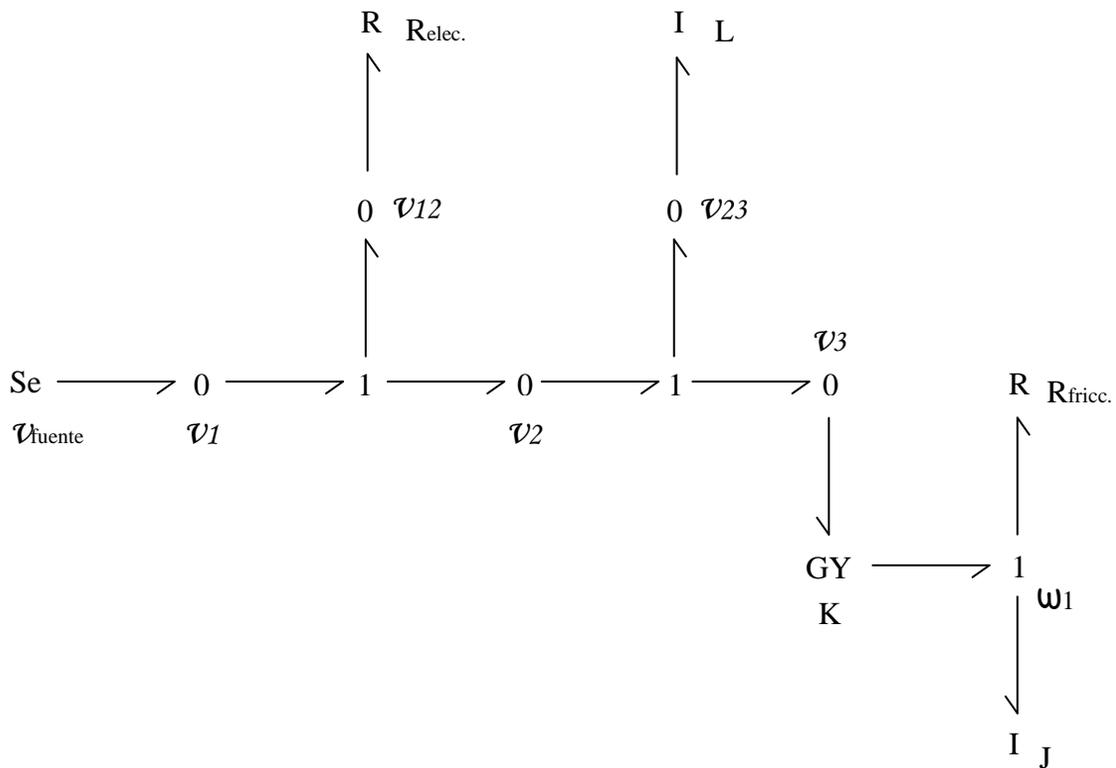


Figura 4.54. Bond Graph del modelo Motor de CC completado.

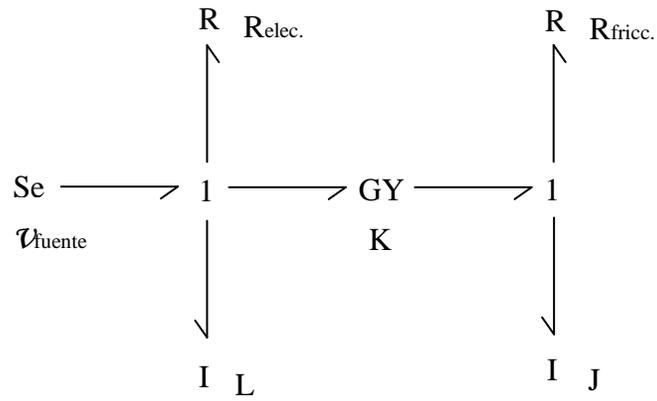


Figura 4.55. Bond Graph del modelo Motor de CC simplificado.



MANUAL 20-SIM VERSIÓN 3.6.0

5.1 INTRODUCCIÓN:

20-SIM es un programa de modelación y simulación avanzado que se ejecuta bajo el sistema operativo Microsoft Windows, en el cual se puede simular el comportamiento de sistemas dinámicos, tales como: sistemas eléctricos, mecánicos, hidráulicos, entre otros, o cualquier combinación de estos sistemas. Este software ha sido desarrollado en el Laboratorio de Control de la Universidad de Twente, como sucesor del famoso paquete de software TUTSIM y está basado en el lenguaje SIDOPS + (System Interdisciplinary Description Of Physical Systems) [4], el cual es un lenguaje de programación estructurada que es muy parecido a la notación matemática estándar, incluye 80 funciones integradas pero permite al usuario crear sus propias ecuaciones que más tarde definirán el comportamiento de los modelos. 20-SIM soporta por completo la modelación gráfica, permitiendo diseñar y analizar sistemas dinámicos de una manera sencilla, sin comprometer la dinámica de la potencia [19].

Este manual ayuda a que el usuario trabaje con 20-SIM en poco tiempo, y se caracteriza por enseñar al usuario a iniciar el programa y simular los ejemplos que se presentan en varios sub-capítulos.

Requerimientos de Software y Hardware:

- Windows 95,98,ME,2K,XP
- Requiere 71.79 MB de memoria libre para su instalación.
- Después de la instalación se necesita en el disco 80 MB de espacio.

5.2 CONOCIENDO 20-SIM

El software de 20-sim consta de 2 ventanas principales, una ventana secundaria y varias herramientas. La primera ventana es el *Editor*, la segunda el *Simulador*, la tercera es



la *Librería de Modelos* la cual se presenta como un explorador de archivos. El *Editor* abre cuando se inicia 20-sim (junto con la *Librería de Modelos*) y es en esta ventana donde se crean los modelos [19].

El *Editor* está conformado por 4 sub-ventanas:

- **Hierarchy** (jerarquía): se pueden agrupar partes del modelo como un submodelo. A esta organización se le denomina jerarquía.
- **Implementation** (implementación): es el espacio de trabajo donde se pueden introducir modelos gráficos y de ecuación.
- **Icon** (icono): todo modelo tiene su propia figura o icono. Los iconos se muestran y pueden ser editados en la sub-ventana *Icon*.
- **Type** (tipo): un modelo puede tener una interfaz definiendo sus puertos y parámetros. Dicha interfaz se muestra y puede ser editada en la sub-ventana *type*.

Observe las Figuras 5.1 y 5.2. En la Figura 5.1 se muestra un esquema sencillo para presentar la estructura organizativa del software 20-sim.

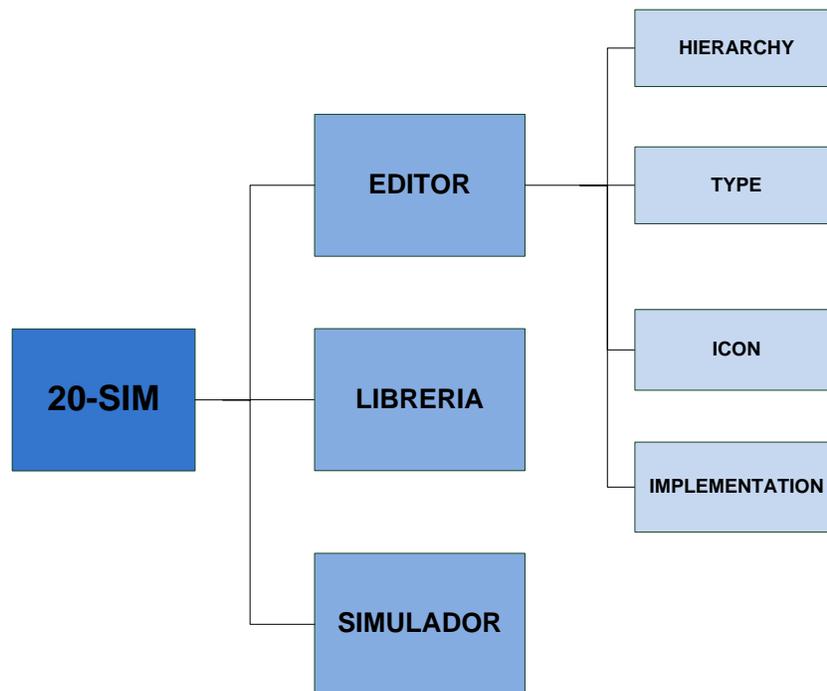


Figura 5.1. Estructura organizativa de 20-sim, versión 3.6.



Para conocer el menú de herramientas del editor de 20-Sim, se procede a mostrar el siguiente modelo `DiscreteController.em` (donde `.em` es la extensión para designar un modelo) proveniente de la carpeta **Demo\Control\Standard control**.

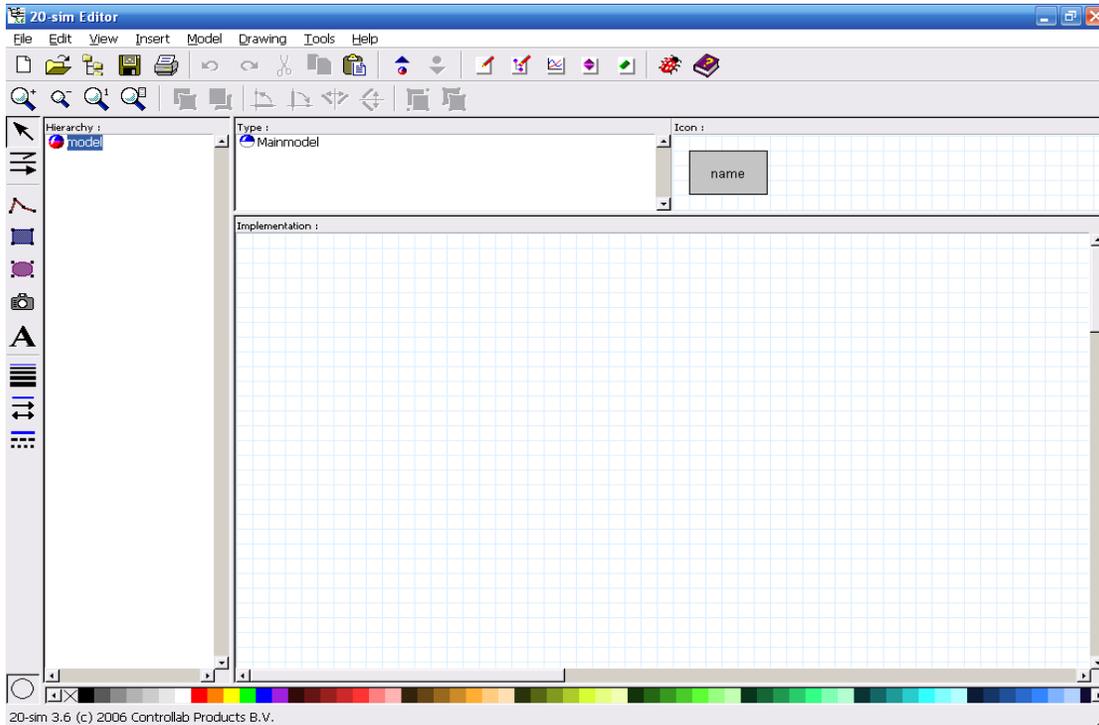


Figura 5.2. Ventana Principal de 20-SIM

1. Inicie 20-SIM y seleccione **File** y luego **Open**. Esta acción abre un explorador de archivos como se muestra en la Figura 5.3:

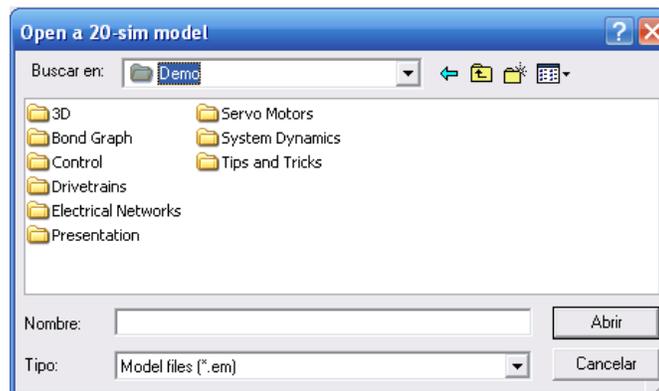


Figura 5.3. Explorador de Archivos.

2. Use los botones del Explorador de Archivos para ir a la carpeta **Demo**.

3. Diríjase a la carpeta **Control\Standard control**.
4. Seleccione el modelo **DiscreteController.em** con el puntero del ratón y seleccione el botón **Open**. Ahora el editor se debería mostrar como en la Figura 5.4:

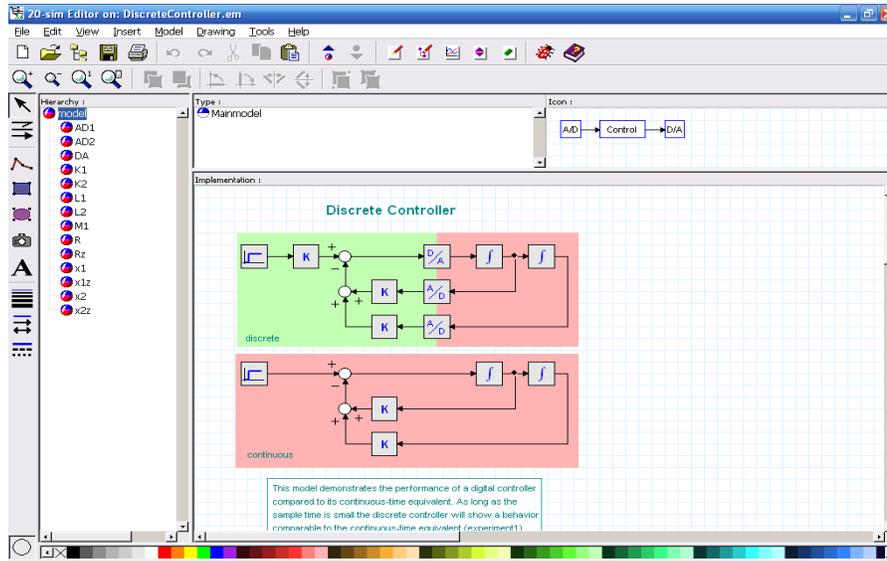


Figura 5.4. Editor mostrando el modelo DiscreteController.em

5. Diríjase al menú **Model** y seleccione el comando **StartSimulator** ó pulse el botón **StarSimulation** , se abrirá una ventana con el resultado del chequeo del modelo y la pantalla debe verse tal como la Figura 5.5.

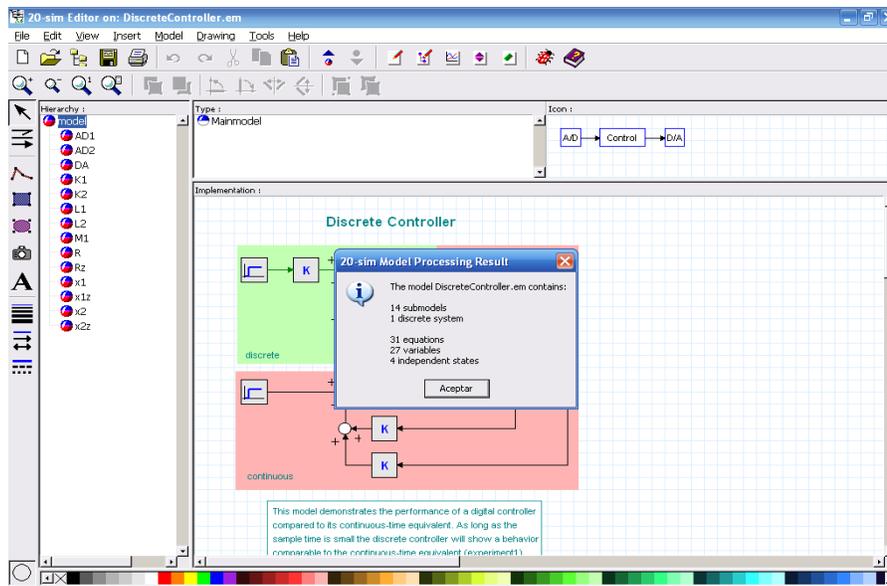


Figura 5.5. Ventana de chequeo del modelo.



Al realizar esta acción se verifica cuántas ecuaciones, variables, submodelos, entre otros, tiene el sistema que se está modelando. Luego de verificar si se corresponde con el modelo que se está ejecutando, se acepta y automáticamente se abrirá una nueva ventana: el Simulador. Se muestra en la Figura 5.6.

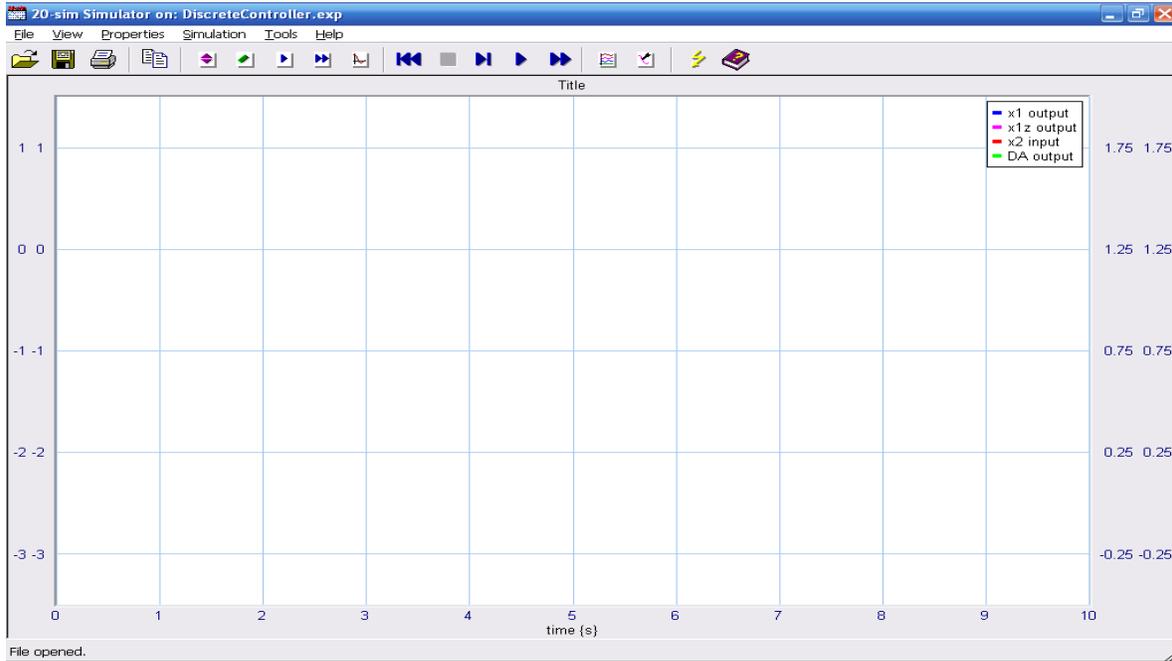


Figura 5.6. Ventana del Simulador de 20-sim.

En este simulador se pueden ejecutar simulaciones y mostrar los resultados gráficos.

Para analizar los resultados de las simulaciones se debe tener en cuenta que al utilizar el comando *StartSimulator* una serie de acciones se acaban de realizar:

- El modelo ha sido compilado para obtener un código de simulación.
- El simulador ha sido abierto y el código del modelo ha sido cargado dentro del simulador.
- El simulador ha detectado un experimento predefinido con el mismo nombre del modelo en el editor y en la misma carpeta (DiscreteController.exp), este experimento se carga automáticamente.



El experimento contiene valores de parámetros, ajustes de simulación, ajustes de trazado de las gráficas, entre otros. El nombre del experimento se muestra siempre en la parte superior del simulador. Debido a que el experimento es predefinido y ha sido cargado automáticamente el usuario sólo deberá llevar a cabo la siguiente instrucción:

6. Diríjase al menú *Simulation* y seleccione el comando *Run* (ejecutar).

Ahora se realizará la ejecución de la simulación de acuerdo a los ajustes del experimento predefinido. La Figura 5.7 deberá mostrarse en la pantalla.

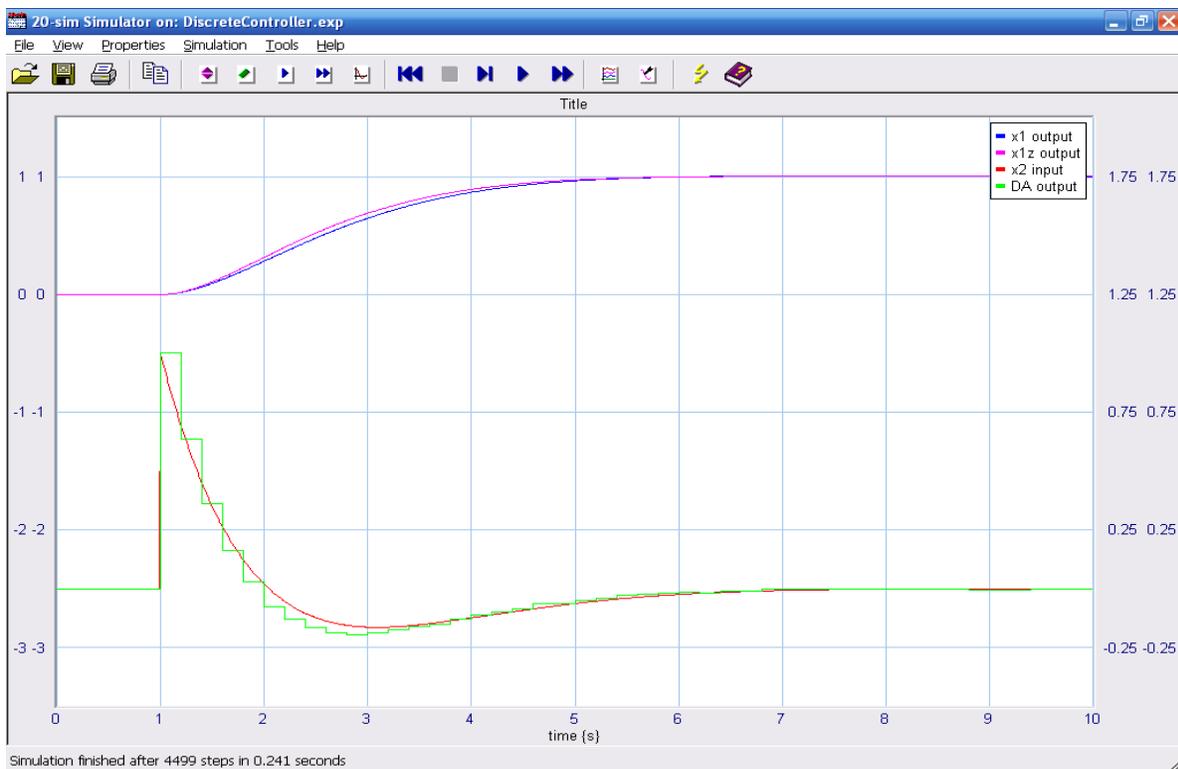


Figura 5.7. Resultados de la ejecución del modelo DiscreteController.em en el Simulador de 20-sim.

Recientemente se mostró como abrir y simular un modelo predefinido, ahora se anima al usuario a cargar y ejecutar otros modelos provenientes de la carpeta *Demo* y así, familiarizarse con el software 20-sim.

5.3 MODELACIÓN Y SIMULACIÓN CON 20-SIM

Anteriormente se explicó que 20-sim está compuesto por dos ventanas principales, el Editor y el Simulador. El primero se utiliza para introducir y editar modelos mientras que el segundo se usa para simular y analizar los modelos. Una sesión de modelación y simulación en 20-sim puede ser visualizada por el diagrama de flujo que se muestra en la Figura 5.8.

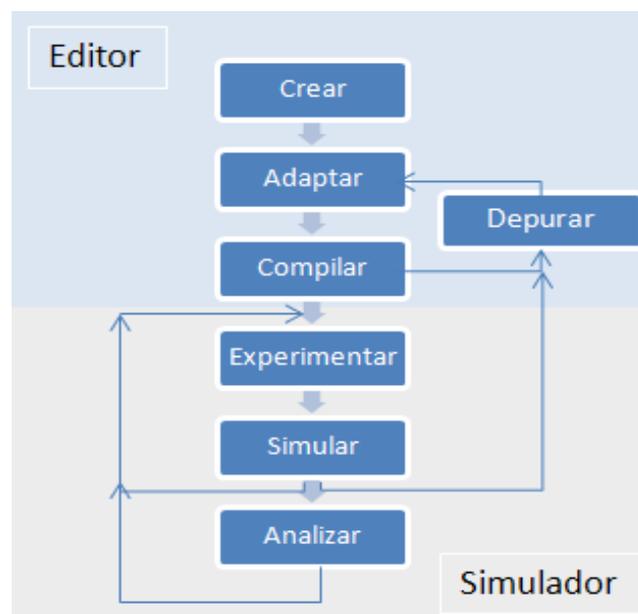


Figura 5.8. Modelación y Simulación en 20-sim.

5.3.1 Creación

Una sesión de modelación comienza con la creación de un modelo. La modelación en 20-sim se puede llevar a cabo utilizando ecuaciones, diagramas de bloque, bond graphs (grafos de enlace), diagramas icónicos o alguna combinación de estas representaciones. El usuario puede definir sus propios modelos o construirlos a partir de partes ya existentes que se almacenan en las diferentes bibliotecas que posee el programa [19].



5.3.2 Compilación/Depuración

Luego de que el modelo haya sido introducido satisfactoriamente, debe ser chequeado y compilado para generar el código de simulación (por ejemplo, archivo.exp). Esto se hace automáticamente en 20-sim cuando se abre el simulador por primera vez o cuando se chequea un modelo. En algunos casos el modelo contiene errores los cuales han de ser solventados mediante la adaptación (corrección) del modelo. Los errores en 20-sim se presentan en una ventana especial y apropiada para ayudar al usuario a resolverlos rápidamente (a esto se le llama: Adaptación) [19].

5.3.3 Experimentación

Después de una compilación exitosa, lo cual significa que no existen errores y que el simulador se abre con el modelo compilado y cargado se introducen los ajustes específicos tales como, el tiempo de inicio, tiempo de finalización, método de integración, entre otros. A este conjunto de ajustes se le reconoce como un Experimento en 20-sim y puede ser almacenado en un archivo de tipo experimento (archivo.exp) [19].

5.3.4 Simulación/Depuración

Cuando los ajustes de la simulación se han introducido, se intenta ejecutar la simulación. Ahora se inicia un lazo de iteración, el cual consiste en la ejecución de la simulación y el cambio de los ajustes de la simulación hasta que los resultados sean satisfactorios. Si la simulación falla (se generan errores), se comienza una sesión de depuración (corrección), esto significa que se tiene que reajustar el modelo, compilarlo y ejecutar la simulación hasta que los errores se hayan resuelto. Para llevar a cabo la simulación, el software 20-sim dispone de una serie de métodos de integración numérica (Revisar *Apéndice A*) [19].



5.3.5 Análisis

Si se ha completado una simulación satisfactoria, se puede analizar el modelo mediante el uso de varias herramientas tales como el análisis con el algoritmo de la transformada de Fourier (FFT), optimización, análisis de tolerancia, entre otros [19].

5.3.6 Modos de Operación

20-sim puede operar en dos modos: Modo Depuración y Modo Rápido. Este estado se muestra mediante el botón **Mode** en la barra de tareas, el cual puede mostrarse  para el modo depuración o  para el modo rápido. Se puede cambiar entre estos dos modos pulsando sobre el botón **Mode** [19].

En el Modo Depuración se realizan todos los chequeos posibles y se generan advertencias sobre los posibles errores del modelo. Durante la simulación se realizan chequeos adicionales tales como división entre cero, crecimiento al infinito, etc. Este modo ha de ser utilizado durante la construcción de los modelos y las primeras simulaciones. En el modo rápido, se realizan un mínimo de chequeos, se generan un mínimo de advertencias y la simulación se ejecuta a máxima velocidad. Este modo se usa cuando el modelo es validado y cuando las simulaciones deberían ser ejecutadas a máxima velocidad.

5.4 JERARQUÍA DE LOS MODELOS

Cualquier modelo o sistema principal en 20-SIM se describe como una composición de submodelos de nivel inferior. Estos mismos submodelos a su vez pueden ser descritos por la composición de submodelos de un nivel más inferior que los anteriores, etc. El nivel

más inferior consiste de submodelos de ecuaciones los cuales no se constituyen de otros submodelos. Ver Figura 5.9

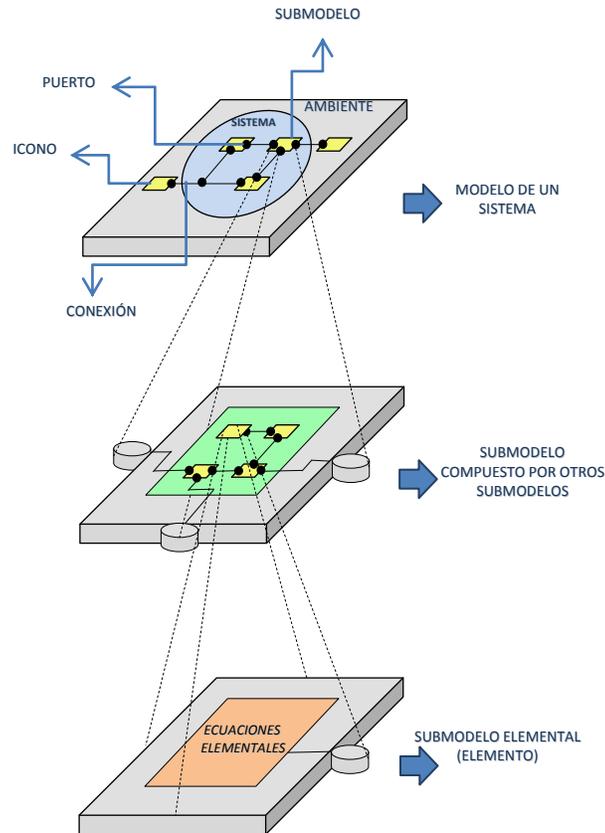
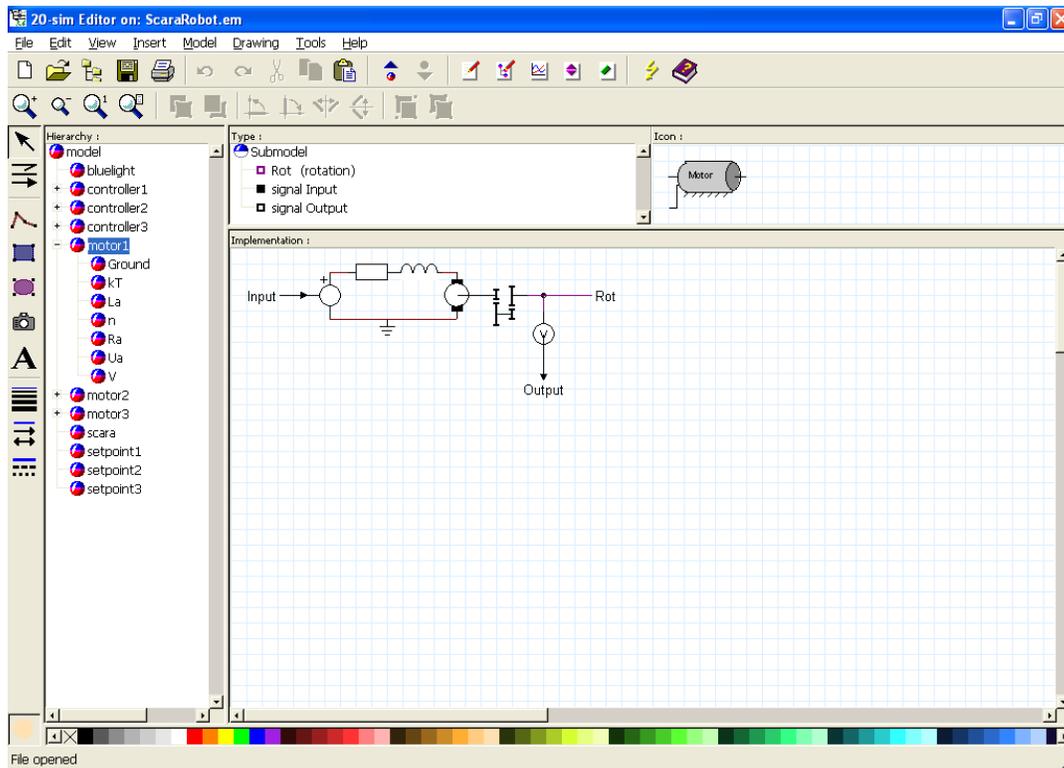


Figura 5.9. Esquema gráfico de la Jerarquía de un modelo.

Esta jerarquía se puede examinar en el *Editor*:

1. Inicie 20-SIM y seleccione **File** y luego **Open**.
2. Elija el modelo **ScaraRobot.em** de la carpeta **Demo/3D**, ahora el editor muestra el modelo completo, a la izquierda una estructura de árbol se visualiza mostrando la jerarquía del modelo
3. Con el puntero del mouse, seleccione el modelo **motor1** en la jerarquía del modelo. Ver Figura 5.10.

Figura 5.10. Submodelo **motor1**.

Como se puede notar el nombre del submodelo **motor1** se resalta en la jerarquía del modelo. Un signo positivo al lado izquierdo del nombre del submodelo indica que ese mismo submodelo contiene otros submodelos. En la ventana **Implementation** se muestra el submodelo, que consiste en un diagrama icónico con partes eléctricas y mecánicas.

Cada submodelo se representa con un círculo, existen 3 tipos, como se observa en la Figura 5.11:

- Azul-Gris: modelo sin descripción interna
- Azul-Rojo: modelo con descripción interna
- Azul-Amarillo: modelo encriptado

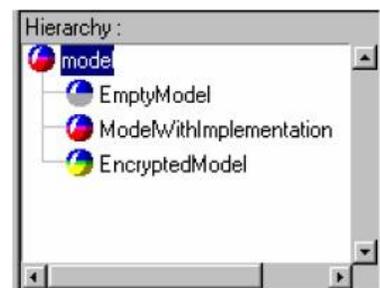


Figura 5.11. Tipos de Submodelos.

Para navegar a través de la jerarquía pulse sobre el signo positivo al lado del modelo **motor1** y proceda como se describió antes. También es posible seleccionar un modelo en la ventana *Implementation* y pulsar el botón *Go Down*  o el botón *Go Up* .

4. En la ventana *Implementation* pulse sobre el resistor.

5. Ahora pulse sobre el botón *Go Down* , la ventana debe mostrarse como en la Figura 5.12, indicando la ecuación de la cual se compone el submodelo.

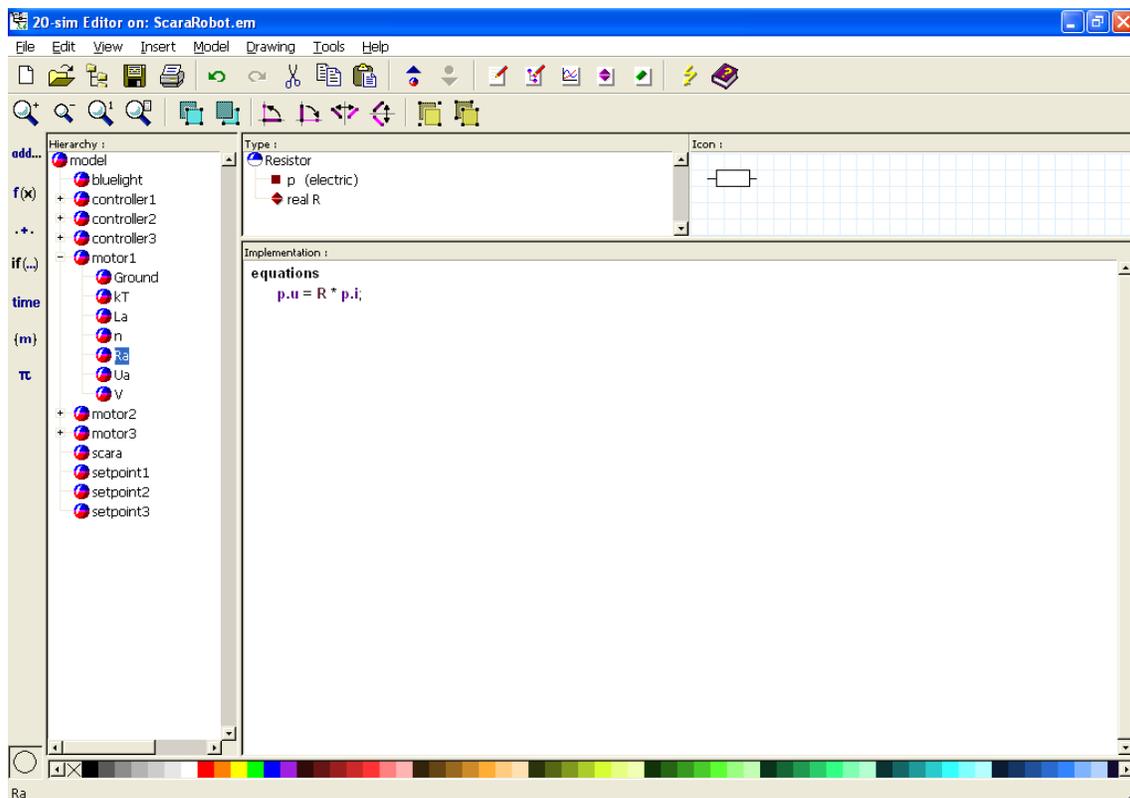


Figura 5.12. Visualización de la jerarquía de modelos y submodelos.

Como se puede notar, el modelo **Ra** está resaltado en la jerarquía del modelo. La ausencia de este signo positivo indica que este submodelo se encuentra en el nivel más inferior y por lo tanto será un submodelo de ecuación (sólo los submodelos de ecuación pueden estar en nivel más inferior de la jerarquía de modelos y submodelos). El submodelo



Ra se muestra en la ventana *Implementation* y consiste de una simple ecuación que describe el voltaje en función de la corriente.

5.5 TIPOS DE MODELOS

El software 20-sim soporta varios tipos de modelación:

- Modelos de Ecuaciones
- Modelos de Diagrama de Bloques
- Modelos de Bond Graph
- Modelos de Diagramas Icónicos

Todos los tipos de modelos se pueden introducir y modificar en el *Editor* de 20-sim, además pueden ser independientes (modelo principal) o formar parte de un modelo (submodelo). En las siguientes descripciones preliminares se observa una descripción breve de cada uno de los tipos de modelación que soporta el software 20-sim y algunos ejemplos del uso como modelos principales o submodelos.

5.5.1 Modelos de Ecuaciones

Los modelos de ecuaciones son apropiados para la introducción de todo tipo de ecuaciones lineales y no-lineales; es importante resaltar que sólo estos modelos pueden formar parte del nivel más inferior de la jerarquía de un modelo o submodelo, es decir la base de cualquier submodelo o modelo en el software 20-sim es una (o varias) ecuación. Un ejemplo de un modelo de ecuación principal se muestra a continuación: el modelo **FunctionMaximization.em** de la carpeta **Demo\Tips and Tricks**. Ver Figura 5.13.

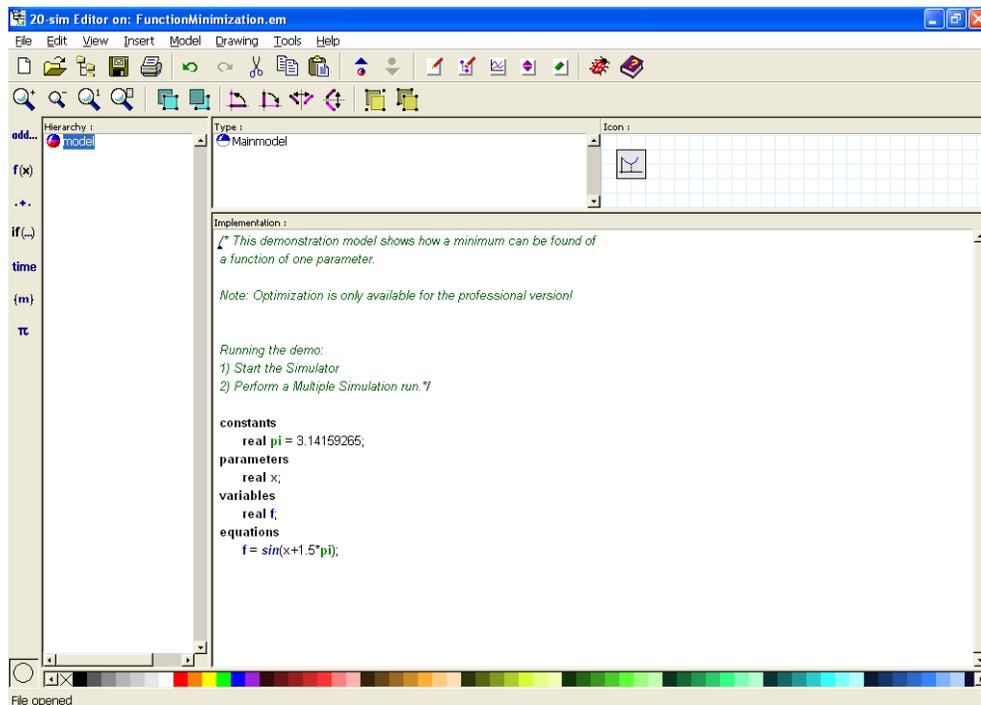


Figura 5.13. Modelo de Ecuación Principal.

Un ejemplo de submodelos de ecuacion se muestra a continuacion, es el submodelo Integral del modelo **DiscreteController.em** que se encuentra en la carpeta **Demo\Control\Standard Control**. Ver Figura 5.14.

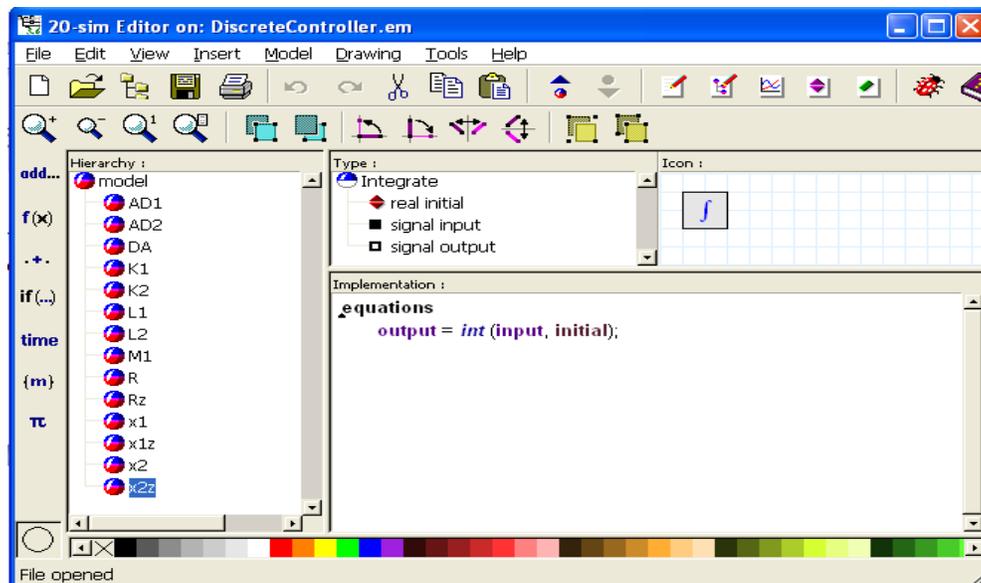


Figura 5.14. Visualización de la jerarquía de modelos y submodelos.

5.5.2 Modelos de Diagrama de Bloques

Los diagramas de bloques se adecuan muy bien para modelos que poseen un claro flujo de información y pueden ser separados en bloques distintos. Un ejemplo del diagrama de bloques es el modelo **DiscreteController.em** de la carpeta **Demo\Control\Standard Control**. Ver Figura 5.15.

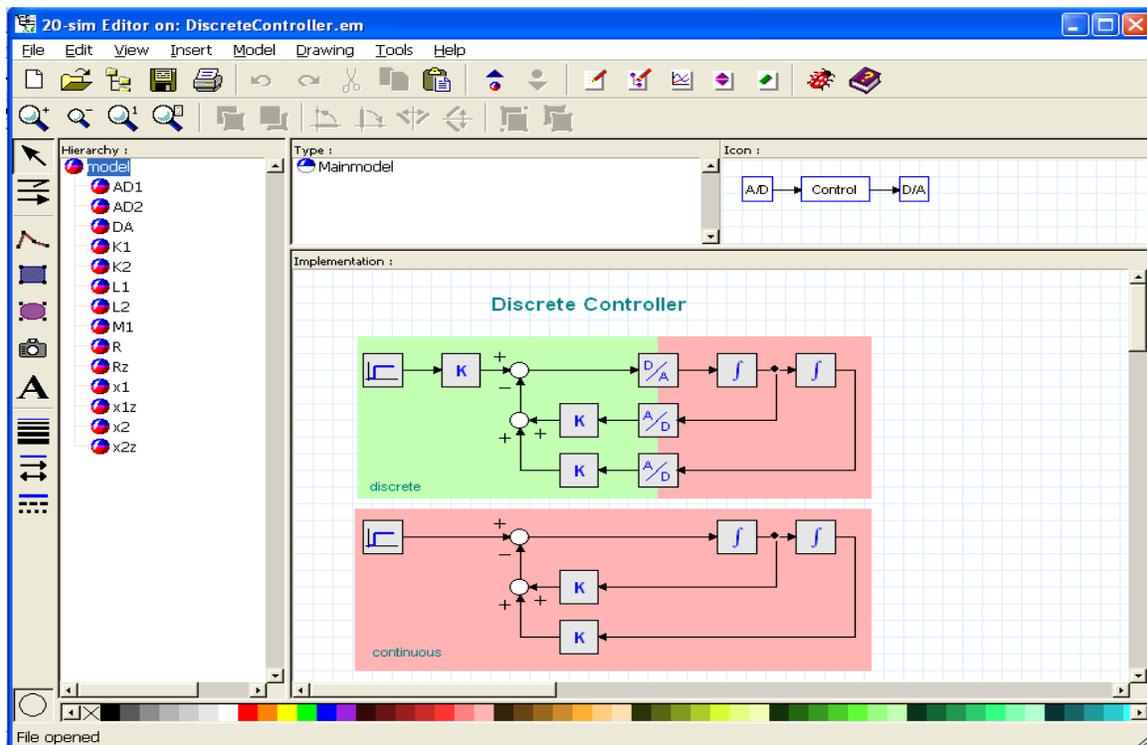


Figura 5.15. Modelo Principal de Diagrama de Bloques.

Un ejemplo del submodelo del diagrama de bloques es el submodelo **LinearMotor.em** que se encuentra en la ruta de acceso **Demo\Control\Advanced Control** y se muestra a continuación. Ver Figura 5.16.

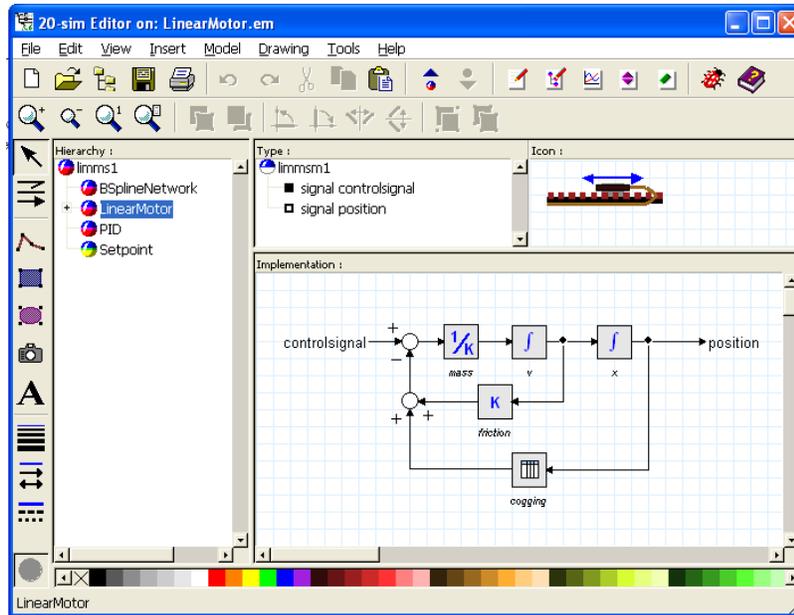


Figura 5.16. Submodelo de Diagrama de Bloques.

5.5.3 Modelos de Bond Graph

Los modelos de Bond Graph se adecuan muy bien para la modelación de sistemas físicos mostrando una clara similitud con los componentes de un sistema físico. El modelo **SimpleRobot.em** de la ruta de acceso Demo\Bond Graph se muestra en la Figura 5.17.

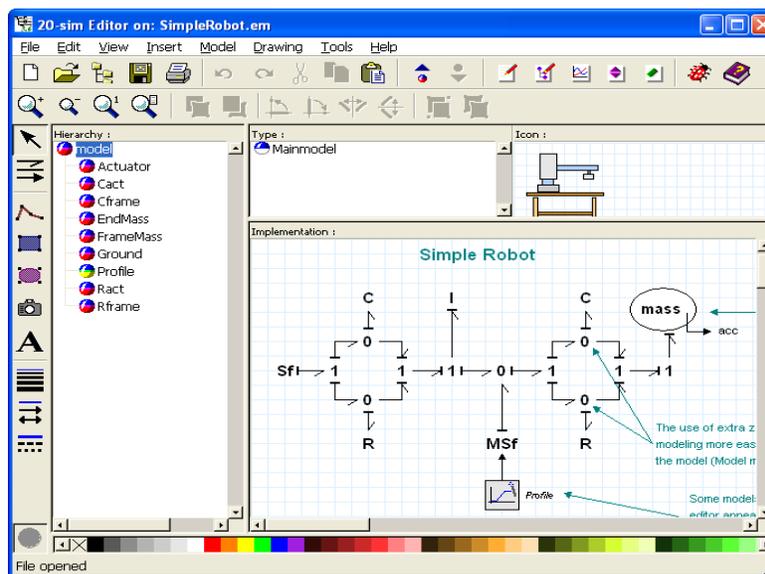


Figura 5.17. Modelo Principal de Bond Graph.

Seguidamente se muestra en la Figura 5.18 el submodelo **Hydraulic** del modelo **FluidLevelControl.em** de la carpeta **Demo\Control\Standard Control** el cual es un ejemplo de un submodelo de Bond Graph.

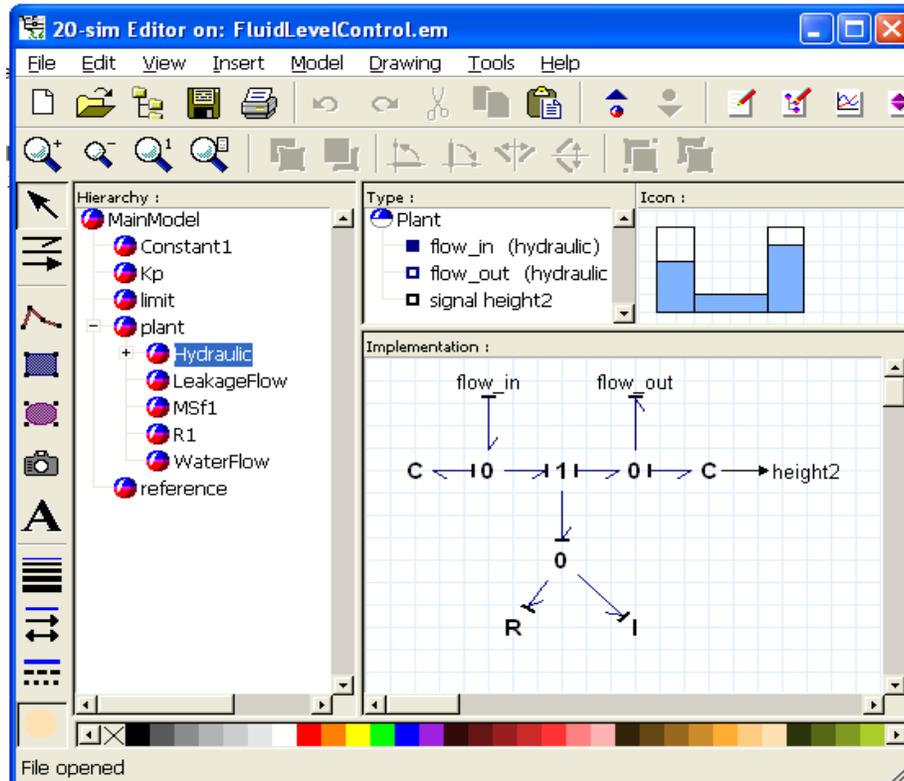


Figura 5.18. Submodelo de Bond Graph.

5.5.4 Modelos de Diagramas Icónicos

Los diagramas icónicos son apropiados para la modelación de sistemas físicos en los cuales se desea tener una imagen de modelación muy similar a la realidad del sistema en estudio. El modelo **Linearize.em** de la carpeta **Demo\Tips and Tricks** es un ejemplo de un modelo principal de diagramas icónicos. Se muestra en la Figura 5.19.

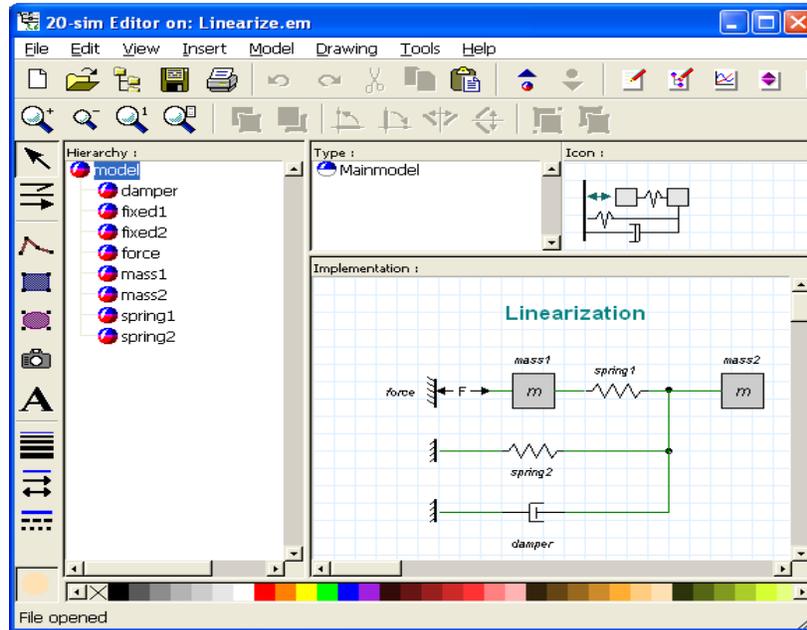


Figura 5.19. Modelo Principal de Diagramas Icónicos.

El submodelo **motor1** del modelo **ScaraRobot.em** de la ruta de acceso **Demo\3D** es un submodelo basado en diagramas icónicos, se muestra en la Figura 5.20.

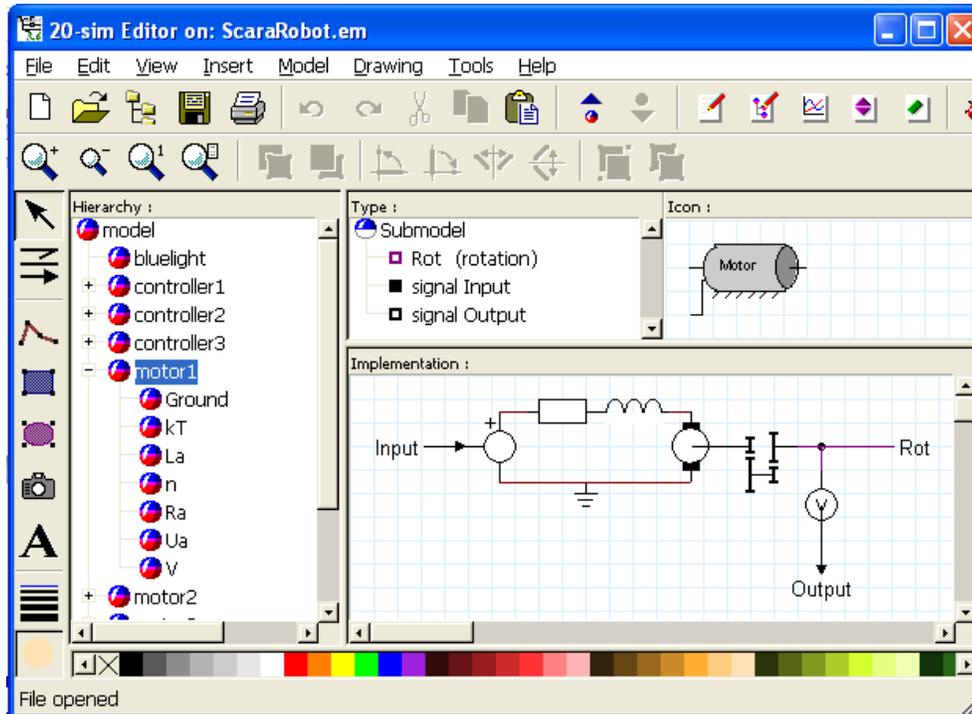


Figura 5.20. Submodelo de Diagramas Icónicos.



5.6 BIBLIOTECA DE MODELOS

20-SIM posee una gran colección de modelos almacenados en el archivo de la biblioteca. Seleccionando, arrastrando y llevando estos modelos de la biblioteca hasta el editor se pueden crear modelos propios. Para inspeccionar los modelos de la biblioteca se puede abrir un navegador en la biblioteca en el editor de 20-sim.

1. Inicie 20-Sim.
2. Seleccione el botón *Open Library*  o desde el menú *File* seleccione el comando *Open Browser*.

Ahora se abrirá un navegador mostrando la carpeta de la biblioteca de 20-Sim, según el tipo de sistema operativo Windows, la aparición de este navegador puede cambiar un poco.

En Windows XP se muestra como en la Figura 5.21.

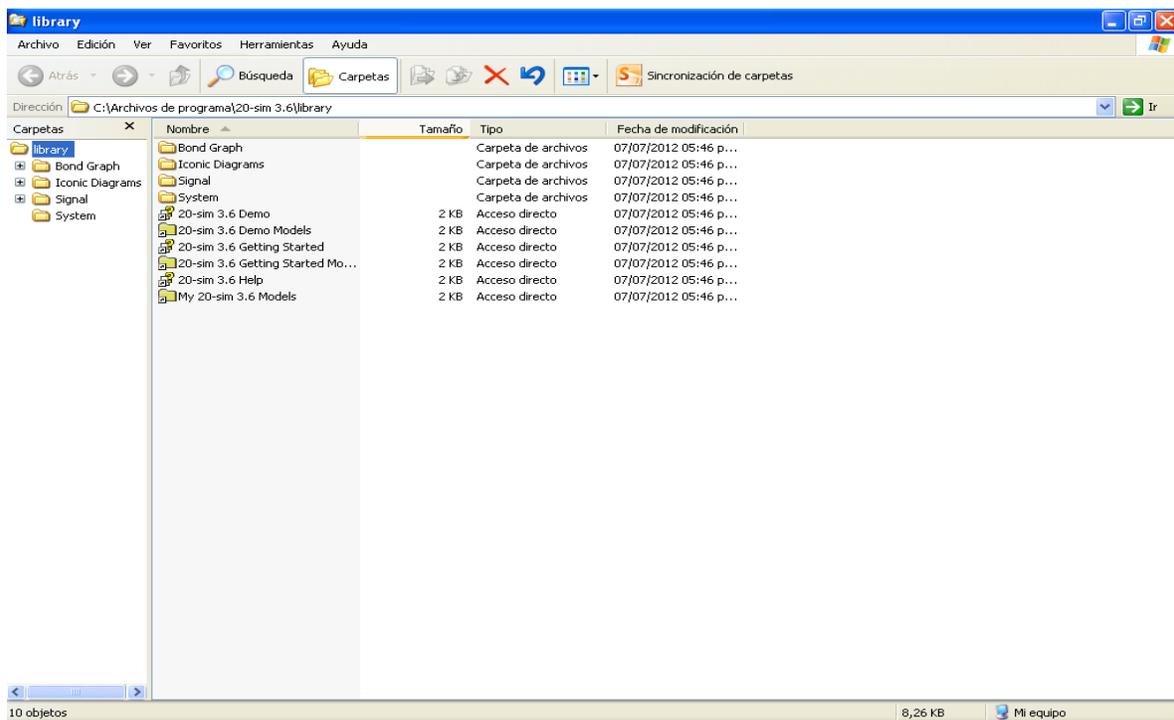


Figura 5.21. Navegador de la biblioteca de 20-sim.



3. Seleccione a la izquierda de la ventana la carpeta **Signal** y a continuación la biblioteca **Block Diagram**.

Ahora el navegador de la biblioteca debe ser similar a la Figura 5.22.

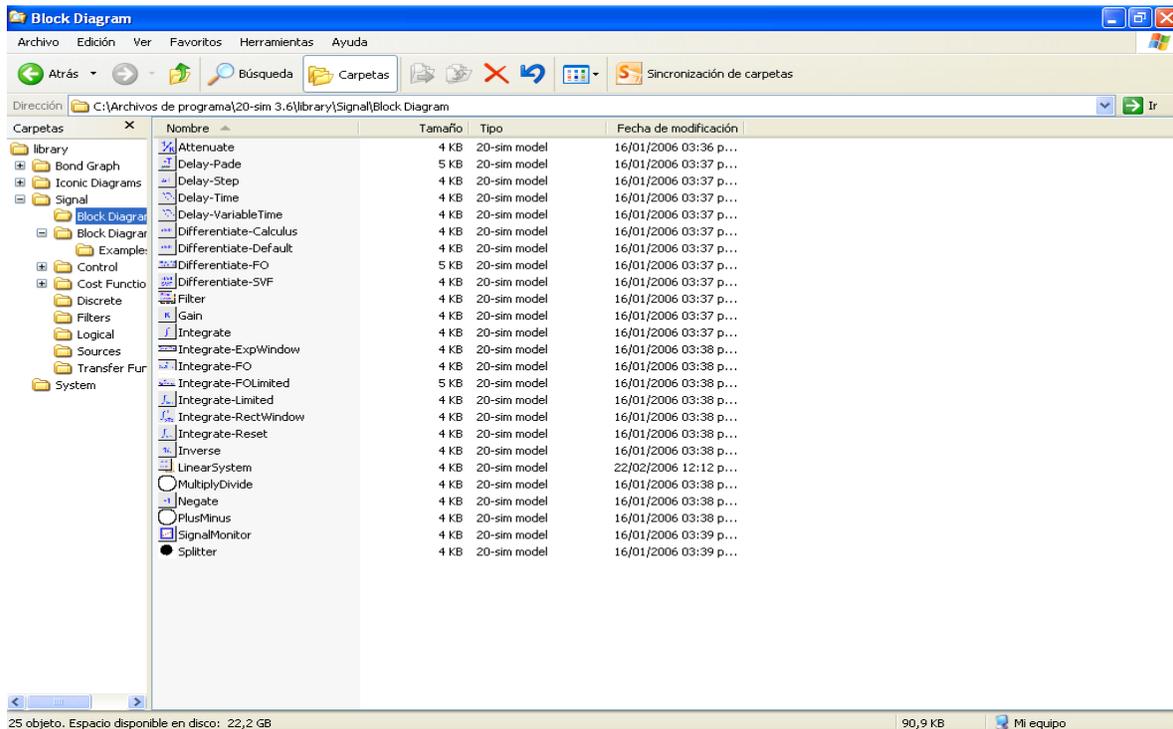


Figura 5.22. Contenido de la carpeta **Block Diagram**.

Como se puede ver la biblioteca está llena de modelos, para cambiar el aspecto del modelo presione el botón **Vistas** .

4. Haga clic en el menú **Ver** en la barra de herramientas y seleccione **Vistas en miniatura** (válido para Windows XP, si tiene otra versión de Windows esta instrucción puede variar).

Ahora el navegador de la biblioteca debe mostrar iconos grandes tal como en la Figura 5.23.

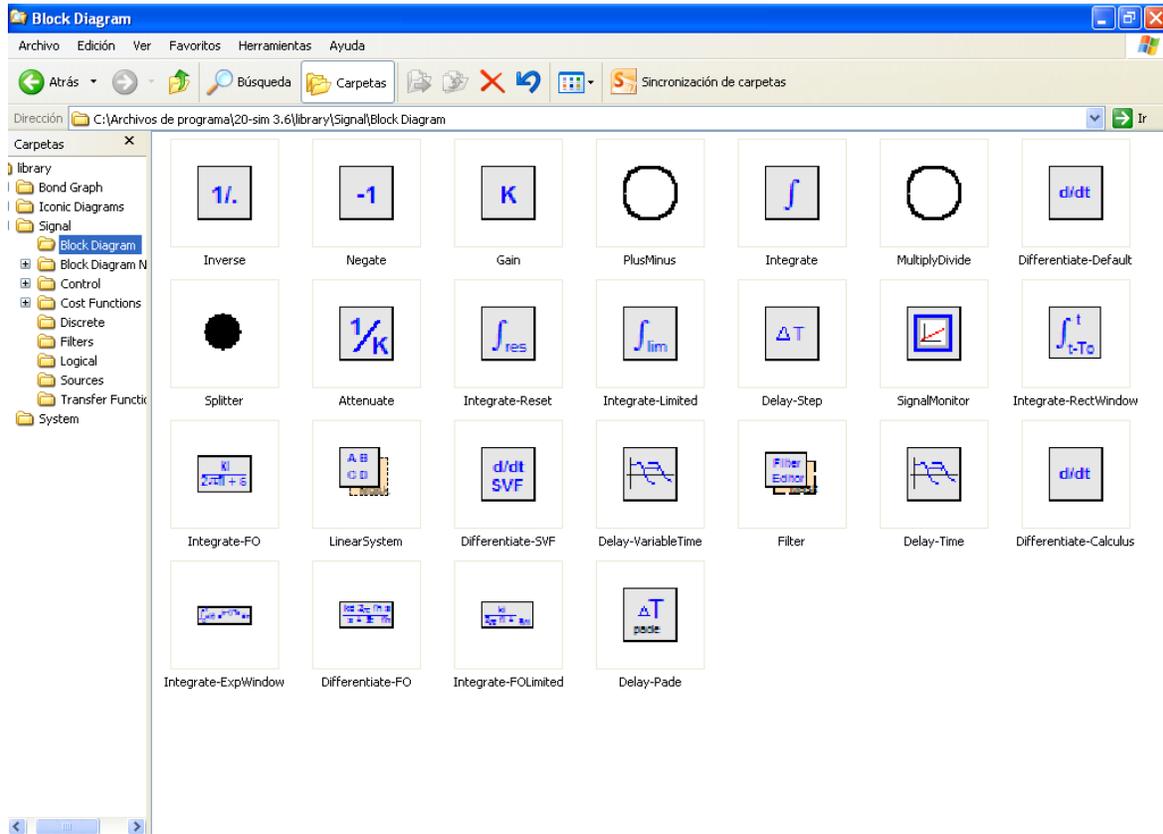


Figura 5.23. Reorganización del contenido de la carpeta *Block Diagram*.

Esta acción ayuda a seleccionar modelos. Todos los modelos en la biblioteca 20-SIM pueden ser arrastrados y llevados al editor del 20-SIM siguiendo las siguientes instrucciones:

5. Seleccione el editor de 20-Sim y en el menú *File* seleccione *New Model* y *Grafical Model*.
6. Ponga el navegador de la biblioteca junto al editor.

Debe aparecer una sub-ventana de cuadrícula la cual indica que se puede crear un modelo grafico. Ver Figura 5.24.

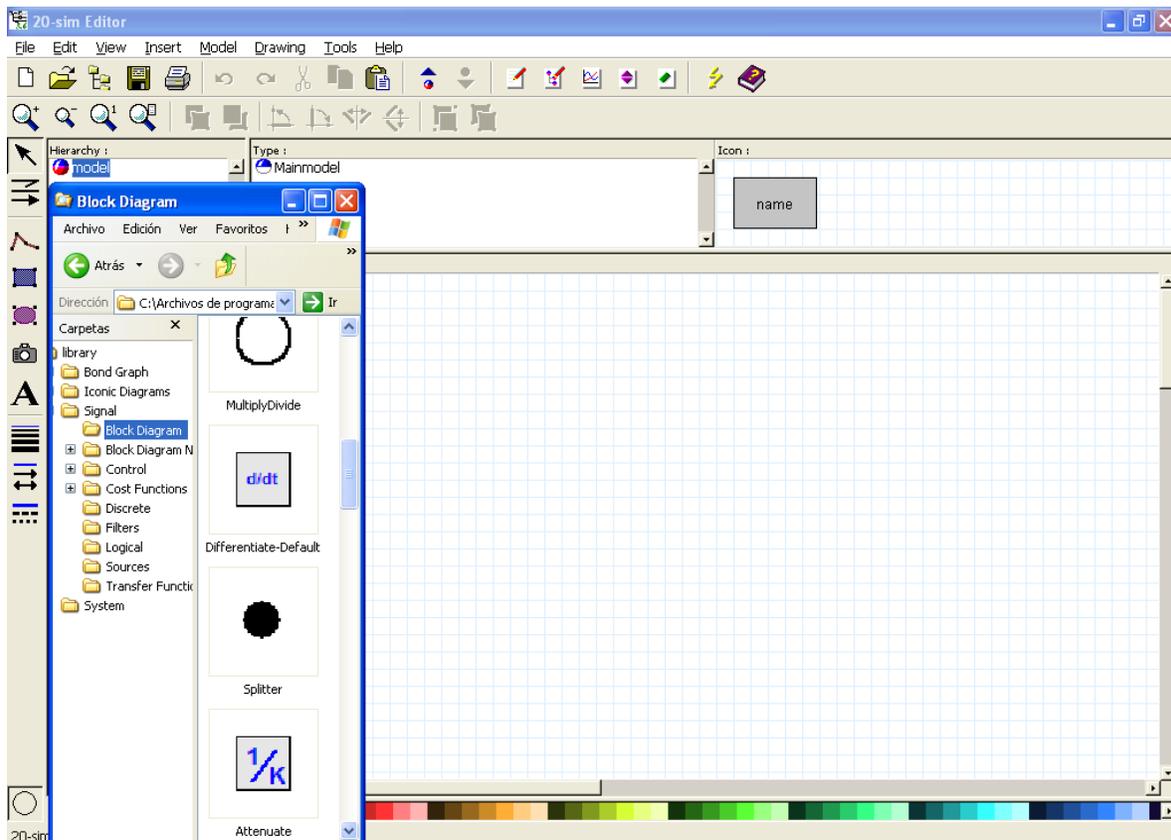


Figura 5.24. Reorganización del espacio de trabajo en 20-sim.

7. Para continuar selección de la biblioteca el modelo Attenuate.em (clic con el ratón).
8. Coloque el puntero del ratón en la parte superior del modelo Attenuate.em
9. Pulse el botón izquierdo del ratón y manténgalo pulsado.
10. Deslice el puntero del mouse hacia la sub-ventana **Implementation** del editor (**Drag**, Arrastrar).
11. Suelte el botón izquierdo del ratón (**Drop**, Soltar).

Si se tiene éxito, un icono semi-transparente se mueve con el puntero del ratón hacia el editor. Una vez realizada esta acción la ventana se muestra en la Figura 5.25.

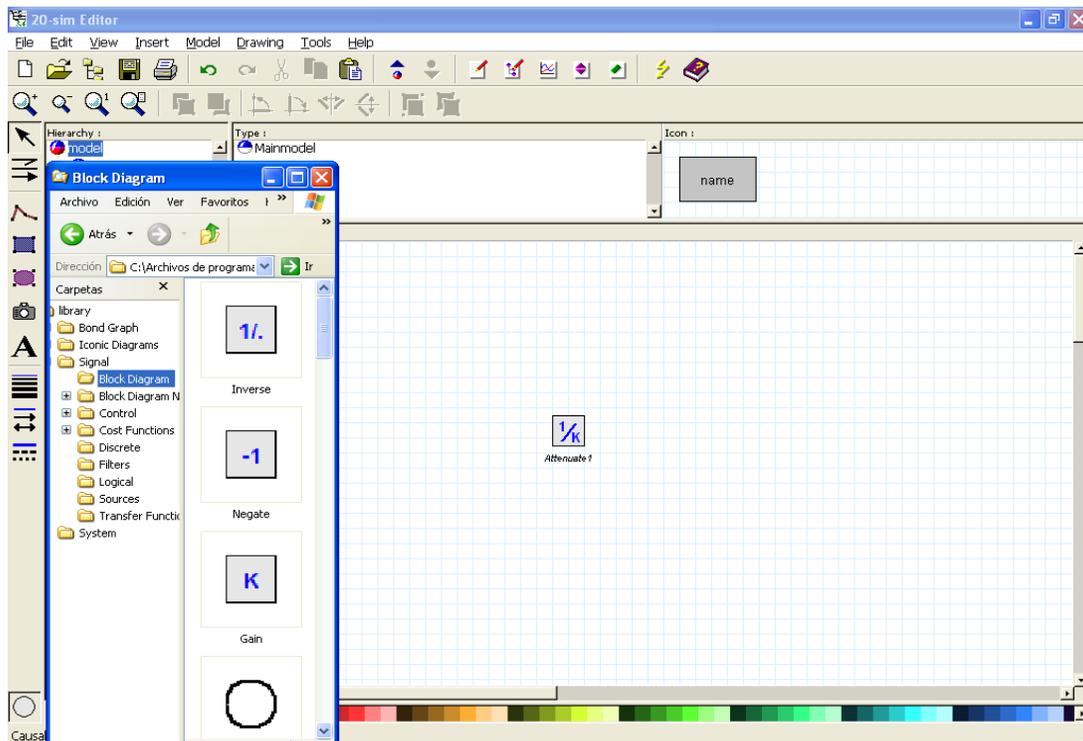


Figura 5.25. Añadiendo un submodelo al Editor desde la Biblioteca de 20-sim.

En la realización de estos 11 pasos de fácil ejecución se muestra el uso de la biblioteca de 20-sim y su practicidad en la creación de nuevos modelos, experiencia que permite el desarrollo con más agilidad de las próximas secciones de este manual.

5.7 EXPERIMENTOS

Se le denomina “experimento” al conjunto de parámetros específicos, tales como: tiempo de inicio, tiempo de finalización, método de integración, etc., que se establecen para simular un modelo que ya ha sido creado y chequeado en el Editor de 20-sim. Una vez introducido un modelo se puede abrir el Simulador (ventana donde se configuran los parámetros necesarios para la simulación) y se puede almacenar en un archivo de tipo

experimento con la extensión “.exp”. En esta sección se explican los elementos básicos de un experimento.

1. Abra 20-SIM y seleccione *File* y *Open*.
2. Desde la carpeta **Demo\ Tips and Tricks** seleccione el archivo modelo **Linearize.em**.

El editor muestra el modelo **Linearize.em** como se ve en la Figura 5.26.

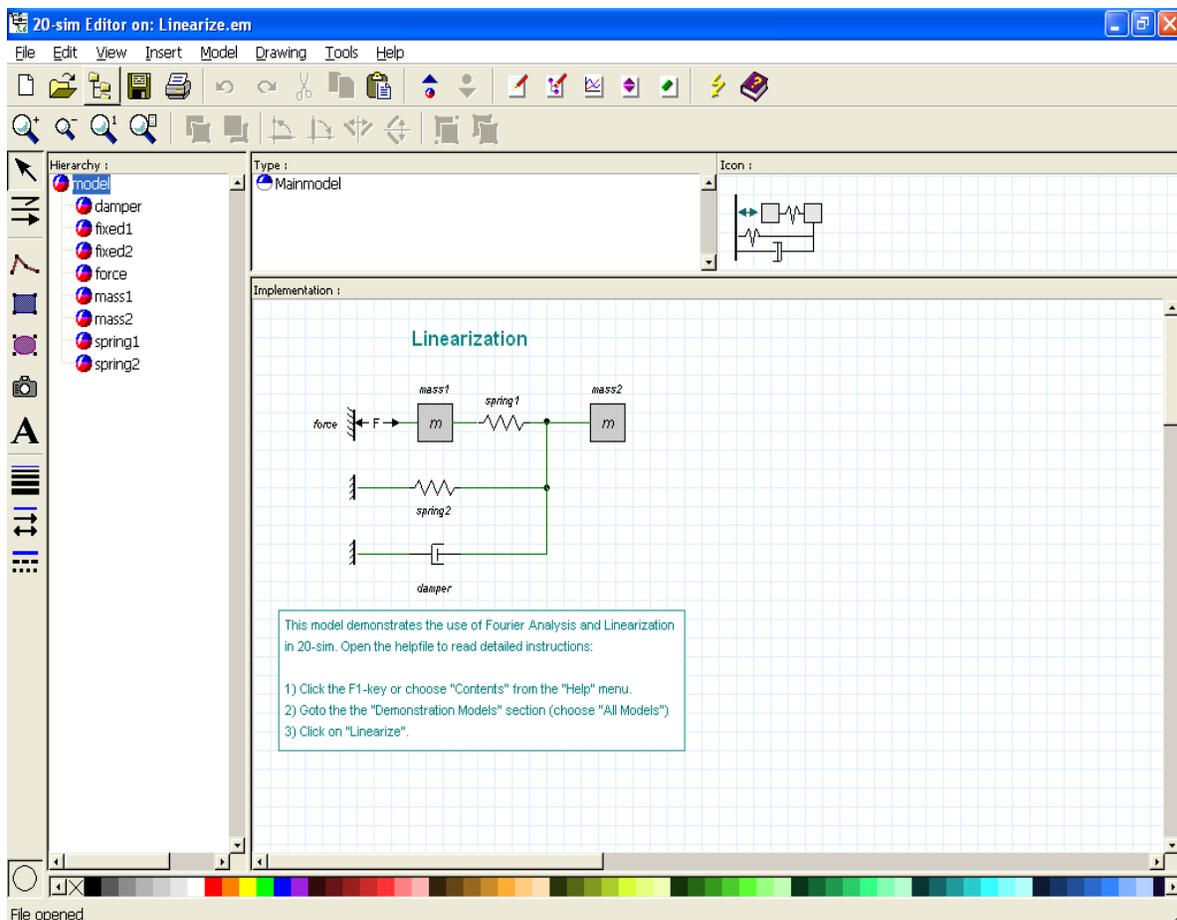


Figura 5.26. Modelo **Linearize.em**.

3. En la barra de herramientas del editor, pulse  o seleccione el comando *Start Simulator* del menú *Model*, esta acción abre la ventana del simulador.



4. Pulse  en la barra de herramientas del simulador, o desde el menú *Simulation* seleccione el comando **Run**.

El paso 4 lleva a cabo una simulación y los resultados de ésta se muestran en un gráfico. Ahora la ventana del simulador debe verse como en la Figura 5.27.

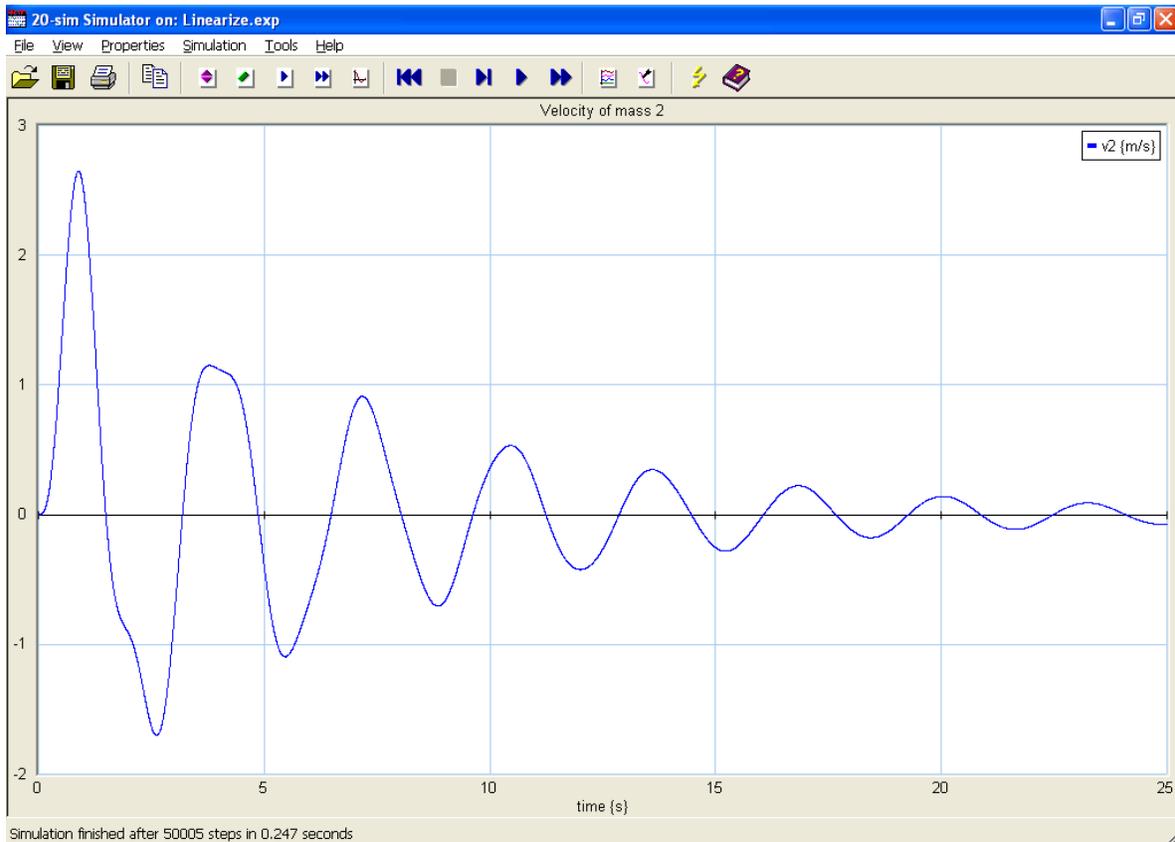


Figura 5.27. Resultados de la simulación del modelo **Linearize.em**

Cuando se usa el comando *Start Simulator* el modelo se ha compilado y cargado en el simulador, además, se ha cargado un experimento predefinido con el mismo nombre que el modelo **Linearize.em** en el editor y en la misma carpeta (**Linearize.exp**). En los próximos pasos se muestran los elementos básicos en un experimento.

Valores de los Parámetros

5. En la barra de herramientas, pulse  o desde el menú *Properties*, seleccione el comando *Parameters*. La ventana del editor de parámetros se muestra a continuación en la Figura 5.28.

En el editor de parámetros se pueden cambiar valores de parámetros, incluso durante una simulación.

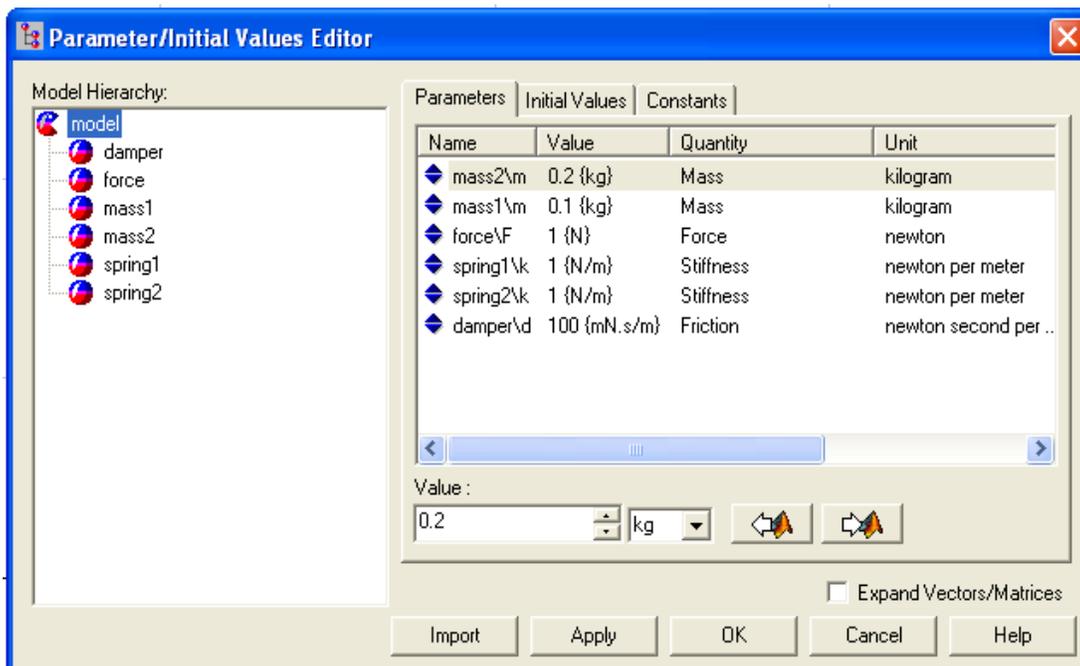


Figura 5.28. Editor de Parámetros.

6. Cambie el parámetro **mass2\m** a 0.2 kg.
7. Cierre el editor de parámetros pulsando el botón OK.
8. En la barra de herramientas, pulse  o desde el menú *Simulation* seleccione el comando *Run*. Una segunda simulación es realizada y los resultados también son dibujados en la pantalla como en la Figura 5.29.

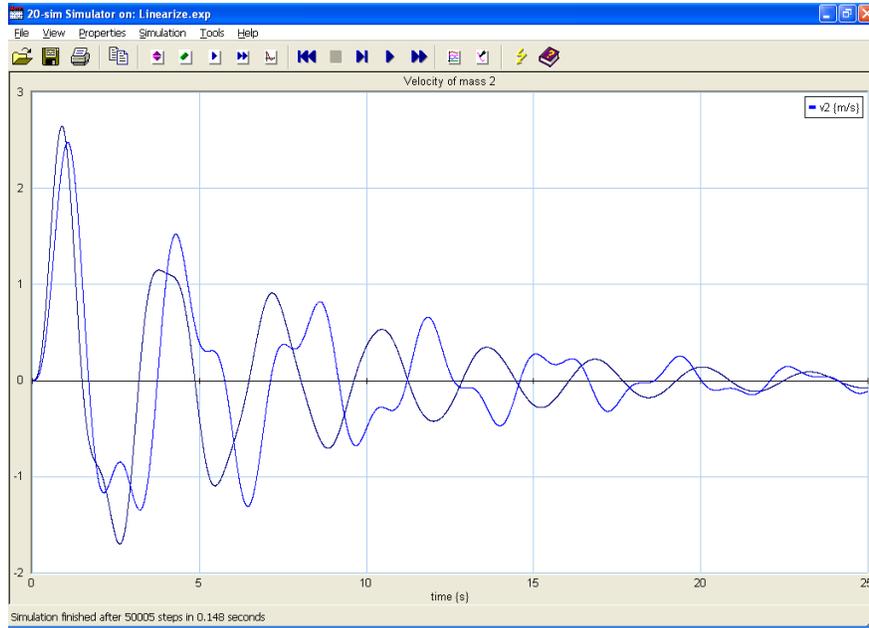


Figura 5.29. Resultados de la primera y segunda simulación.

Propiedades de Ejecución

9. En la barra de herramientas, pulse  o en el menú **Properties** seleccione el comando **Run**, aparecerá la ventana de la Figura 5.30:

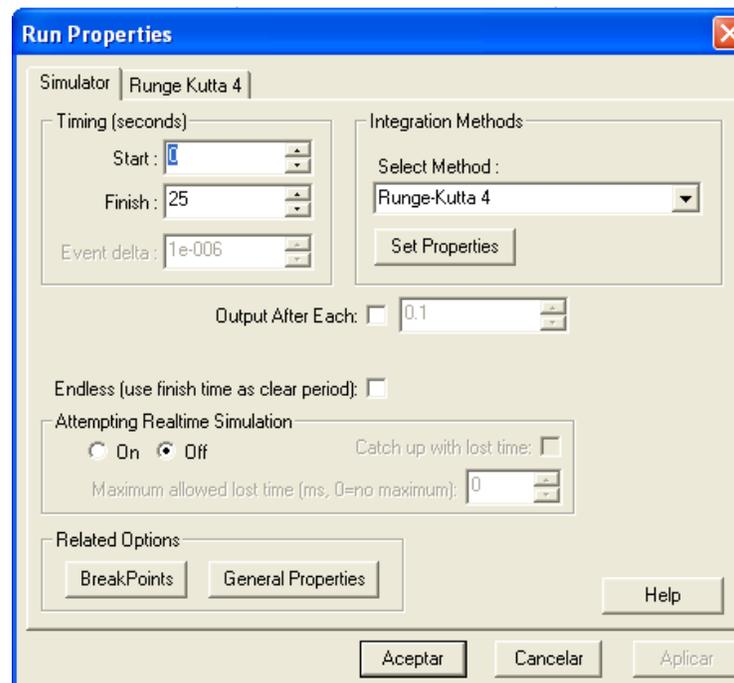


Figura 5.30. Propiedades de Ejecución.



En el editor de propiedades de ejecución se puede establecer el tiempo de inicio, tiempo de finalización y el método de integración (el método que se utiliza para simular el modelo, ver Apéndice A). Algunas opciones avanzadas también están disponibles si se necesita controlar la simulación de manera más estricta.

10. En la sección **Timing** cambie el parámetro **Finish** a 50.
11. Cierre el editor de parámetros pulsando el botón **OK**.
12. En la barra de herramientas, pulse  o desde el menú **Simulation** seleccione el comando **Run**.

Una tercera simulación se lleva a cabo y los resultados se señalan también en la gráfica. Su simulador ahora debe verse como la Figura 5.31.

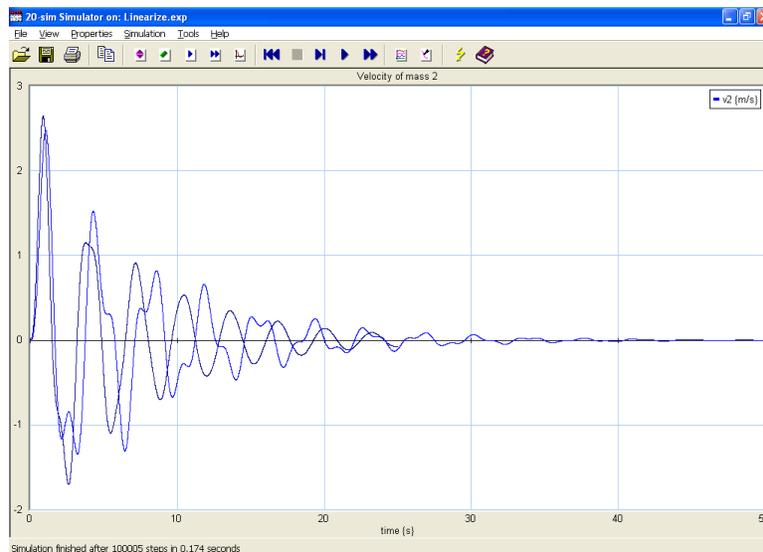


Figura 5.31. Los resultados de la simulación después de cambiar el tiempo **Finish** a 50.

Para evitar que las simulaciones se ejecuten una encima de la otra se borra la corrida anterior.

13. En el menú **Simulation** seleccione **Clear** y luego **Previous Run**.

Especificaciones de la Gráfica

14. En la barra de herramientas pulse el botón  ó seleccione el comando **Plot**

del menú *Properties*. El Editor de Propiedades de la Gráfica aparecerá en pantalla. Ver Figura 5.32.

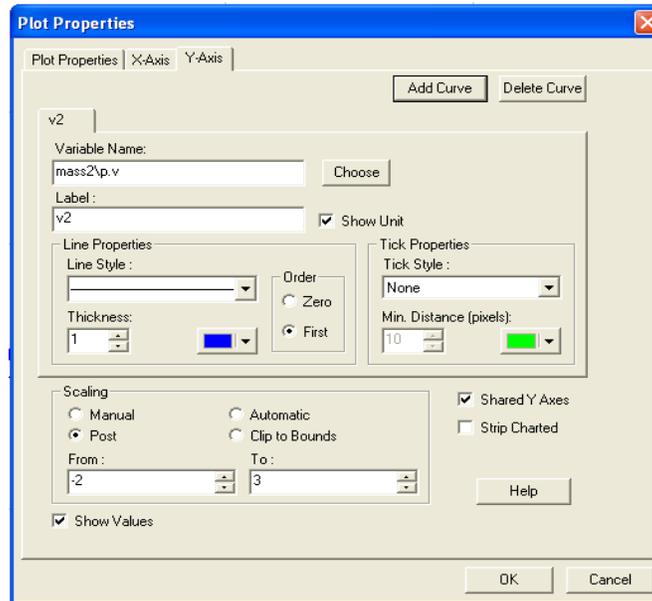


Figura 5.32. Editor de Propiedades de la Gráfica.

En este editor se pueden introducir los parámetros y elegir las configuraciones específicas como la escala de la gráfica, los estilos de línea, las etiquetas, colores del trazado, etc.

15. Seleccione la pestaña *Y-axis* y pulse el botón *Add Curve*. Esta acción abrirá el Seleccionador de Variables como se muestra en la siguiente Figura 5.33.

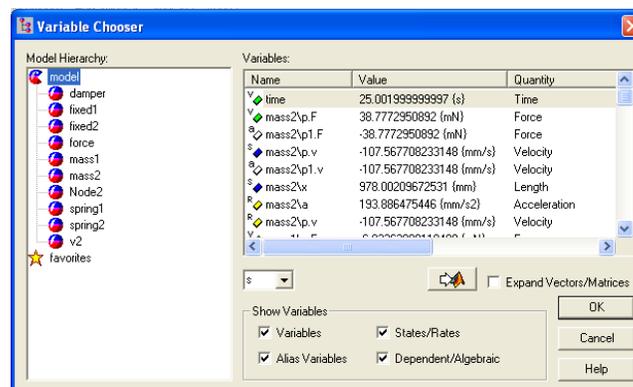


Figura 5.33. Seleccionador de Variables.

En esta ventana se puede elegir una variable y la unidad en que debería expresarse la variable seleccionada.

16. Seleccione la variable **mass1\p.v** de la lista (ver el Apéndice B para conocer el significado de la notación). Como se puede observar la unidad para esta variable es m/s.

17. Pulse el botón **OK** para cerrar el seleccionador de variables, seguidamente una segunda pestaña se agrega mostrando la variable **mass1\p.v**.

18. Cambie la etiqueta (Label) a v1. La ventana de Propiedades de la Gráfica debe parecerse a la presentada en la Figura 5.34:

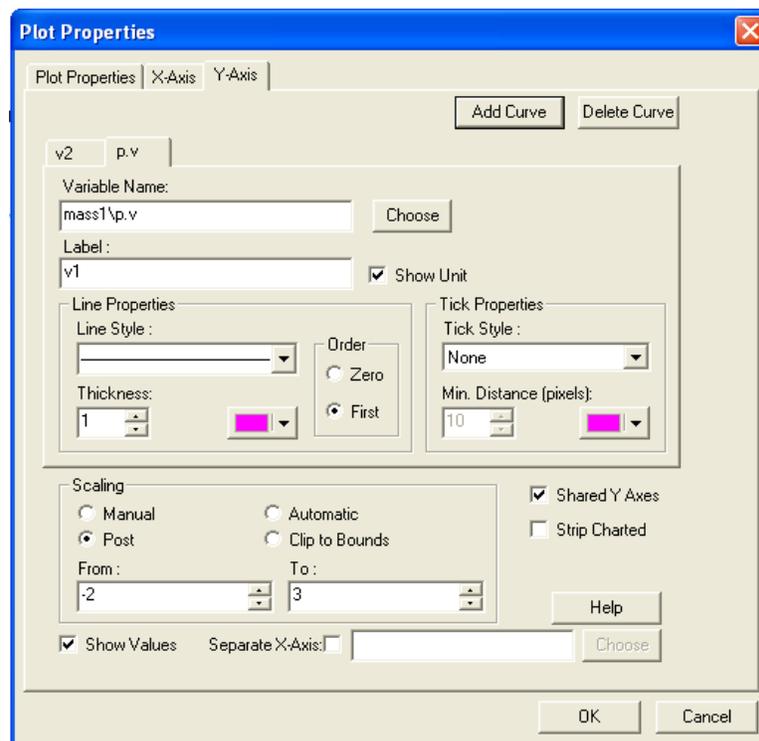


Figura 5.34. Editor de Propiedades de la Gráfica con una Variable

19. Cierre el Editor de Propiedades de la Gráfica pulsando el botón **OK**.

20. En la barra de herramientas, pulse el  botón o seleccione el comando **Run** del menú **Simulation**. De esta manera se realiza la simulación y los resultados se dibujan en la gráfica. El simulador debe mostrarse como en la siguiente ventana, Figura 5.35:

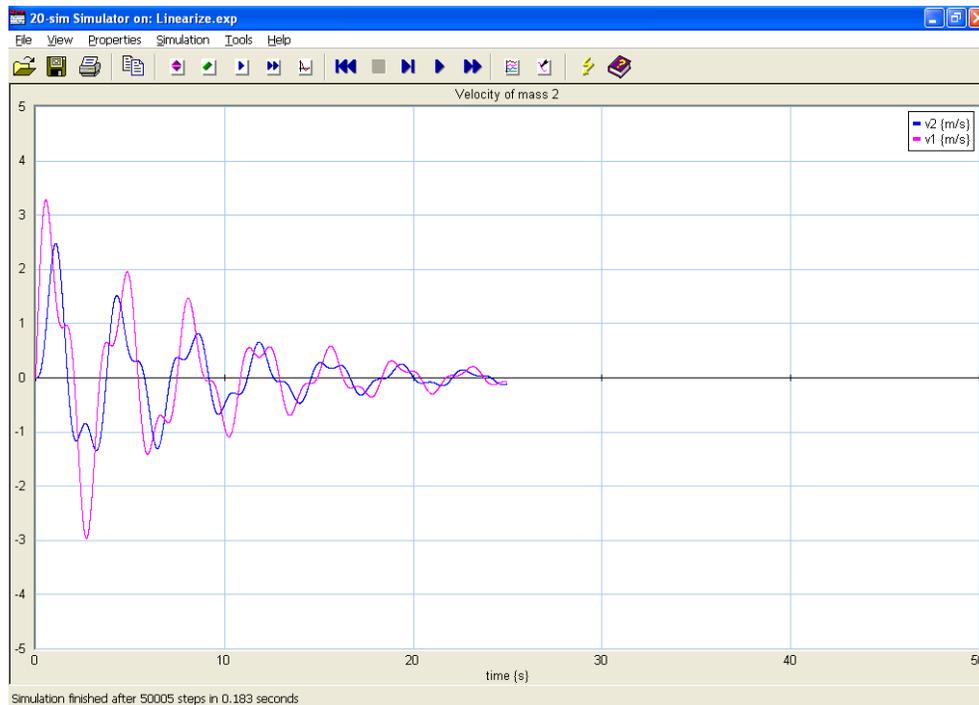


Figura 5.35. Resultados de la simulación después de adicionar una variable.

En el transcurso de los pasos anteriores, se obtuvo una breve descripción de los elementos básicos de un experimento en 20-sim. Si se cierra el simulador, aparecerá una ventana para guardar el experimento debido a que 20-sim ha detectado cambios. Como no se requieren guardar cambios ya que este es un experimento que se utiliza como ejemplo, entonces se procede a continuación:

21. Cierre el simulador y NO guarde el experimento.

5.8 MODELACIÓN Y SIMULACIÓN DE SISTEMAS FÍSICOS LINEALES APLICANDO LA TEORÍA DE BOND GRAPH CON EL SOFTWARE 20-SIM

Una vez revisado los ejemplos contenidos en la biblioteca de 20-sim, ahora se presentarán una serie de ejercicios de sistemas físicos lineales elementales en los cuales, al ir avanzando en el desarrollo de los ejercicios, también se incrementará su complejidad para mostrar al usuario las aplicaciones factibles y las ventajas del uso de la Teoría de Bond Graph y el software 20-sim para la modelación y simulación (estudio) de sistemas físicos y

el análisis de sus respuestas en el tiempo incluso antes de la construcción real de dicho elemento; además sirve como excelente herramienta para el estudio y análisis de diferentes sistemas físicos en el campo académico.

5.8.1 SISTEMA MASA-RESORTE-AMORTIGUADOR

En la Figura 5.36 se muestra el sistema Masa-Resorte-Amortiguador [19] y en la Figura 5.37 se tiene el grafo de enlace (Bond Graph) de dicho sistema.

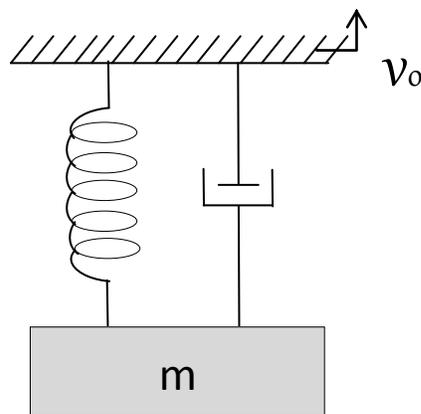


Figura 5.36. Diagrama Iconico Sistema Masa-Resorte-Amortiguador.

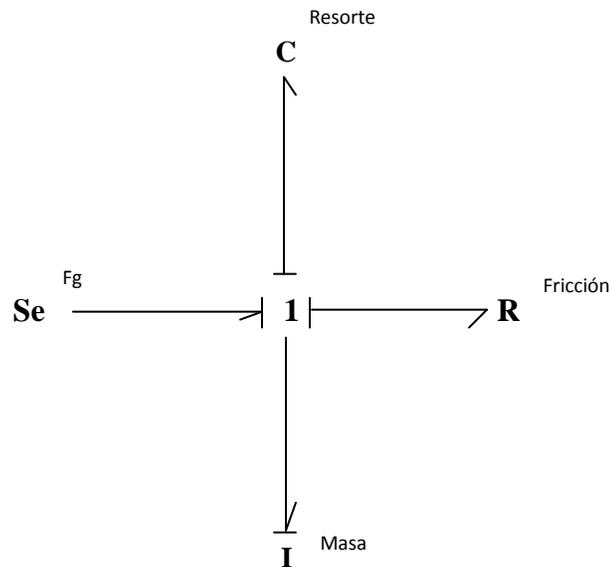


Figura 5.37. Modelo en Grafo de Enlace.



1. Abra 20-sim y seleccione el menú **File, New y Graphical Model**. Ver Figura 5.38.

Se abrirá el editor del 20-sim, asegúrese de que 20-sim esté en **Debug Mode** (Modo Depuración). Esto se indica en la barra de herramientas con el botón **Mode**  .

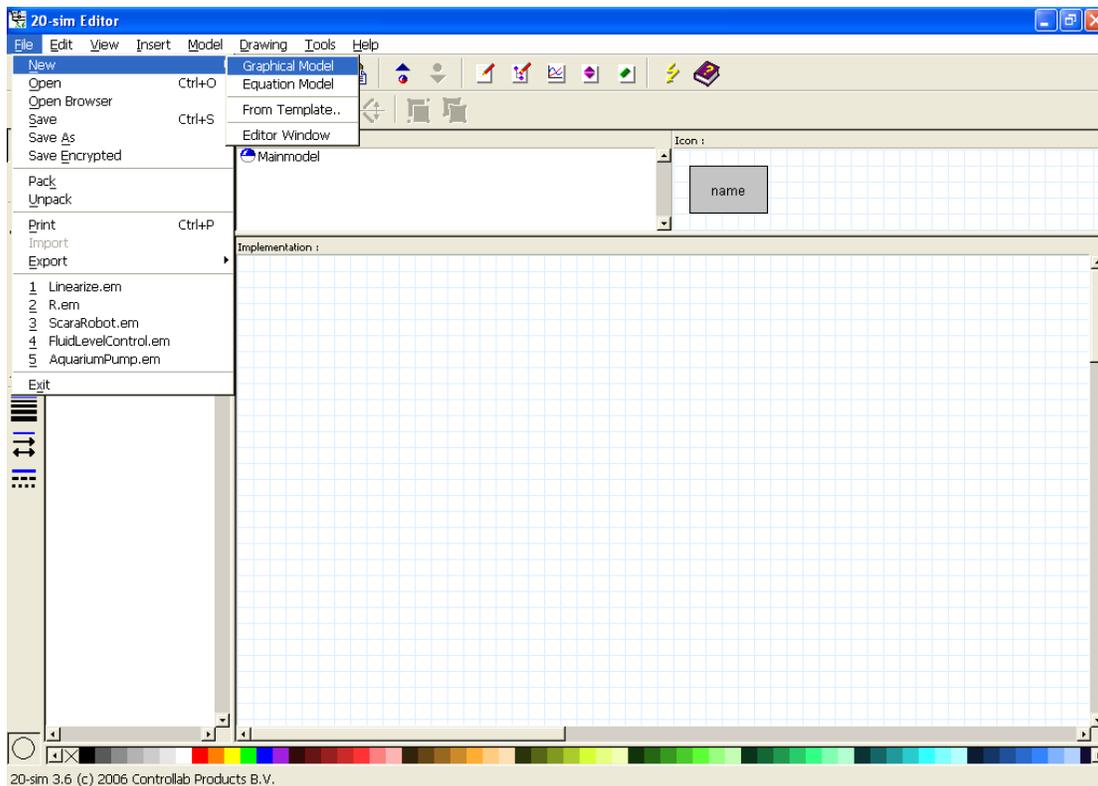


Figura 5.38. Nuevo modelo en el Editor.

2. Si el 20-sim esta en **Fast Mode** (Modo Rápido) indicado por este botón  , haga clic hasta que cambie al modo depuración  .
3. Posicione el puntero del mouse sobre cualquier parte de la ventana **Implementation**, desde el menú del botón derecho del ratón o desde el menú **Insert**, seleccione **Insert, Knot y OneJunction**. Esta acción añade la Unión 1. Ver Figura 5.39.

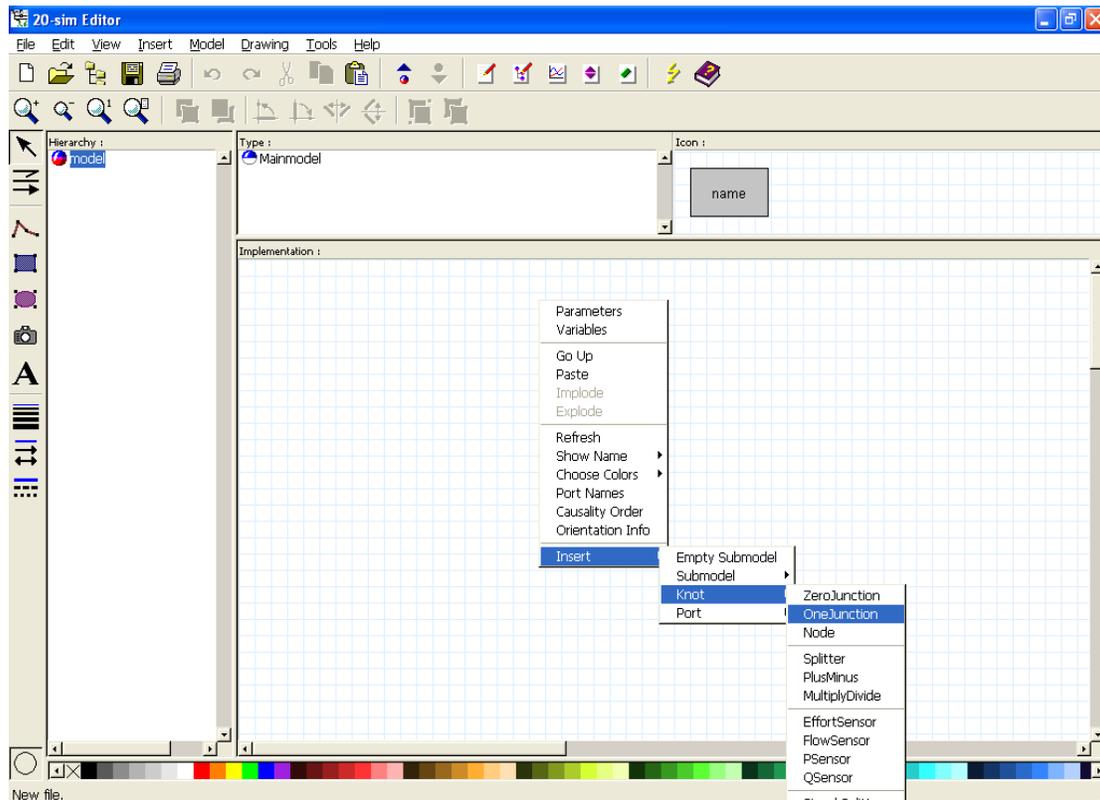


Figura 5.39. Insertando Unión 1.

- Abra el navegador de la biblioteca de modelos (si no está abierto) haciendo clic en el botón **Open Library**  en la barra de herramientas. El navegador mostrará la biblioteca de modelos de 20-sim.
- Arrastre los submodelos siguientes al Editor (ventana **Implementation**) de la carpeta **Bond Graph**. Ver Figura 5.40.

Carpeta de la Biblioteca	Modelo
Bond Graph	C.em
Bond Graph	I.em
Bond Graph	R.em
Bond Graph	Se.em

Tabla 5.1. Elementos para usar en el modelo.

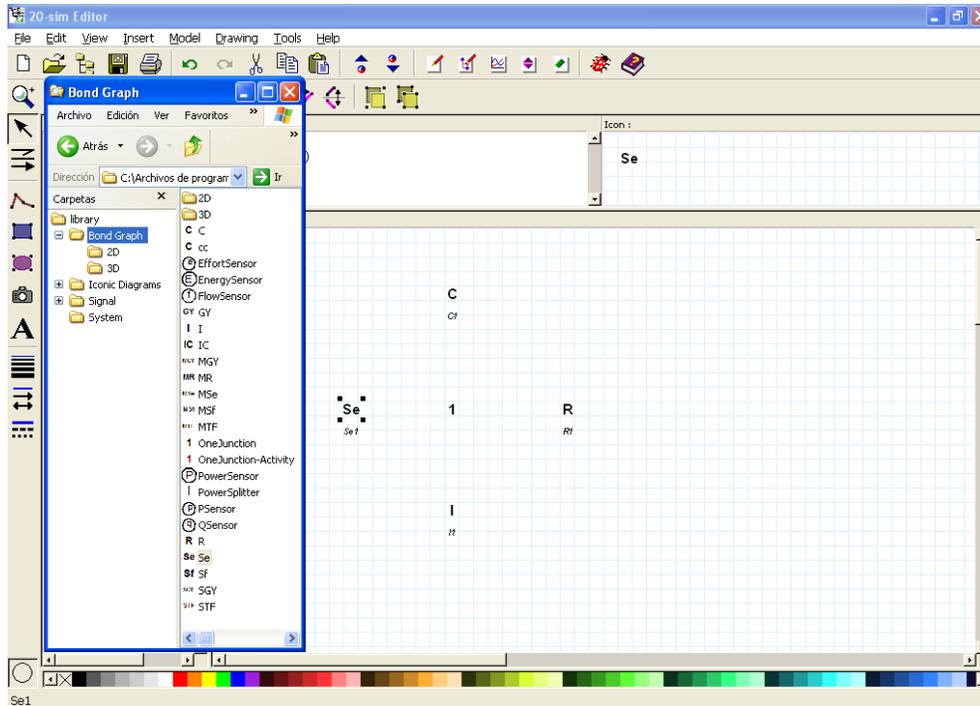


Figura 5.40. Insertando elementos de Bond Graph.

6. Seleccione el submodelo **Se**.
7. Desde el menú del botón derecho del ratón o desde el menú **Model**, seleccione el comando **Attributes** (se abrirá el Editor de Atributos). Ver Figura 5.41.

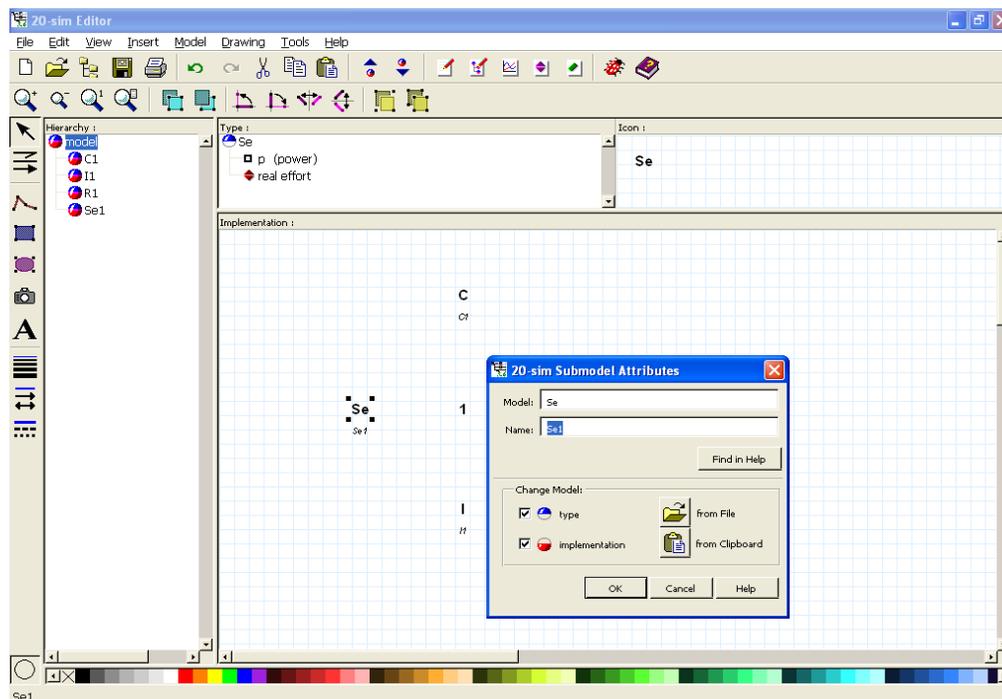


Figura 5.41. Editor de Atributos.



8. Cambie el nombre del modelo si así lo desea para que éste tenga el mismo que aparece en el grafo de enlace al inicio de este ejemplo. Escriba el nuevo nombre en el recuadro *Name*, luego pulse *OK*. Haga lo mismo con el resto de los submodelos. Ver Figura 5.42.

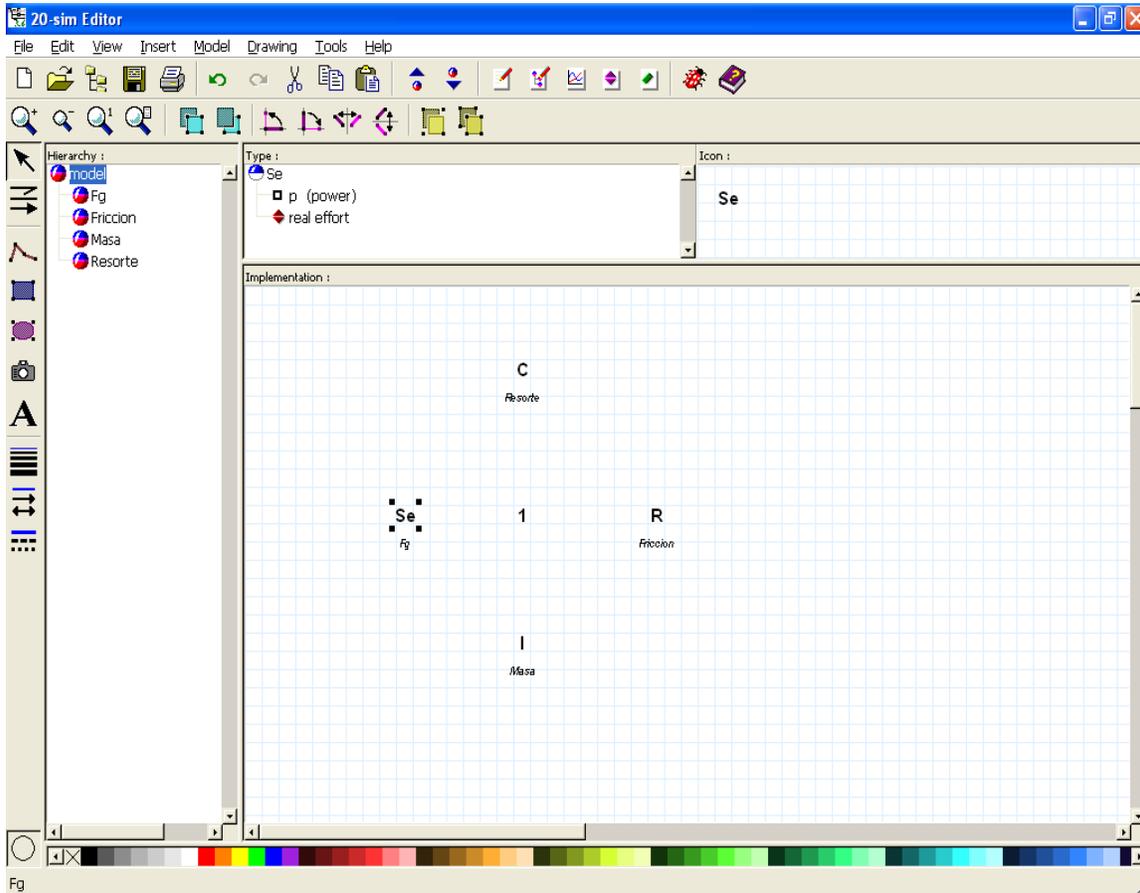


Figura 5.42. Edición del nombre de los elementos.

9. Seleccione el elemento **C** y pulse el botón derecho del ratón, seleccione el menú *Show Name* y el comando *Right*. Esta acción pondrá el nombre del elemento a la derecha del mismo. Ver Figura 5.43.

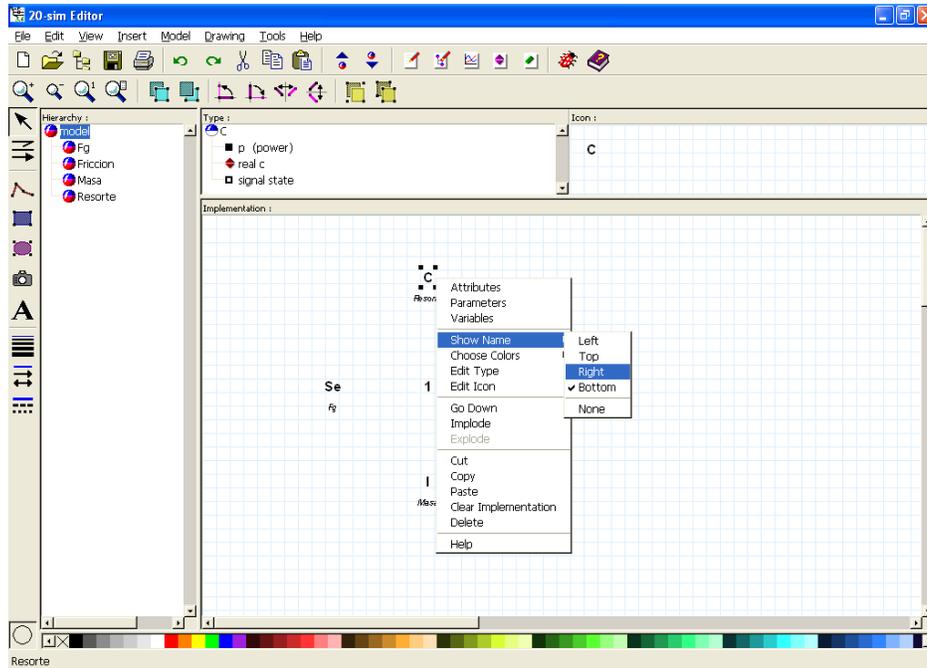


Figura 5.43. Comandos para reposicionar el nombre de los elementos.

10. Repita este procedimiento las veces que sea necesario hasta completar su modelo el cual debe lucir como la Figura 5.44.

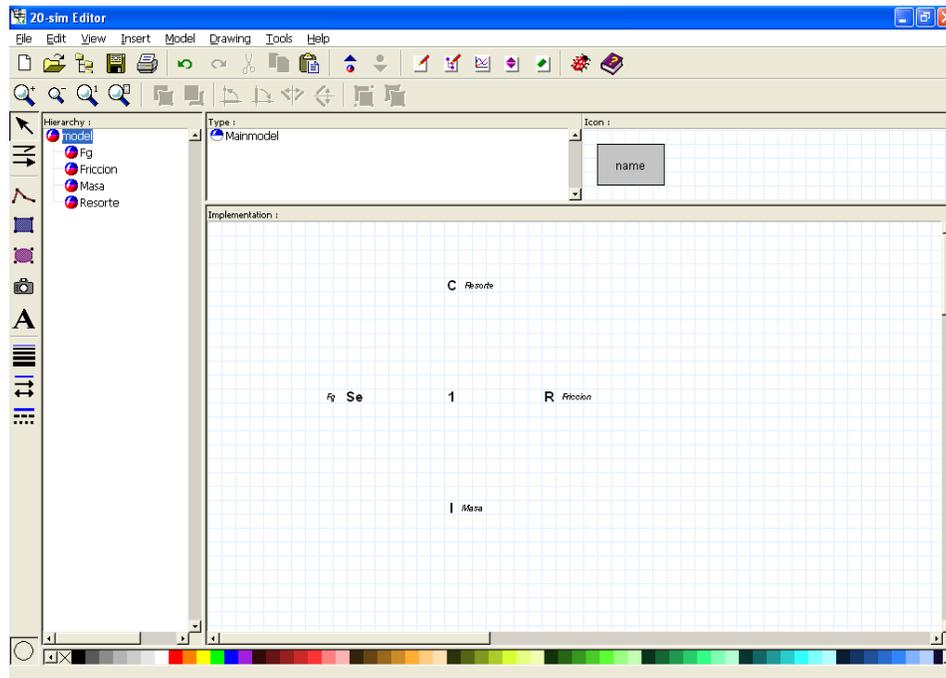


Figura 5.44. Reposicionando el nombre de los elementos.



Nota: El nombre del elemento **Se** (Fg) se le asignó a la izquierda para mejor visibilidad, para ello ejecute el paso 9 de la misma forma con excepción de la última orden, seleccione **Left**, en lugar de **Right**.

11. En la barra de herramientas lateral izquierda, pulse el botón  para activar el modo de conexión.
12. Seleccione el submodelo **Se** (clic sobre el elemento **Se**) y luego seleccione el elemento **1** (Unión 1) para crear el primer enlace. Ver Figura 5.45.

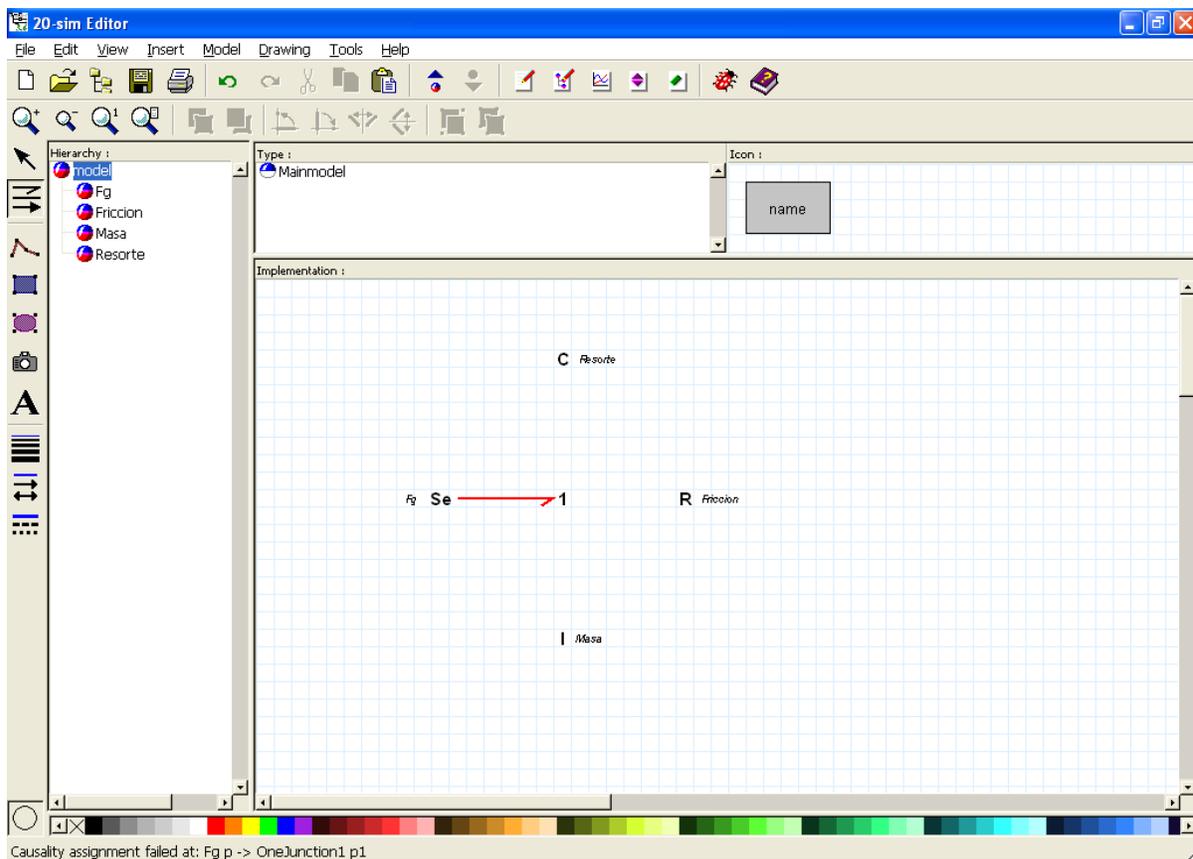


Figura 5.45. Creación del primer enlace.

Se muestra un enlace visible en el modelo, este enlace es de color rojo lo que significa que la causalidad no ha sido asignada, ya que 20-SIM tiene una asignación de causalidad en línea y nota que solo hay un enlace conectado a la Unión 1.

13. Dibuje un vínculo entre la Unión 1 y el submodelo C, ahora el modelo debe ser similar a la Figura 5.46.

El enlace del elemento C tiene una línea naranja lo que significa que el elemento C en esta configuración tiene una causalidad no predefinida.

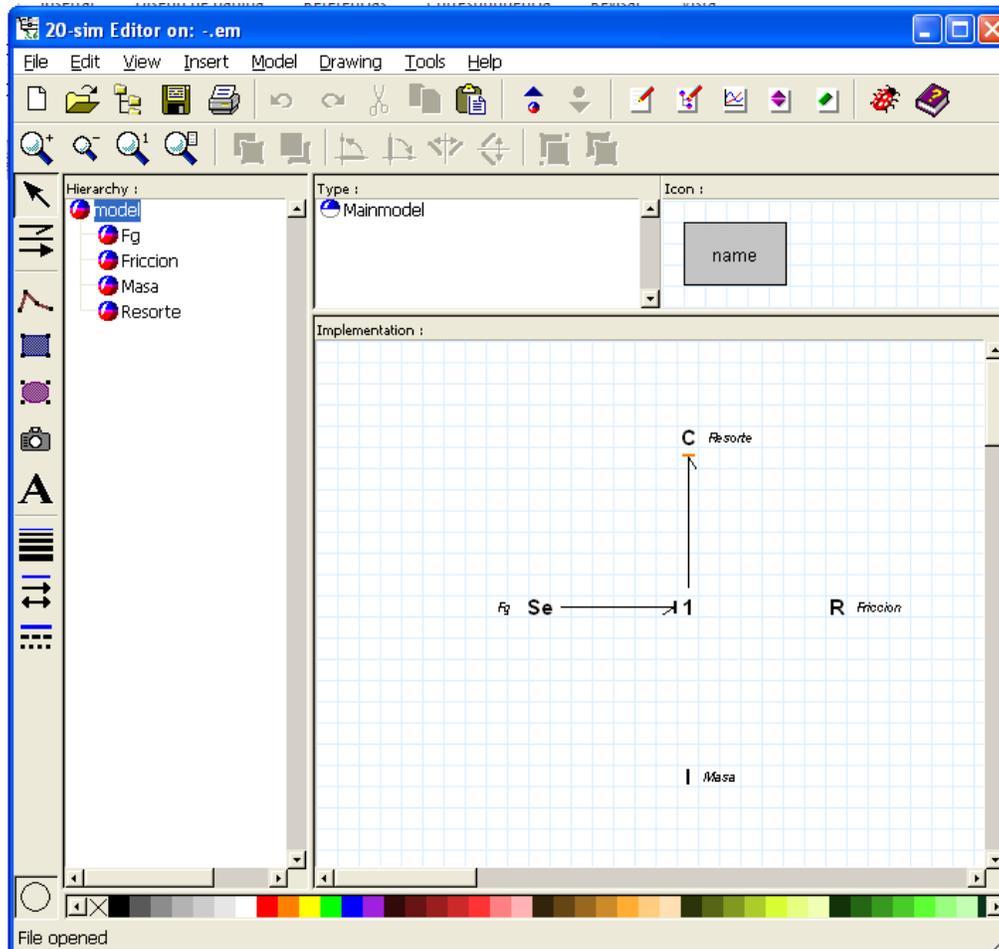


Figura 5.46. Creación del segundo enlace.

14. Introduzca el resto de conexiones hasta completar su modelo. Ver Figura 5.47.
15. En la barra de herramientas presiona el botón  o desde el menú en **Model** selecciona el comando **Check Complete Model** de comprobación del modelo completo.

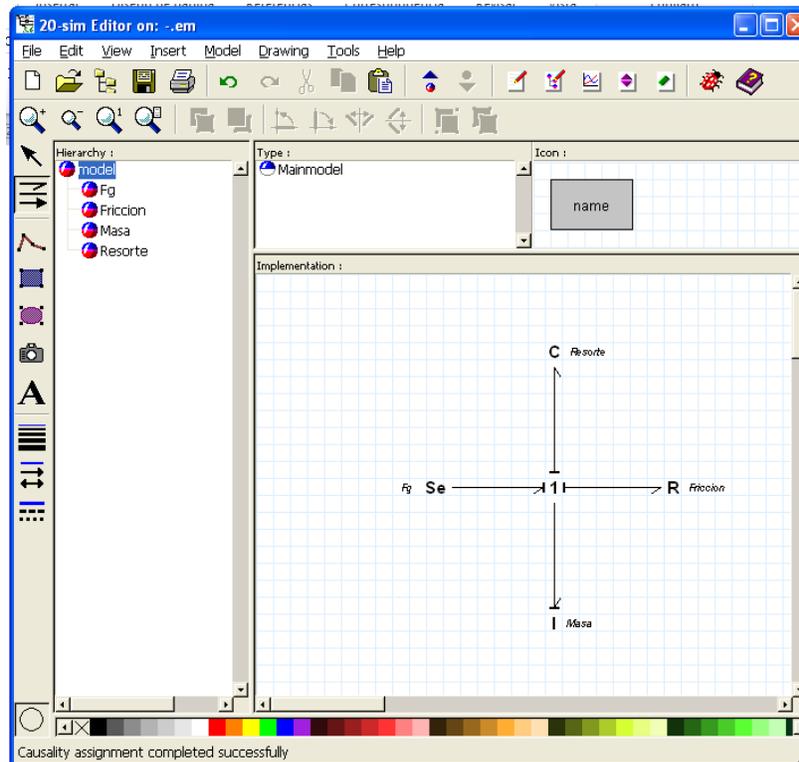


Figura 5.47. Modelo terminado.

Ahora el modelo completo será revisado. Si el modelo es correcto aparece una ventana de mensaje (sólo en modo depuración) que muestra el número de ecuaciones, variables, etc. (ver Figura 5.48). Si se encuentra algún error, una ventana de mensaje aparecerá, mostrando los errores que ha encontrado 20-SIM. Ver Figura 5.49.

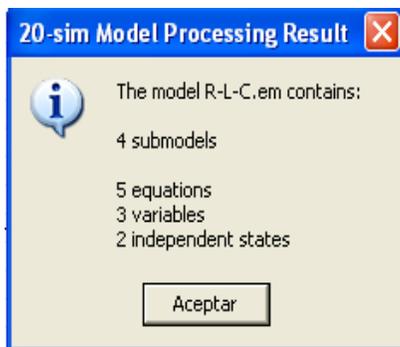


Figura 5.48. Ventana de Resultados del Chequeo de un Modelo Correcto.

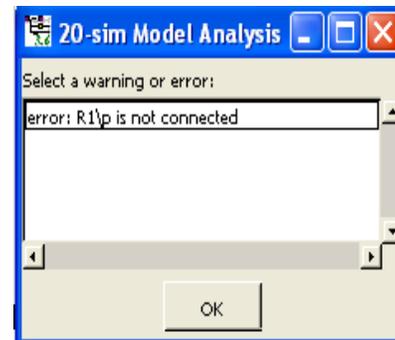


Figura 5.49. Ventana de Resultados del Chequeo de un Modelo Incorrecto.



- 16. Si se produce algún error debe corregirlo.
- 17. En el menú **File**, haga clic en **Save As**. Guarde el modelo en una carpeta temporal (por ejemplo c:/temp) con el nombre de MasaResorteAmortiguador.em

Si tiene problemas en introducir el bond graph o chequeando el modelo, entonces cargue el modelo **BondGraphDemo.em** de la ruta de acceso **Getting Started/Entering Bondgraph Models**. Luego de chequear y revisar exitosamente, se procederá con la simulación del modelo que se acaba de construir. El objetivo de esta simulación es verificar las respuestas en el tiempo de las variables involucradas en el ejercicio.

- 18. En la barra de herramientas del editor seleccione el botón  o el comando **Start Simulator** del menú **Model**. Se desplegara la ventana del simulador como en la Figura 5.50.

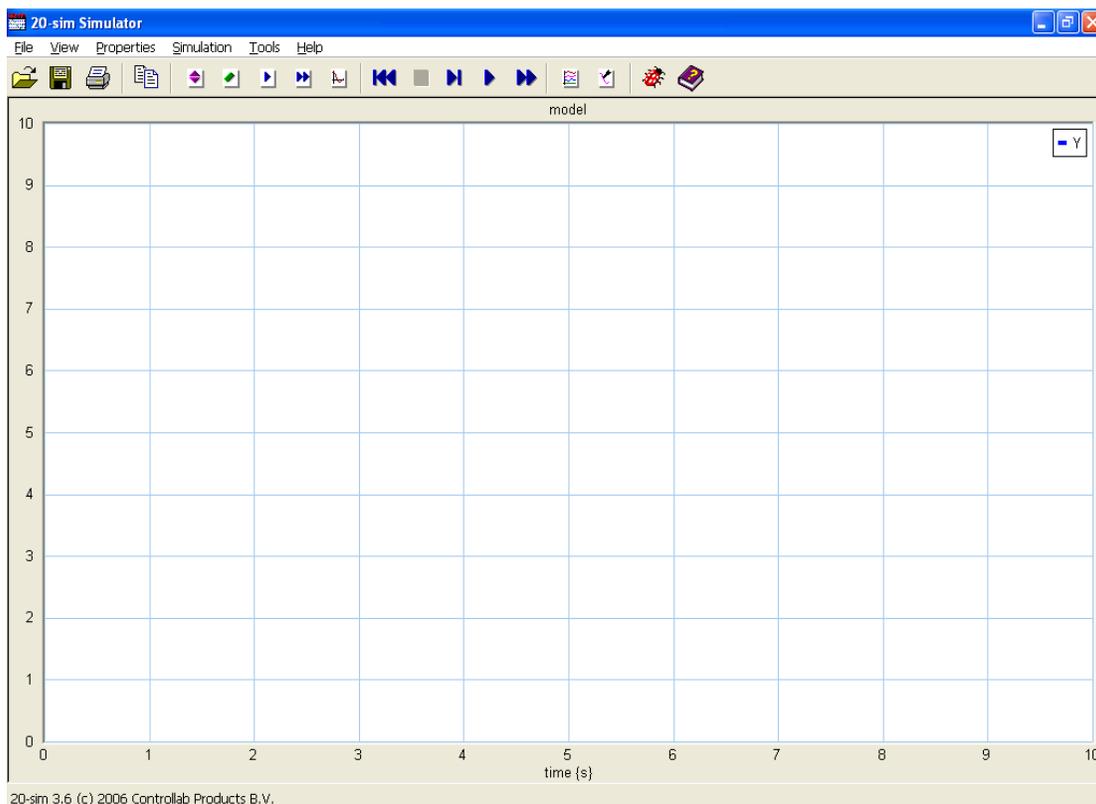


Figura 5.50. Ventana del simulador.



19. En la barra de herramientas del simulador, haga clic sobre el icono  o en el menú **Model**, seleccione el comando **Show Parameters** y cambie los valores de los parámetros por defecto del modelo como se indica a continuación en la Tabla 5.2, (ver Figura 5.51):

Elemento	Valor
Fricción/r	1
Fg/effort	-9.8
Masa/i	1
Resorte/c	0.5

Tabla 5.2. Valores de los elementos del modelo.

20. En la barra de herramientas, presione sobre el icono  o desde el menú **Properties** seleccione el comando **Run** y cambie los valores por defecto de los siguientes parámetros de simulación de la Tabla 5.3, (ver Figura 5.52):

Parámetro	Valor
Start	0
Finish	15
Method	Euler
Step Size	0.2

Tabla 5.3. Valores de los parámetros de simulación del modelo.

Nota: El valor del **Step Size** se modifica en la pestaña **Euler**, la cual es contigua a la pestaña **Simulator** de la ventana **Run Properties** (ver Figura 5.52).

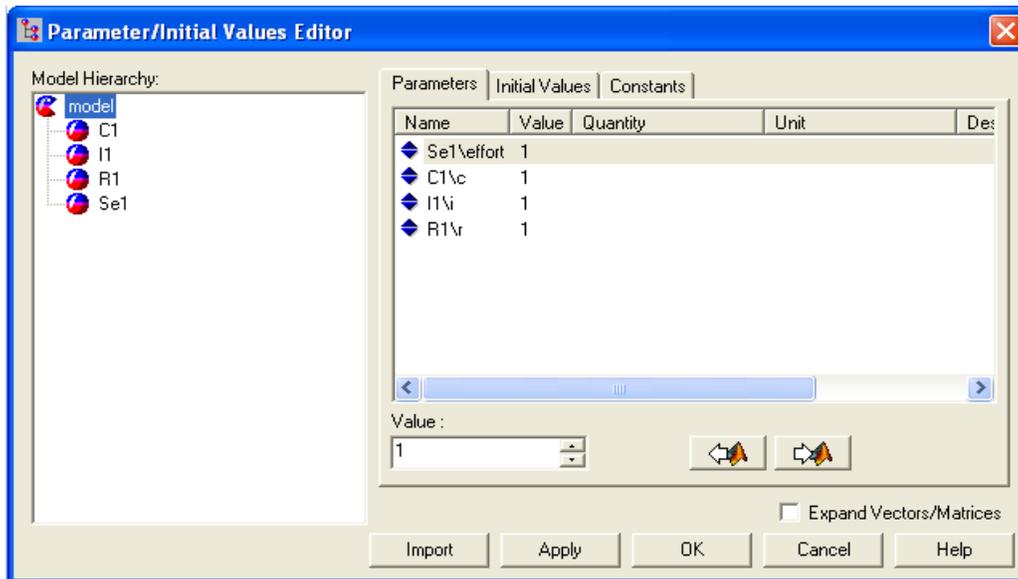


Figura 5.51. Valores de los elementos del modelo.

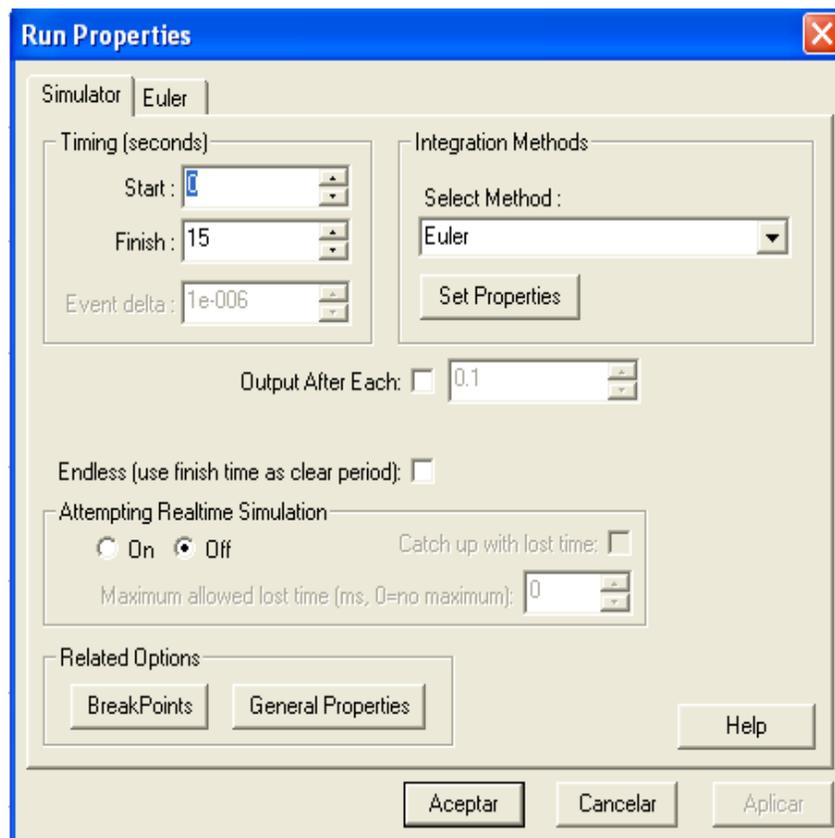


Figura 5.52. Edición de los valores de los parámetros de simulación del modelo.

21. Presione  o en el menú **Properties** seleccione el comando **Plot**.
22. En la nueva ventana, seleccione la pestaña **Plot Properties** para cambiar el título por defecto del modelo y escriba el siguiente: “Modelo Bondgraph” o el de su preferencia. Observe la Figura 5.53.

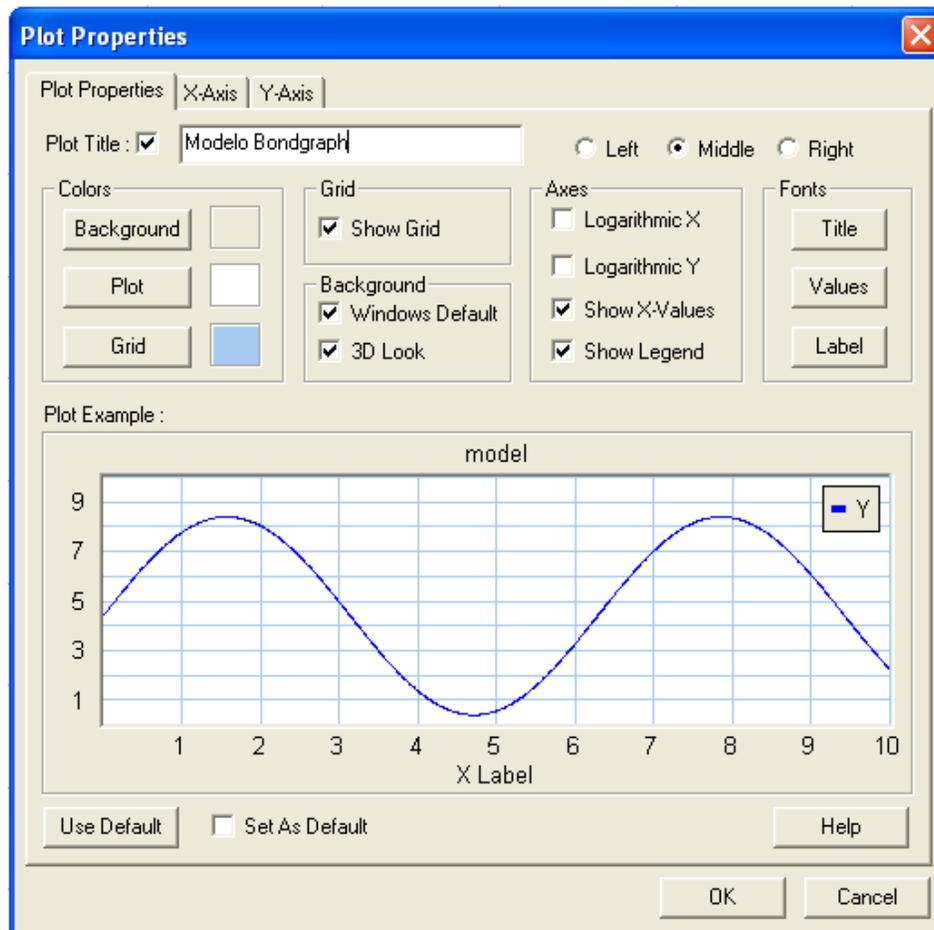


Figura 5.53. Edición del nombre de la gráfica.

23. Ahora seleccione la pestaña **Y-axis** y haga clic en **Choose** para elegir la variable a simular.
24. Seleccione la variable **Masa\p.f** de la lista, la cual representa la velocidad de la masa, y luego haga clic en **OK**. Ésta será la variable mostrada en la simulación. Ver figura 5.54.

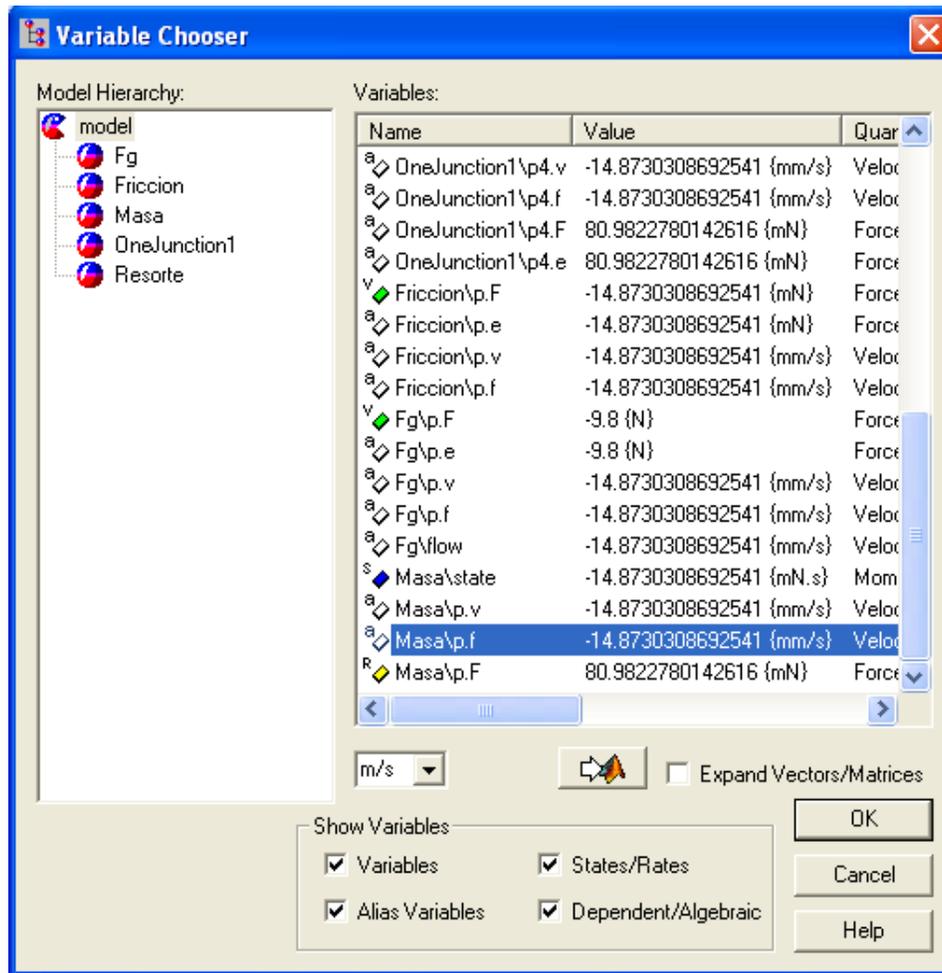


Figura 5.54. Elección de la variable a simular.

25. En la misma ventana, cambie la etiqueta *label* por “Velocidad de la Masa”.
26. Establezca los siguientes parámetros en las diferentes secciones de la misma pestaña *Y-axis* (ver Figura 5.55), sírvase de la Tabla 5.4:

Tick Properties		Scaling	
Tick Style	Diamond Closed	Scaling	Manual
Min. Distance (pixels)	2	From	-6
Color	Rojo	To	4

Tabla 5.4. Parámetros de la edición de apariencia de la simulación.

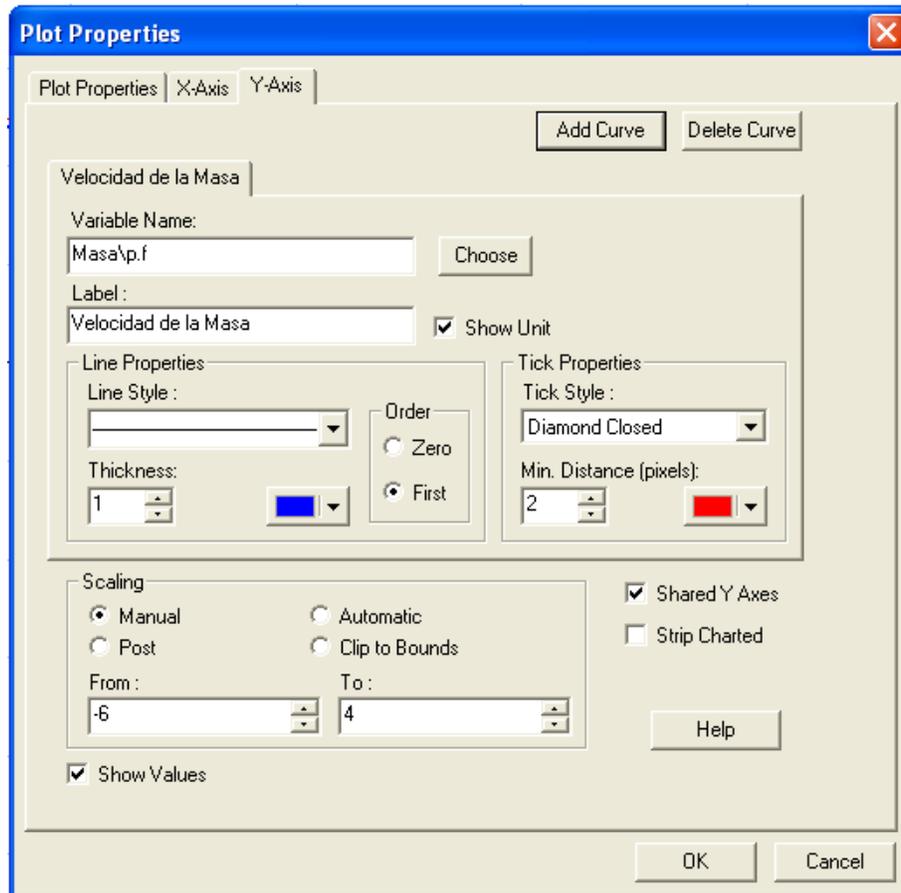


Figura 5.55. Edición de apariencia de la simulación.

27. Cierre el editor *Plot Properties* haciendo clic en **OK**.
28. En la barra de herramientas, haga clic sobre  o en el menú *Simulation* seleccione el comando **Run** para iniciar la simulación. El resultado de la simulación debe ser similar a la Figura 5.56.
29. Para guardar haga clic sobre el icono  o en el menú *File* y luego seleccione **Save**. Guarde el experimento (extensión .exp) usando el mismo nombre del modelo y en la misma carpeta donde se almacenó el modelo (archivo con extensión .em).

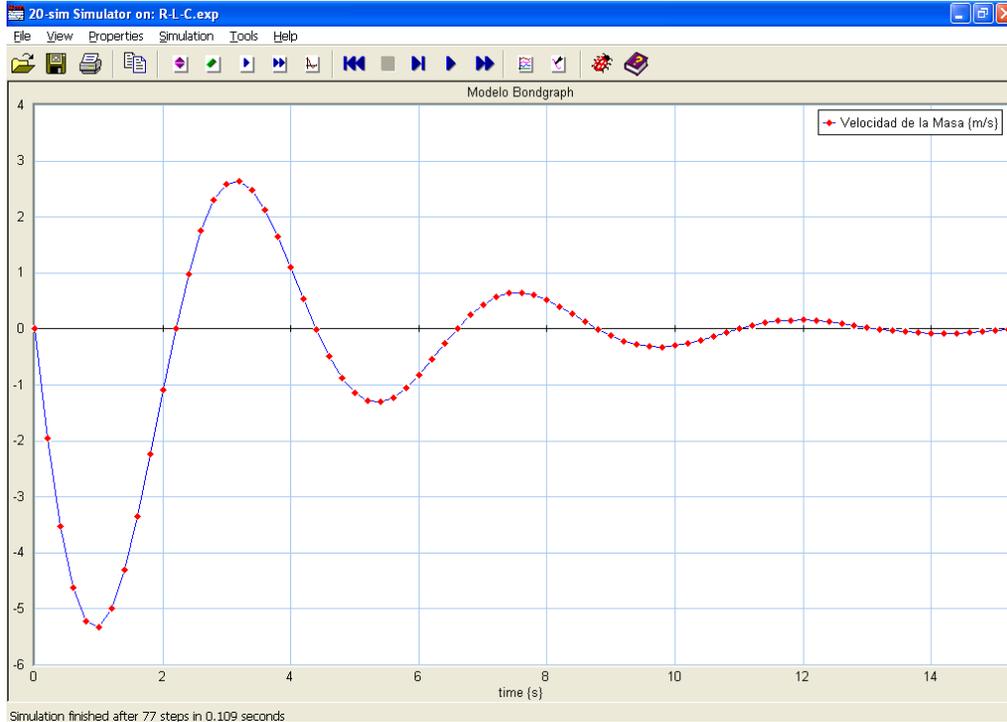


Figura 5.56. Resultado de la simulación del modelo Masa-Resorte-Amortiguador usando el Método de Euler con paso de avance igual a 0.2 en 20-sim.

La simulación del ejercicio anterior se plantea con el Método Euler, aunque la resolución de este método se asemeja a la solución analítica de las ecuaciones que describen al sistema anterior, el error que se conoce teóricamente y lleva consigo este método es apreciable, como se explica en el Apéndice A, y por lo tanto hace la resolución de este método inexacta. Por esta causa es necesario elegir un método de integración numérica que provea una solución más aproximada a la solución analítica (real) del ejercicio.

Para resaltar la diferencia entre los métodos de integración que posee el software 20-sim, observar el error apreciable del método de Euler y obtener una mejor aproximación de la solución, se simula el mismo ejercicio anterior pero usando el Método Runge-Kutta 4 (ver Apéndice A), el cual es el método que se adopta de este punto en adelante para la simulación de los ejercicios que se plantean. Por lo tanto, se continúa con la modificación del método de integración y el paso de avance como se ilustra en el paso 20 del ejercicio anterior:

20. En la barra de herramientas, presione sobre el icono  o desde el menú *Properties* seleccione el comando *Run* y cambie los valores por defecto de los siguientes parámetros de simulación de la Tabla 5.5, (ver Figura 5.57):

Parámetro	Valor
Start	0
Finish	15
Method	Runge-Kutta 4
Step Size	0.1

Tabla 5.5. Valores de los parámetros de simulación del modelo.

Nota: El valor del *Step Size* se modifica en la pestaña *Runge-Kutta 4*, la cual es contigua a la pestaña *Simulator* de la ventana *Run Properties* (ver Figura 5.57).

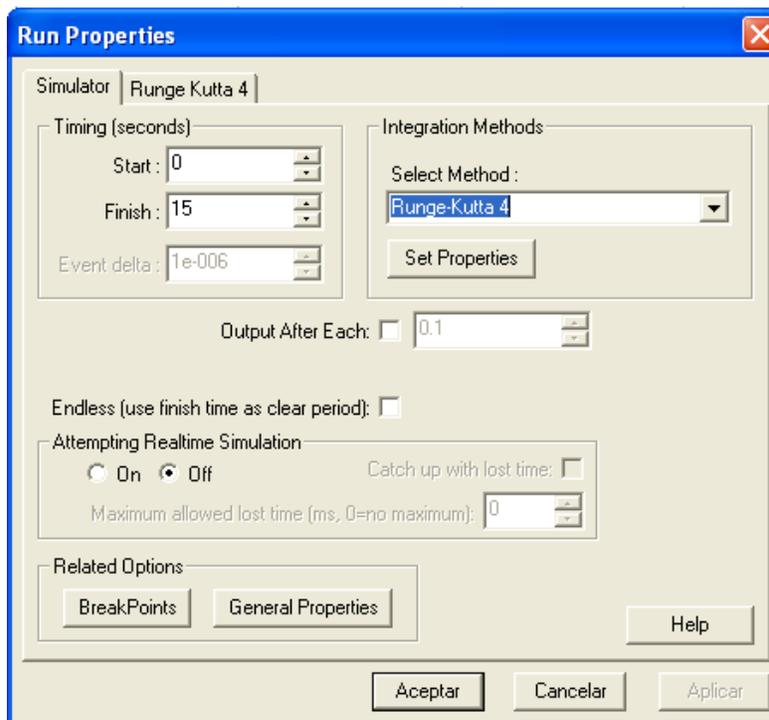


Figura 5.57. Edición de los valores de los parámetros de simulación del modelo.



Luego de hacer los cambios de los parámetros de ejecución de la simulación, se salta hasta el paso 28 del ejercicio anterior, en la cual se lleva a cabo la simulación.

Nota: Antes de ejecutar el siguiente Paso 28, para evitar que la gráfica anterior (resultado de la aplicación del método de Euler) se superponga sobre el nuevo resultado, es indispensable realizar la siguiente acción: en el menú *Simulation*, seleccione el comando *Clear*, y luego seleccione *All Runs*.

28. En la barra de herramientas, haga clic sobre  o en el menú *Simulation* seleccione el comando *Run* para iniciar la simulación. El resultado de la simulación debe ser similar a la Figura 5.58.

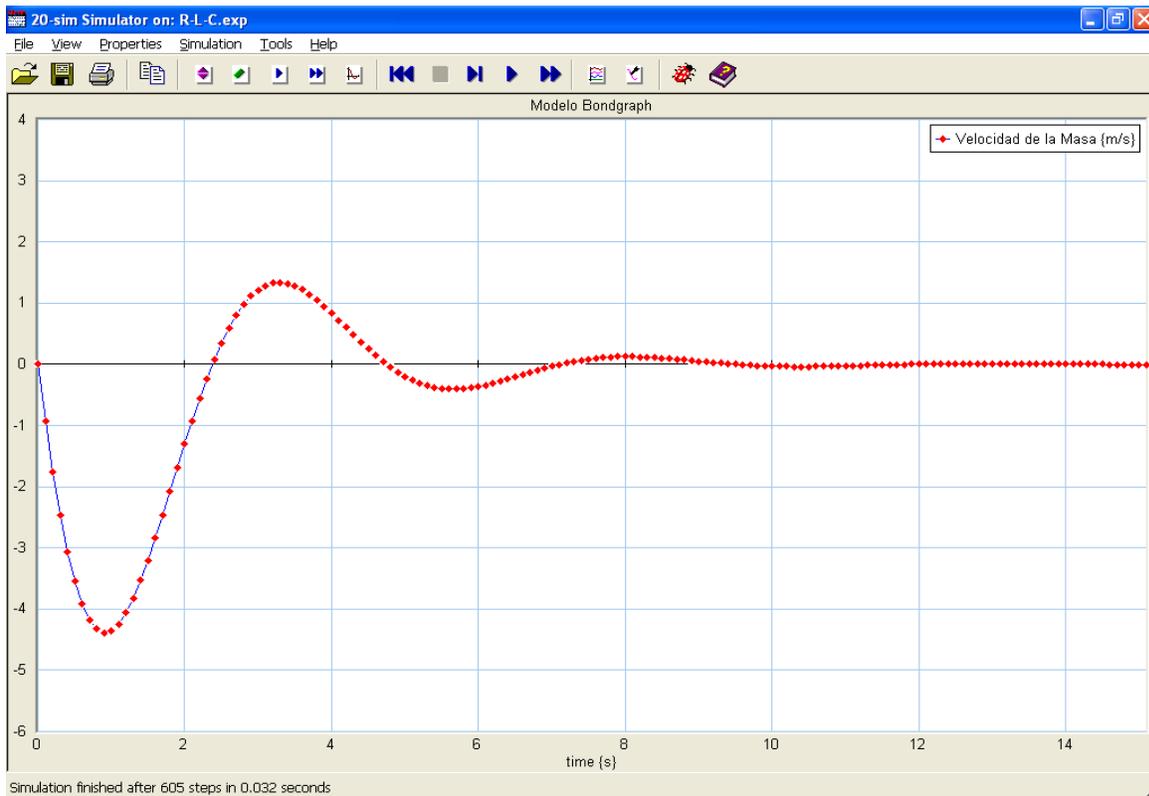


Figura 5.58. Resultado de la simulación del modelo Masa-Resorte-Amortiguador usando el Método de Runge-Kutta 4 con paso de avance igual a 0.1 en 20-sim.

5.8.2 CIRCUITO RLC

En la Figura 5.59 se presenta el circuito eléctrico Resistencia-Inductor-Capacitor (RLC) y en la Figura 5.60 se tiene el grafo de enlace (bond graph) de dicho sistema [10]. Se puede notar que el grafo de enlace es idéntico al del sistema Masa-Resorte-Amortiguador.

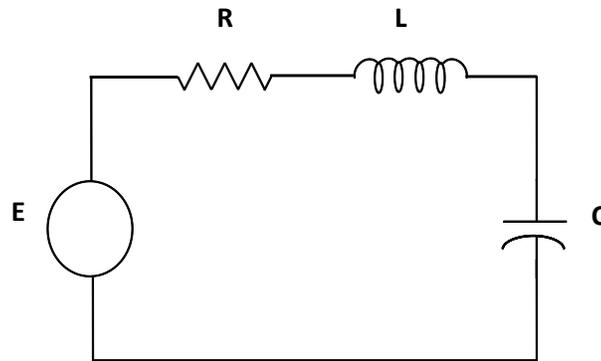


Figura 5.59. Circuito RLC.

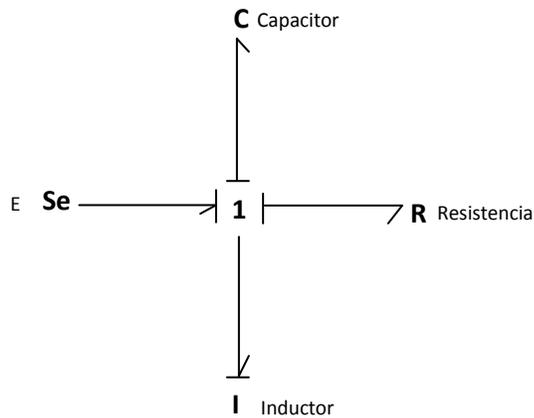


Figura 5.60. Circuito RLC.

1. Abra 20-SIM y en el menú **File** seleccione **New** y luego **Graphical Model**.

Se abra el editor del 20-SIM, asegúrese de que 20-SIM este en modo depuración. Esto se indica en la barra de herramientas con el botón **Mode** . (Ver Figura 5.61).

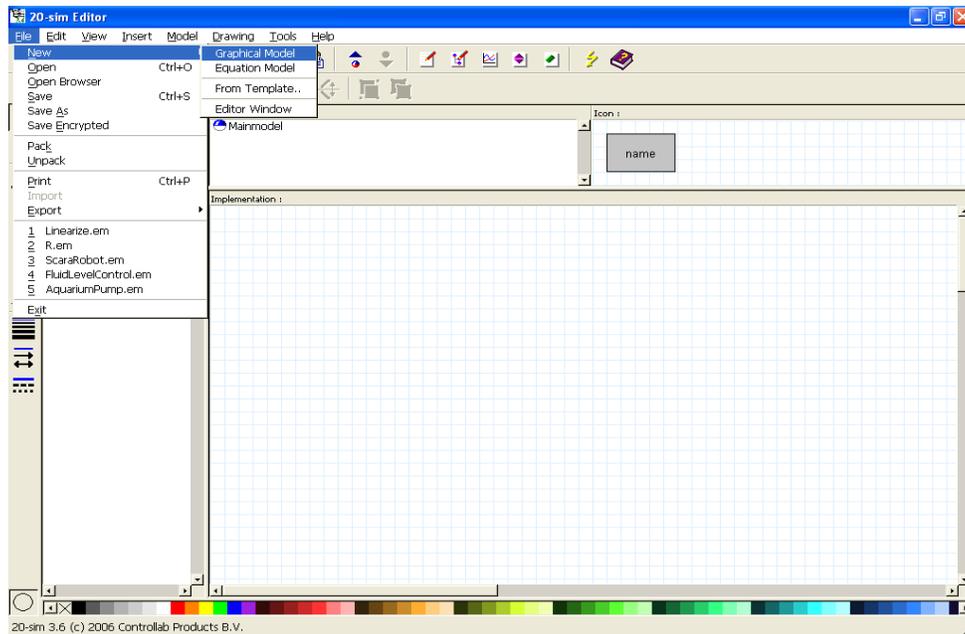


Figura 5.61. Selección del espacio de trabajo.

2. Si el 20-SIM está en modo rápido indicado por este botón , haga clic hasta que cambie al modo depuración .
3. En el menú del botón derecho del ratón o desde el menú *Insert*, seleccione *Knot* y después *OneJunction*. Se añade la unión 1, (ver Figuras 5.62 y 5.63).

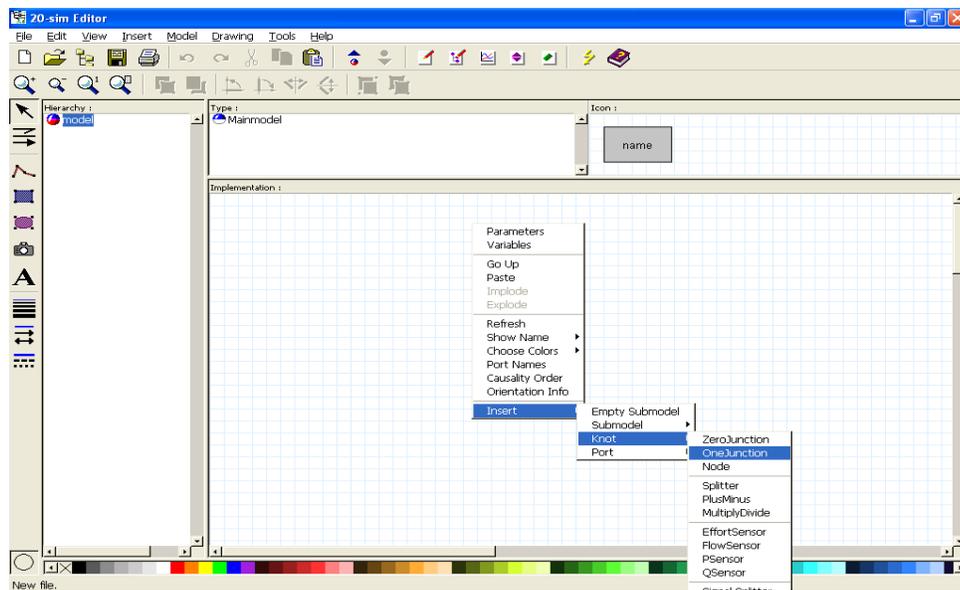


Figura 5.62. Inserción de la Unión 1.

4. Revise si al abrir 20-sim se ha desplegado el navegador de la biblioteca en una ventana independiente de la del editor de 20-sim, en caso contrario proceda a pulsar el siguiente botón  en la barra de herramientas.

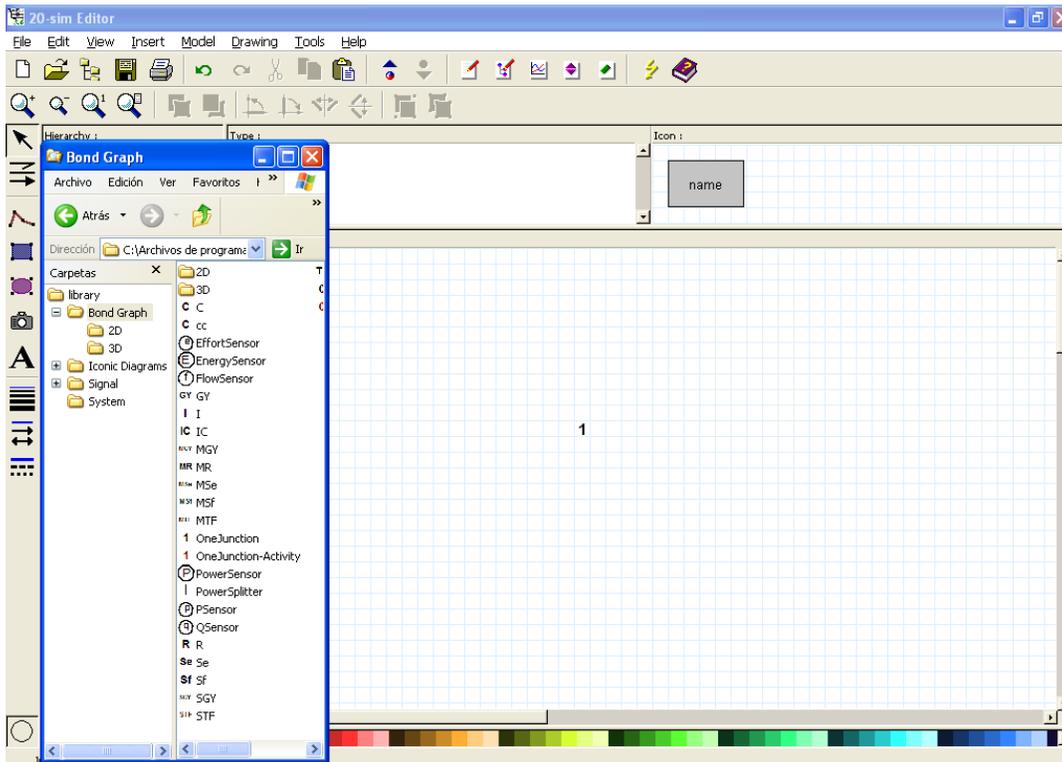


Figura 5.63. Navegador de la biblioteca de 20-sim.

5. En el navegador seleccione la carpeta **Bond Graph** y luego arrastre los elementos siguientes de la librería de Bond Graph al editor. (Ver Tabla 5.6).

Carpeta de la Biblioteca	Modelo
Bond Graph	C.em
Bond Graph	I.em
Bond Graph	R.em
Bond Graph	Se.em

Tabla 5.6. Elementos para usar en el modelo.

6. Seleccione el submodelo **Se.em** y arrástrelo hacia la ventana de implementación.
(Ver Figura 5.64).

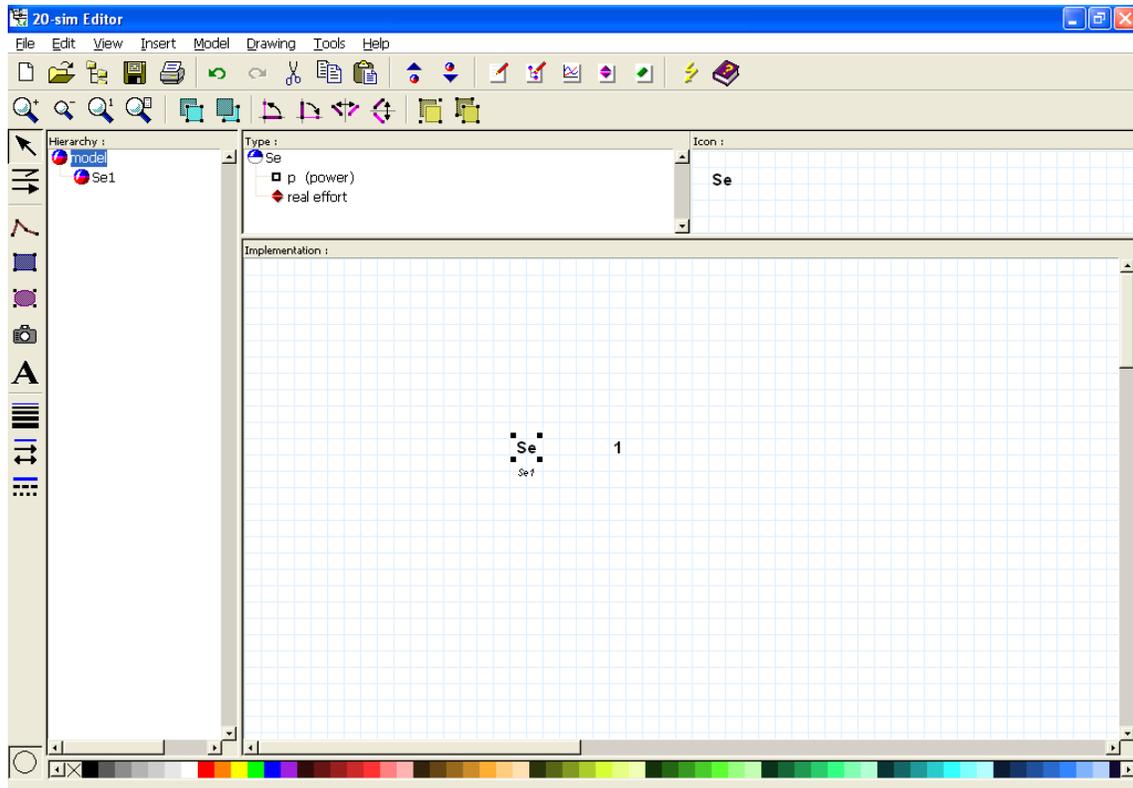


Figura 5.64. Inserción del elemento Se.

7. Desde el botón derecho del ratón o desde el menú modelo, seleccione el comando de atributos (se abrirá el editor de atributos).
8. Cambie el nombre del modelo si así lo desea. En este caso se mantiene el asignado por defecto. (ver Figura 5.65).
9. Seleccione cada uno de los elementos que conforman su modelo desde la librería de Bond Graph hasta completar su grafo de enlace.
10. Pulsando el botón derecho del ratón sobre cualquier elemento y seleccionando el menú **Show Name** se puede colocar el nombre del elemento en la posición que desee. Ver figura 5.66.

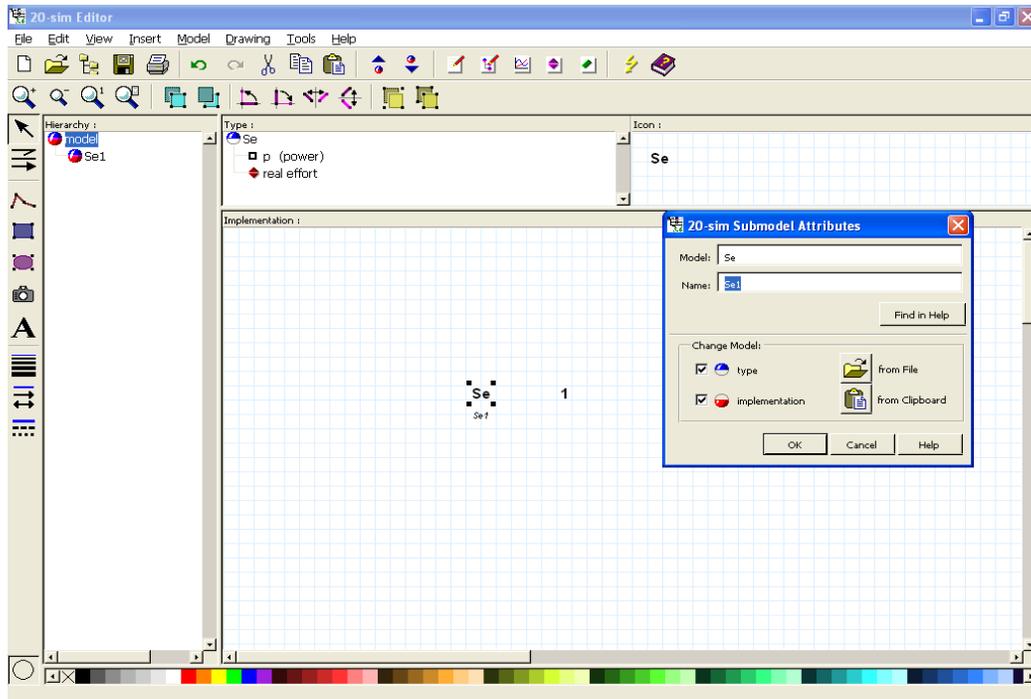
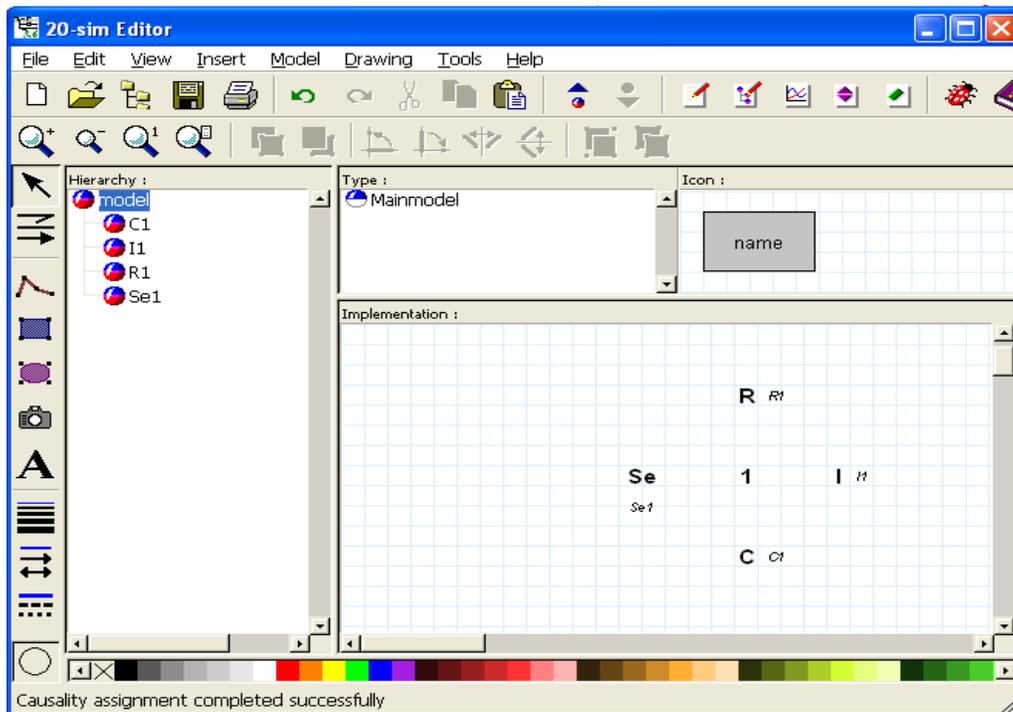
Figura 5.65. Edición del nombre del modelo **Se.em**.

Figura 5.66. Edición de la posición del nombre del elemento.

11. En la barra de herramientas haga clic en el botón para colocar el modo de conexión.
12. Seleccione el elemento **Se** (clic con el ratón en el botón izquierdo sobre el elemento) y luego clic sobre la unión 1 para crear el primer enlace. (Ver Figura 5.67).

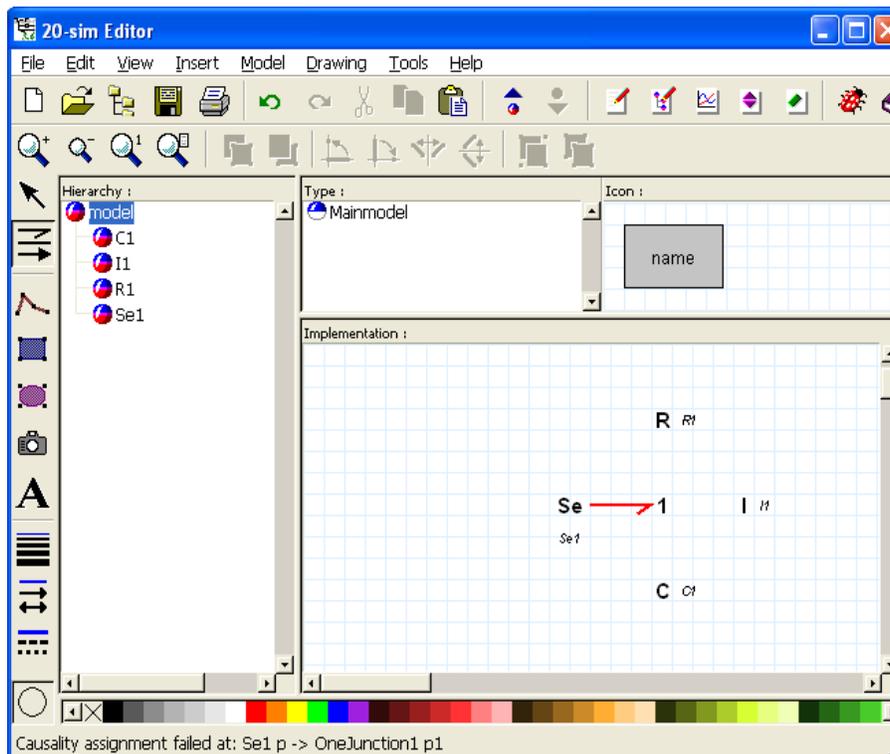


Figura 5.67. Realización del primer enlace.

Ahora un enlace es visible en nuestro modelo, este es de color rojo lo que significa que la causalidad no ha sido asignada.

13. Proceda a crear un vínculo entre la unión 1 y el submodelo **C**, ahora el modelo debe verse como en la Figura 5.68.

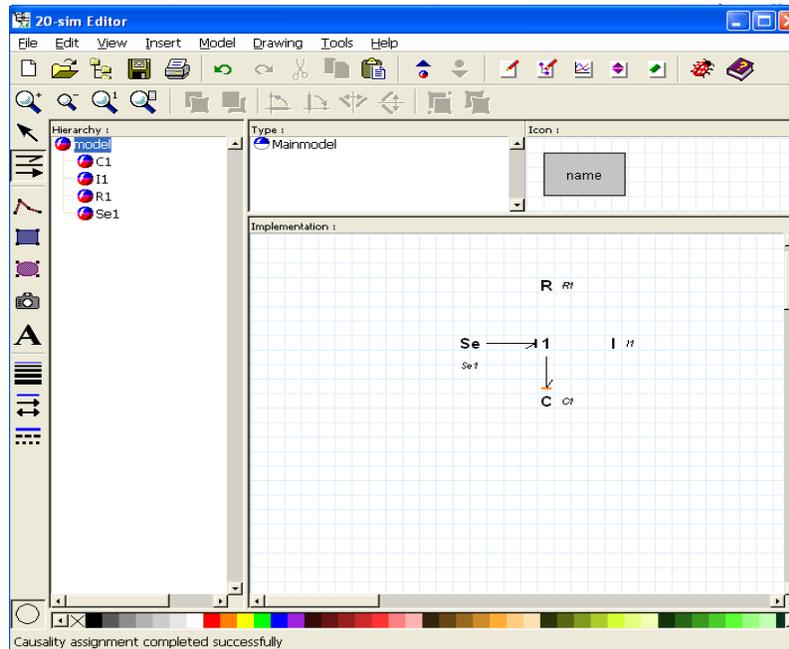


Figura 5.68. Realización del segundo enlace.

14. Introduzca el resto de las conexiones hasta completar el modelo. (Ver Figura 5.69).

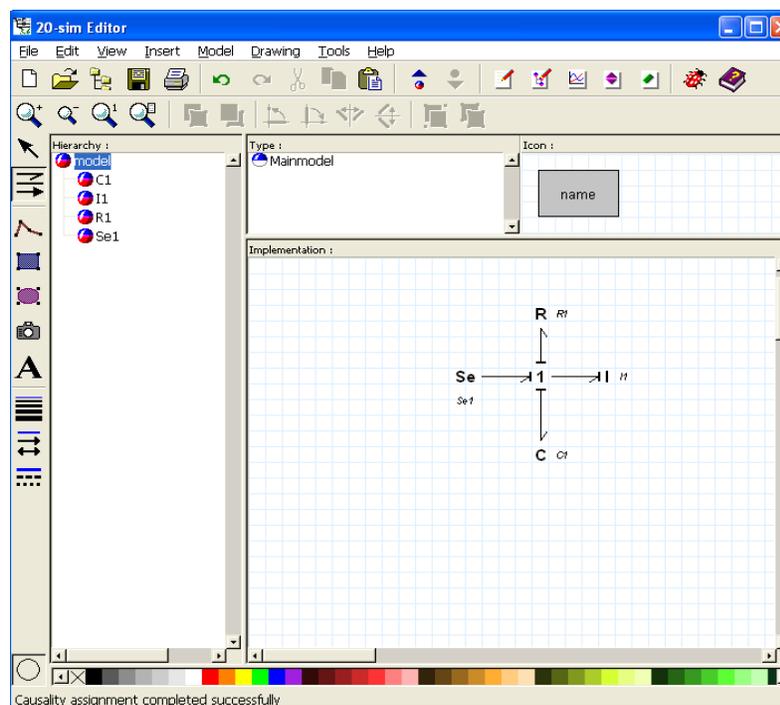


Figura 5.69. Realización del resto de los enlaces.

15. En la barra de herramientas haga clic sobre el icono  para la comprobación del modelo. Si el modelo está correcto aparecerá una ventana como en la Figura 5.70.

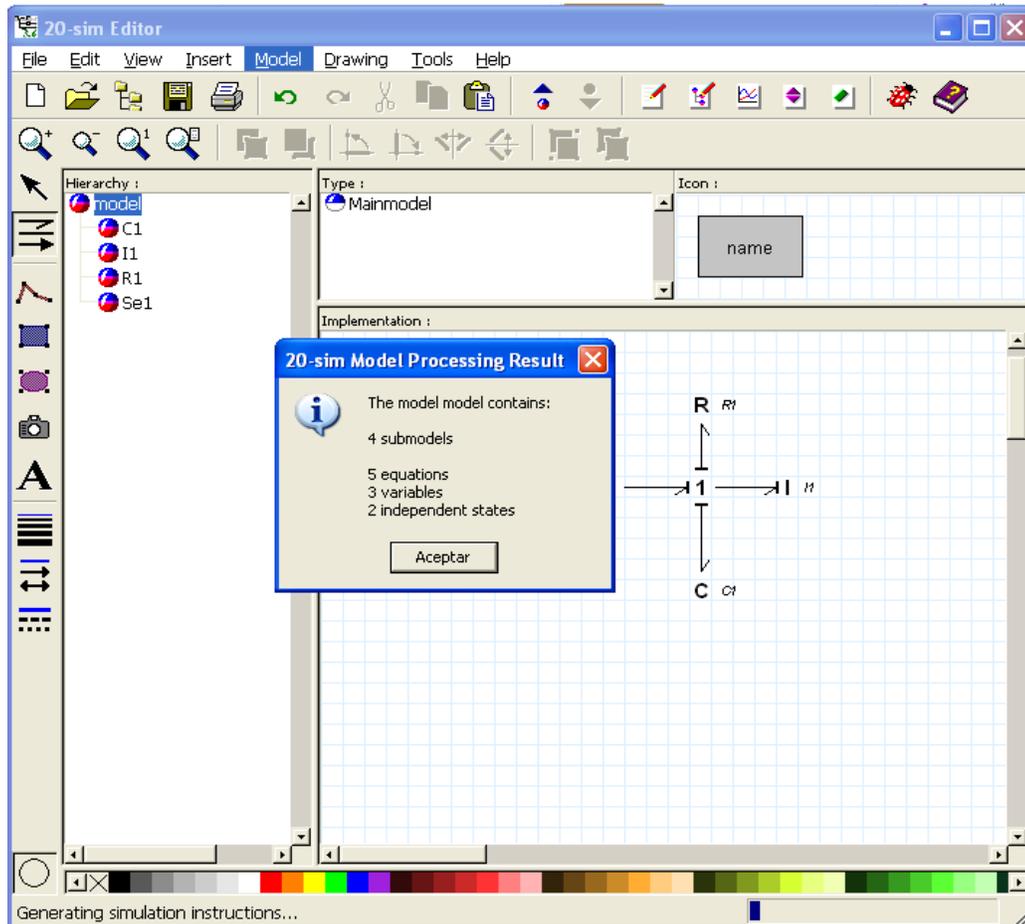


Figura 5.70. Ventana de comprobación de el modelo.

16. Si se produce algún error se debe corregir.
17. En el menú **File**, haga clic en **Save As**, guarde el modelo en una carpeta temporal (por ejemplo c:/temp o seleccione la carpeta de su preferencia) con el nombre de CircuitoRLC.em
18. En la barra de herramientas del editor seleccione el icono **Start Simulation** , esta acción despliega la ventana del simulador. (Ver Figura 5.71).

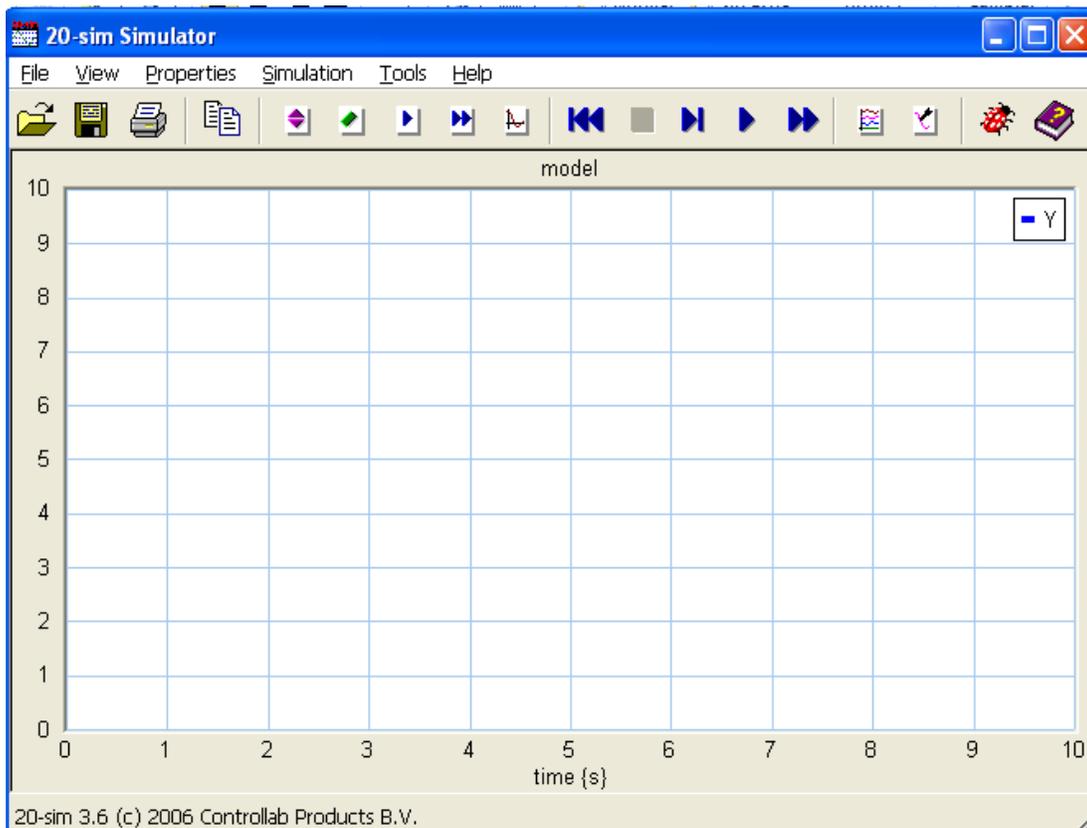


Figura 5.71. Ventana del simulador de 20-sim.

19. En la barra de herramientas del simulador haga clic en **Properties** luego seleccione el comando **Parameters** y cambie los valores de los parámetros por defecto a los siguientes que aparecen en la Tabla 5.7. (Ver Figura 5.72).

Elemento	Valor
R	0.67 Ω
L	1 H
C	5 F
Fuente	5 V

Tabla 5.7. Valores de los elementos del modelo.

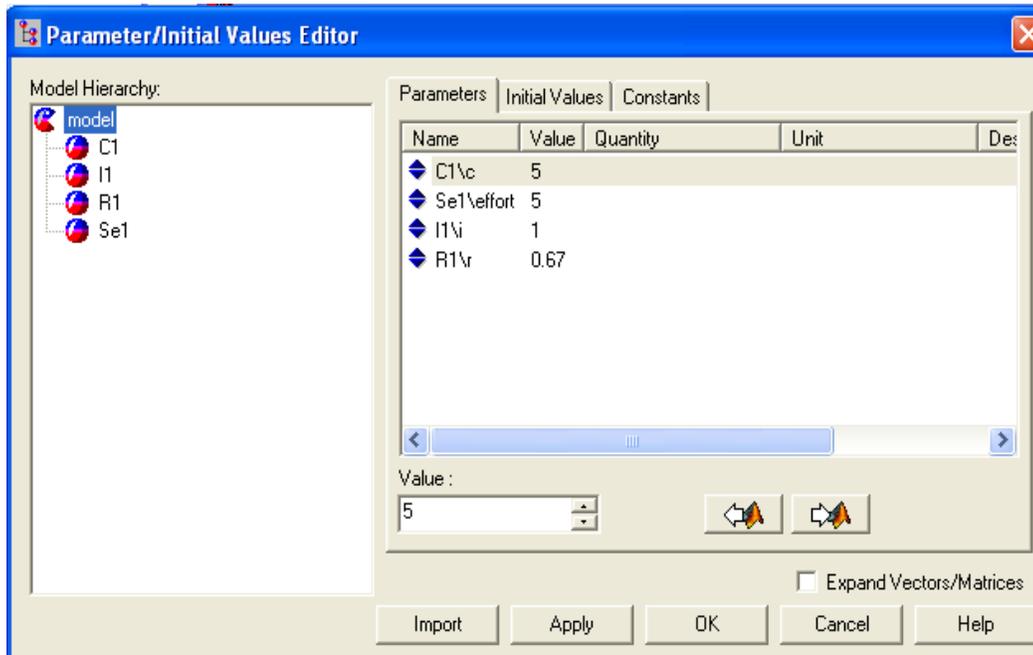


Figura 5.72. Edición de los valores de los elementos del modelo.

20. Desde el menú **Properties** seleccione el comando **Run** y cambie los valores por defecto a los siguientes parámetros de simulación de la Tabla 5.8, observe la Figura 5.73.

Parámetro	Valor
Start	0
Finish	30
Method	Runge-Kutta 4
Step Size	0.1

Tabla 5.8. Valores de los parámetros de simulación del modelo.

21. En el menú **Properties** seleccione el comando **Plot**.
22. En la ventana que se despliega seleccione la pestaña **Plot Properties** para cambiar el título por defecto del modelo a “**Modelo RLC**”.

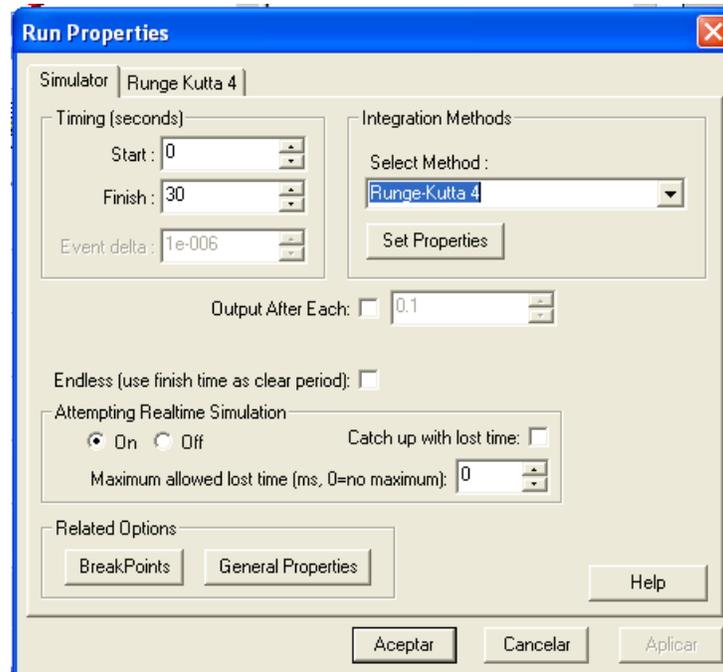


Figura 5.73. Edición de los valores de los parámetros (tiempo) de simulación modelo.

23. Seleccione la pestaña **Y-axis** y haga clic en **Choose** para elegir la variable a simular. En este caso se elige la variable **I1\p.f**, la cual es la corriente del inductor y también la corriente del circuito. Presione con el puntero sobre el submodelo **I1** a la izquierda, luego **p.f** a la derecha y finalmente presione **OK**. (Ver Figura 5.74).

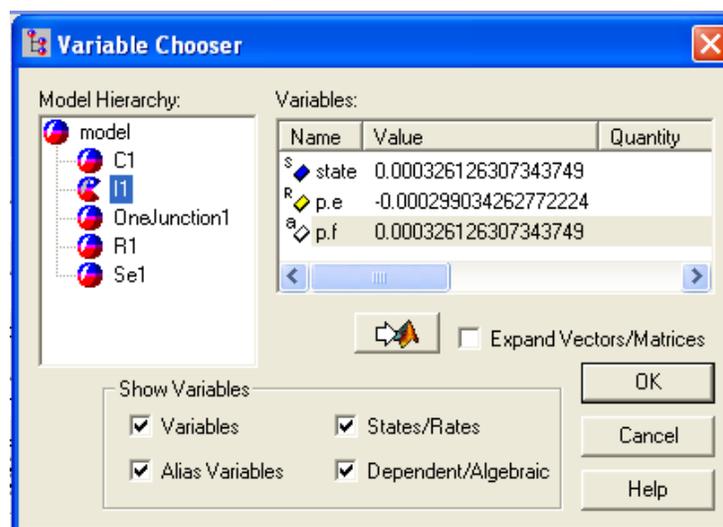


Figura 5.74. Elección de la variable a simular.



- 24. En la celda **Label** coloque el nombre del parámetro que se va a simular “Corriente del Circuito”. Observe la Figura 5.75.
- 25. Establezca los siguientes valores que se muestran en la Tabla 5.9:

Tick Properties	
Tick Style	None
Color	Azul
Scaling	
Scaling	Manual
From	-1
To	5

Tabla 5.9. Valores de los parámetros de simulación del modelo.

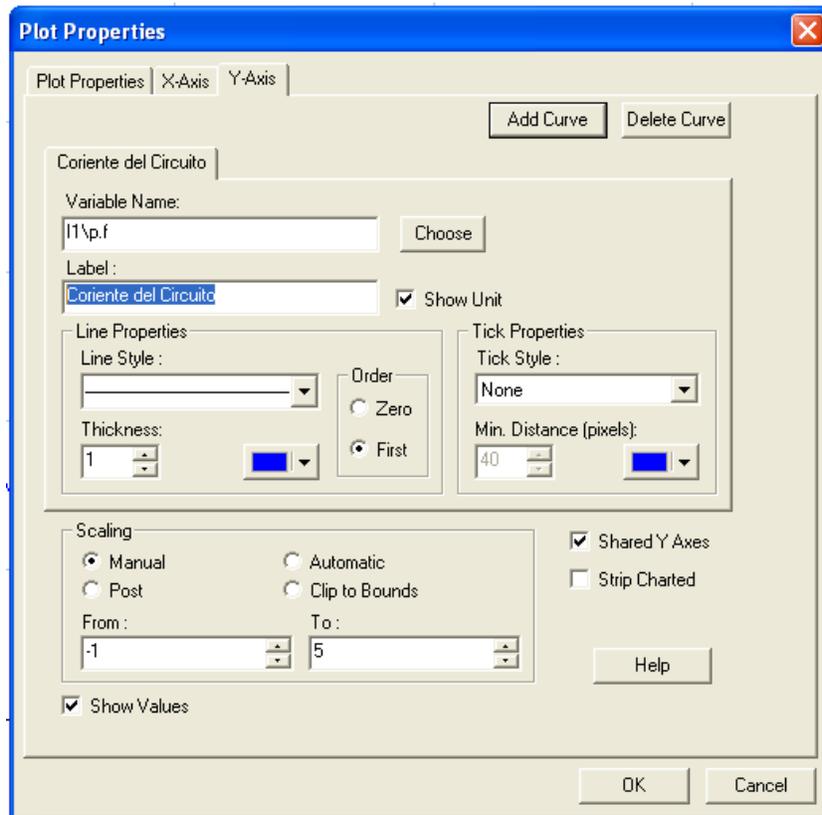


Figura 5.75. Identificación de la variable a simular.

26. Cierre el editor *Plot Properties* haciendo clic en **OK**.
27. Presione el botón  para ejecutar la simulación del experimento. El resultado de la simulación debe ser similar a la Figura 5.76.

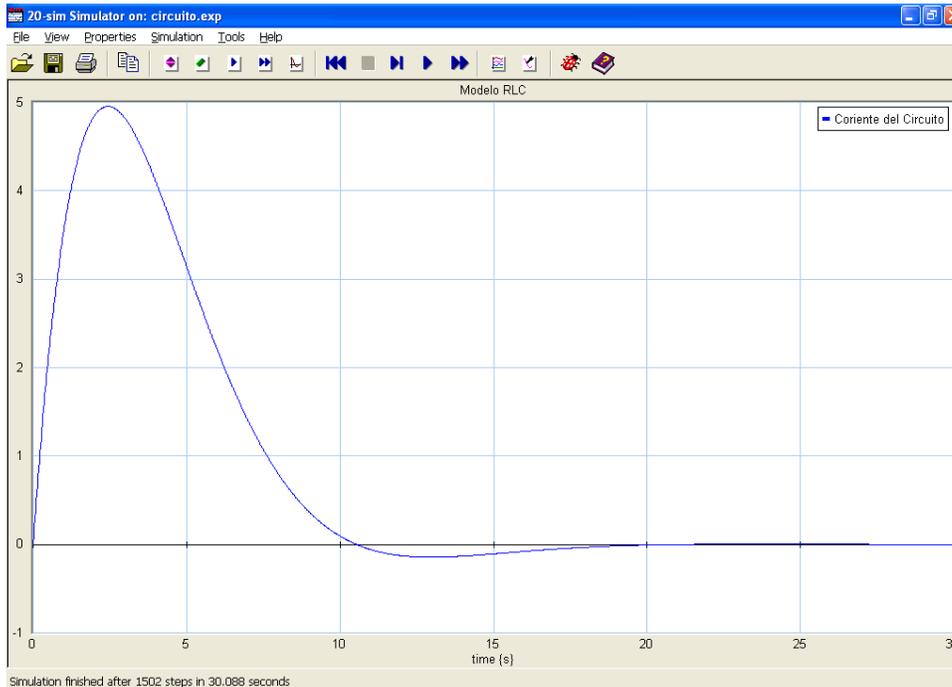


Figura 5.76. Corriente del circuito RLC.

28. Ahora se grafican los diferentes voltajes de cada uno de los elementos (resistor, capacitor, inductor y fuente). Para llevar a cabo esta acción se modifica la variable a simular en cada caso, como en los pasos 21, 23 y 24. En la siguiente Tabla 5.10 se indican las nuevas variables.

Elemento	Variable	Variable en 20-sim	Label
Resistor	Voltaje	R1\p.e	Voltaje del Resistor
Capacitor	Voltaje	C1\p.e	Voltaje del Capacitor
Inductor	Voltaje	I1\p.e	Voltaje del Inductor
Fuente	Voltaje	Se1\p.e	Voltaje de la Fuente

Tabla 5.10. Valores de los parámetros de simulación del modelo.



En las siguientes figuras (5.77, 5.78, 5.79, 5.80) se muestran las gráficas de los diferentes voltajes de los elementos del circuito eléctrico.

Es posible cambiar las especificaciones de las gráficas según las preferencias del usuario, se incentiva a realizar diferentes cambios en el color de la gráfica, límites del eje Y, tipo de línea de la grafica, tipo de marca (paso 25), los límites de tiempo de simulación (eje X) y tipo de método numérico a utilizar (paso 20), etc. Cada una de las características anteriores hacen del software 20-sim un programa de modelación y simulación versátil, explícito, potente (respecto a los métodos de integración que contiene) y didáctico.

El uso del software 20-sim, desde el punto de vista académico, es una herramienta práctica que amplía el conocimiento del estudiante de ingeniería y carreras afines, en cuanto al rango de métodos aplicables en la resolución de problemas y estudio de sistemas físicos de diferentes dominios y combinaciones de los mismos.

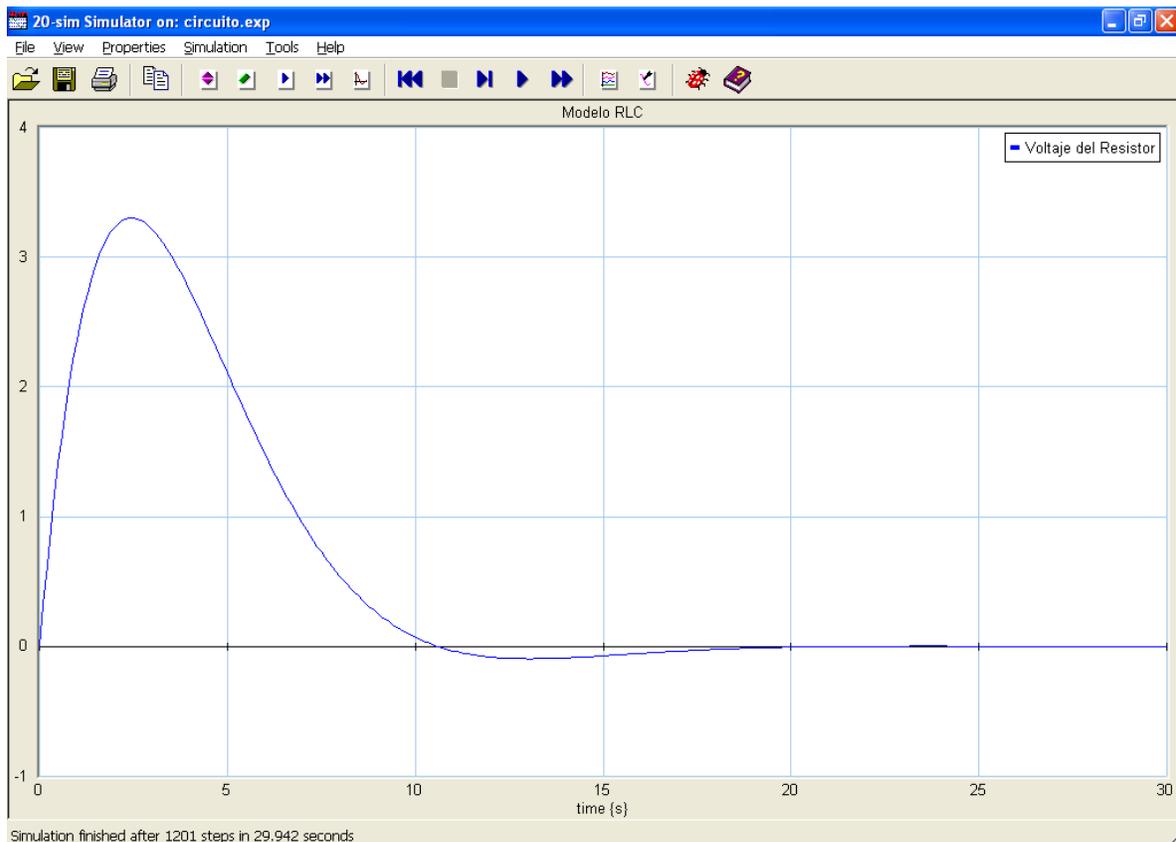


Figura 5.77. Voltaje del Resistor.

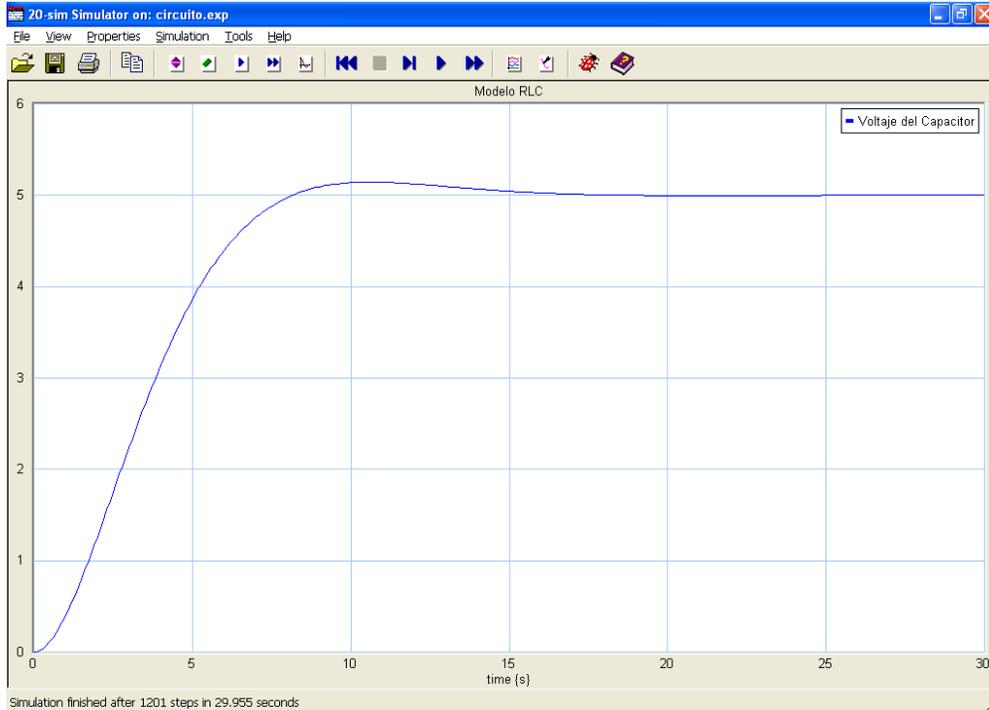


Figura 5.78. Voltaje del Capacitor.

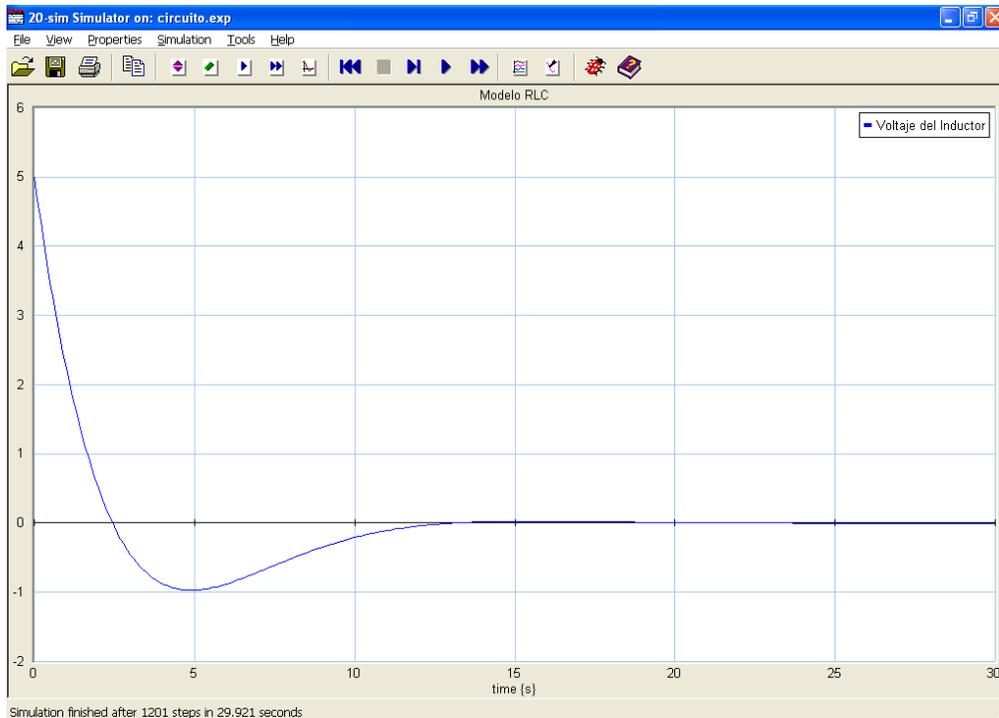


Figura 5.79. Voltaje del Inductor.

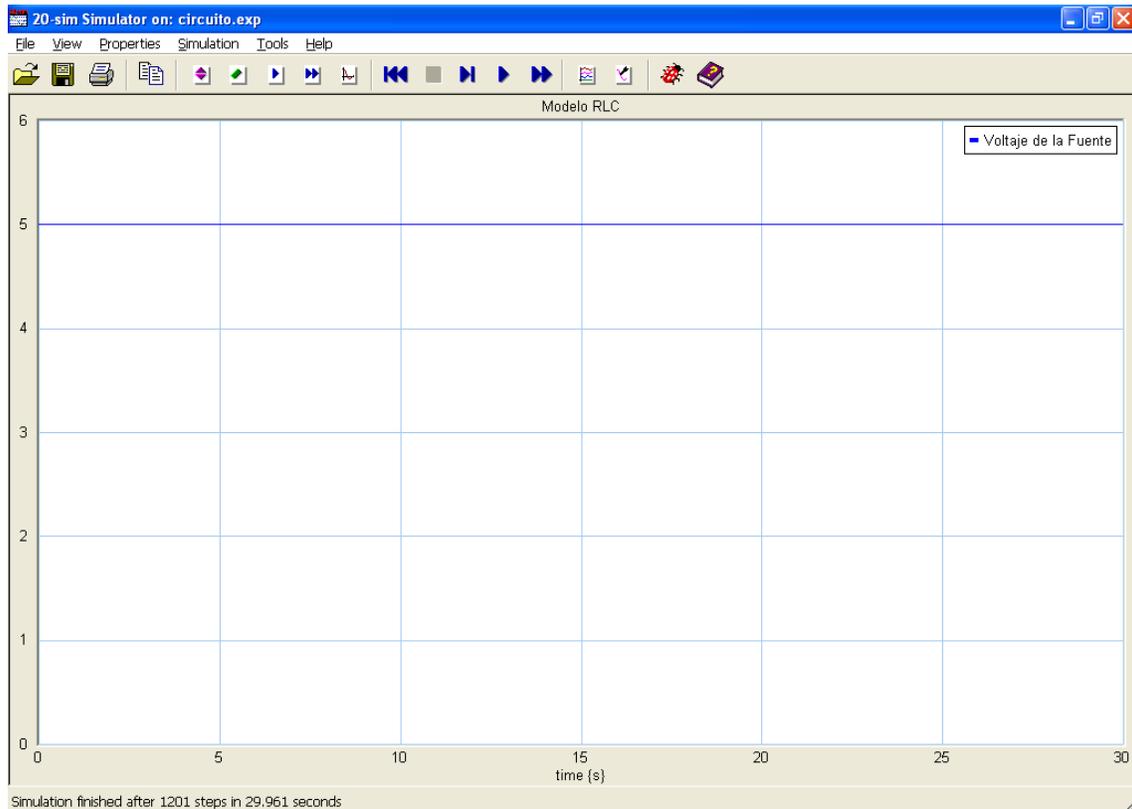


Figura 5.80. Voltaje de la Fuente.

29. Para guardar haga clic en **File** luego **Save**.

5.8.3 CIRCUITO ELÉCTRICO MIXTO

El circuito mixto que se presenta a continuación en la Figura 5.81 contiene dos resistencias (R_1 y R_2), un inductor (I), un capacitor (C) y la fuente de voltaje (E). En la Figura 5.82 se presenta el grafo de enlace (bond graph) del mismo circuito [8].

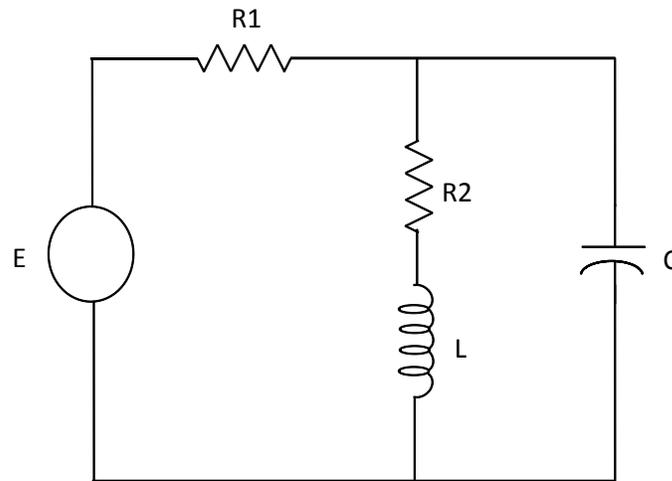


Figura 5.81. Circuito Eléctrico Mixto

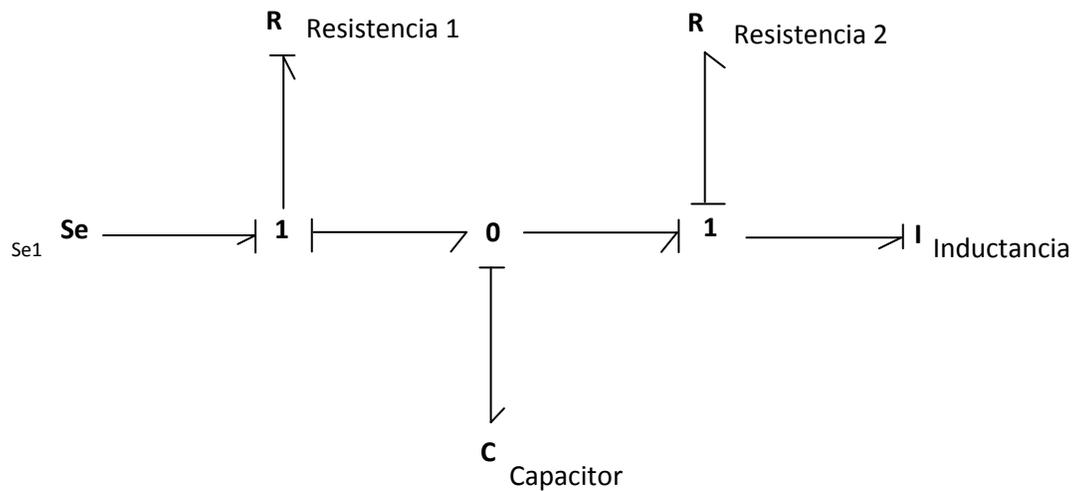


Figura 5.82. Modelo de Bond Graph del Circuito Eléctrico Mixto

1. Inicie 20-SIM y en el menú **File** seleccione **New** y luego **Graphical Model**. Se abrirá el editor del 20-SIM, asegúrese de que 20-SIM esté en modo depuración.

Esto se indica en la barra de herramientas con el botón **Mode** . (Ver Figura 5.83)

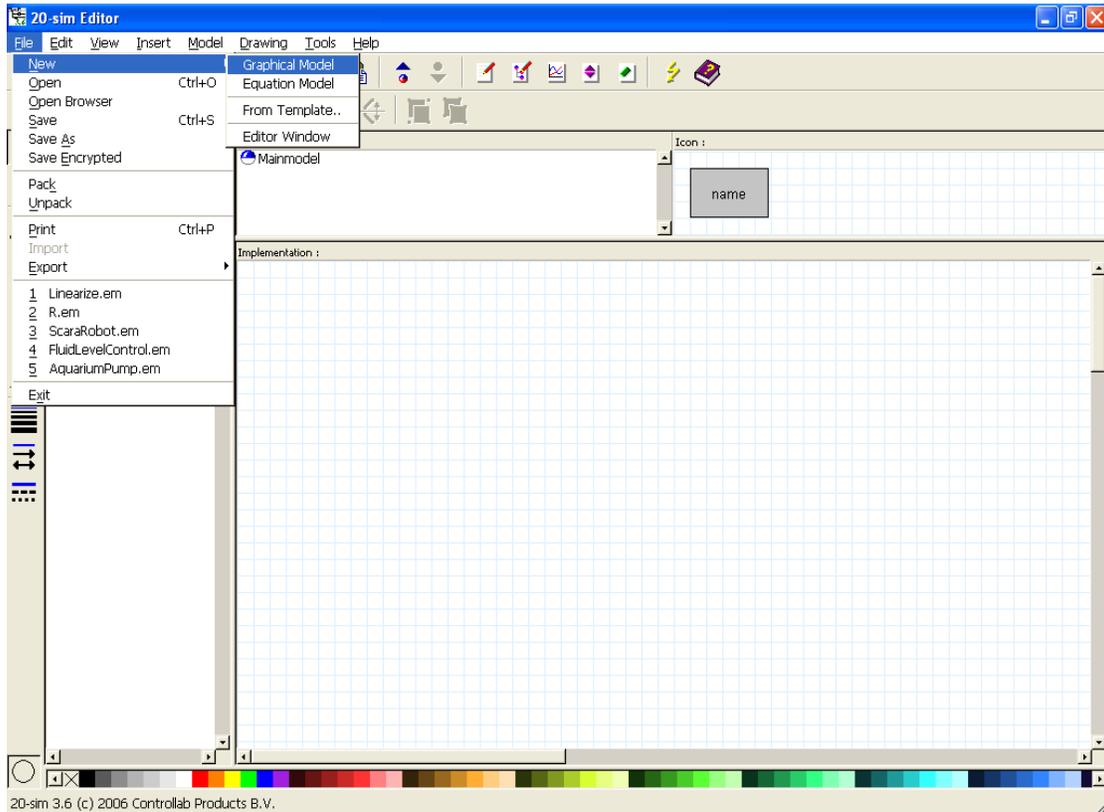


Figura 5.83. Nuevo Modelo.

2. Si el editor de 20-sim está en modo rápido indicado por el botón **Mode**  , haga clic hasta que cambie al modo depuración  .
3. En el menú derecho del ratón sobre la ventana **Implementation**, seleccione **Insert** y luego seleccione **Knot** para colocar cada uno de las uniones que conforma el grafo de enlace. Fíjese en las Figuras 5.84 y 5.85.
4. Desde el navegador de la biblioteca de 20-sim en la carpeta de Bond Graph, presione con el ratón sobre el submodelo **Se** y arrástrelo hacia la ventana **Implementation** como en la Figura 5.86.

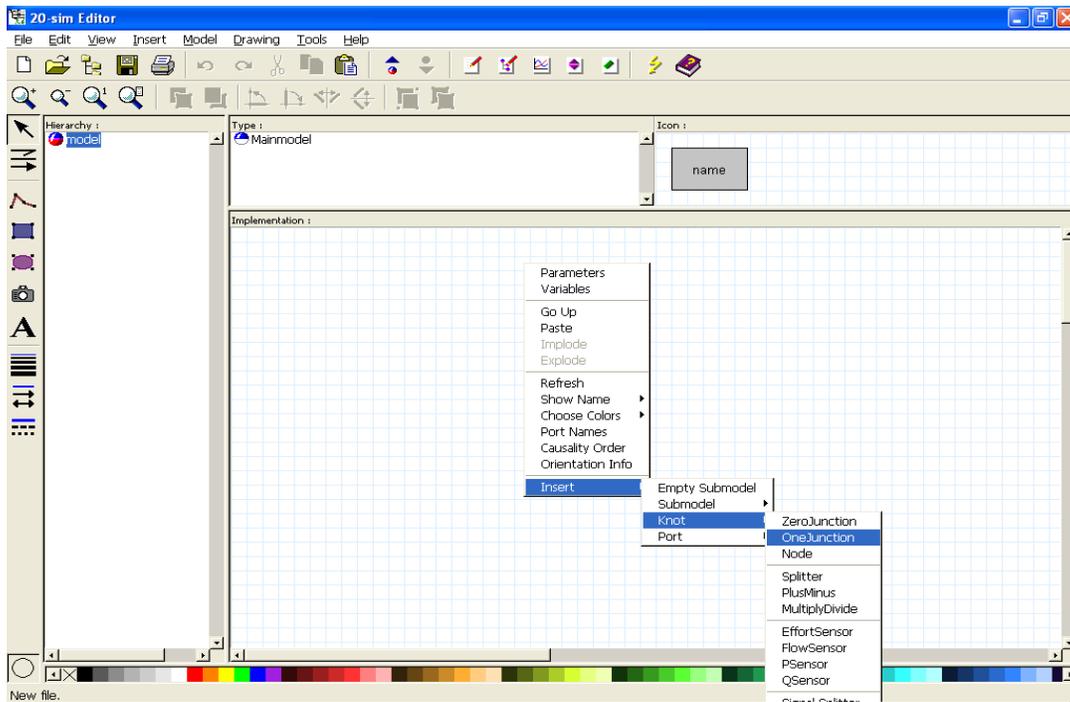


Figura 5.84. Inserción desde el menú del botón derecho del ratón.

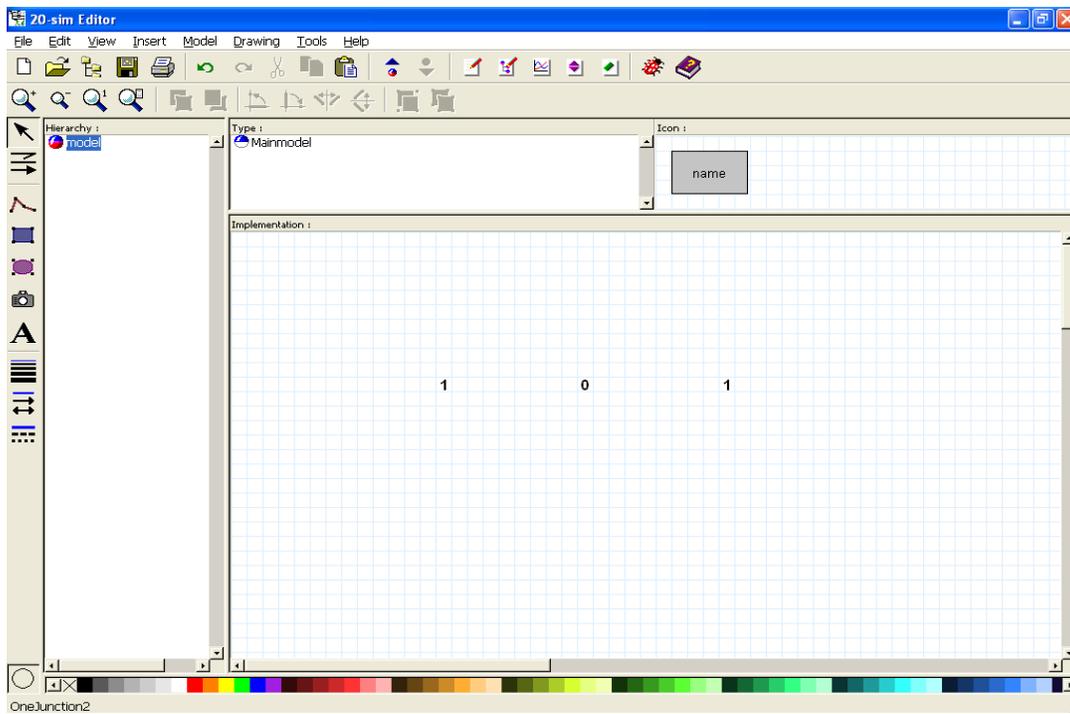


Figura 5.85. Inserción de uniones 0 y 1.

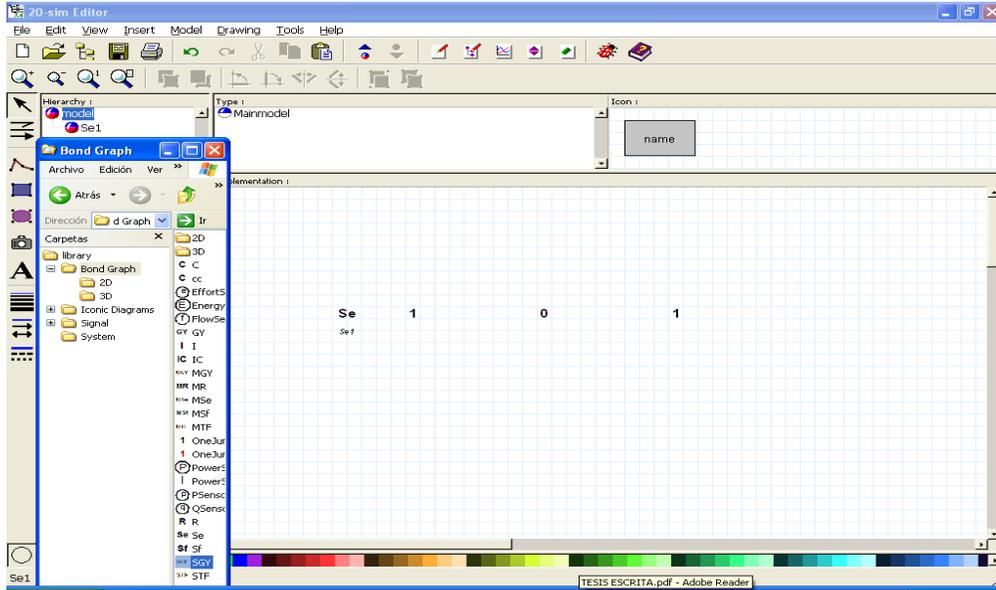


Figura 5.86. Inserción del elemento *Se*.

5. Seleccione el elemento *Se* en la ventana **Implementation** y desde el botón derecho del ratón seleccione **Attributes**. Se abrirá el editor de atributos, asigne un nuevo nombre si lo prefriere. Observe la Figura 5.87.

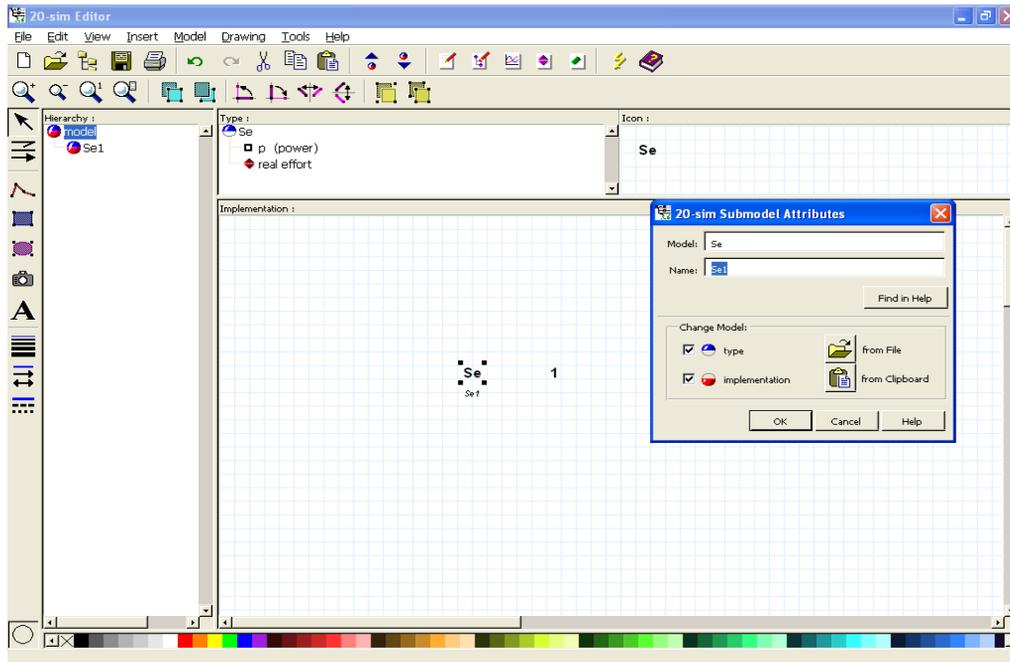


Figura 5.87. Edición del nombre del elemento *Se*.

6. Seleccione cada uno de los elementos (R , C , I) que conforman su modelo desde la librería de Bond Graph a la ventana *Implementation* hasta completar su grafo de enlace como en la Figura 5.88.

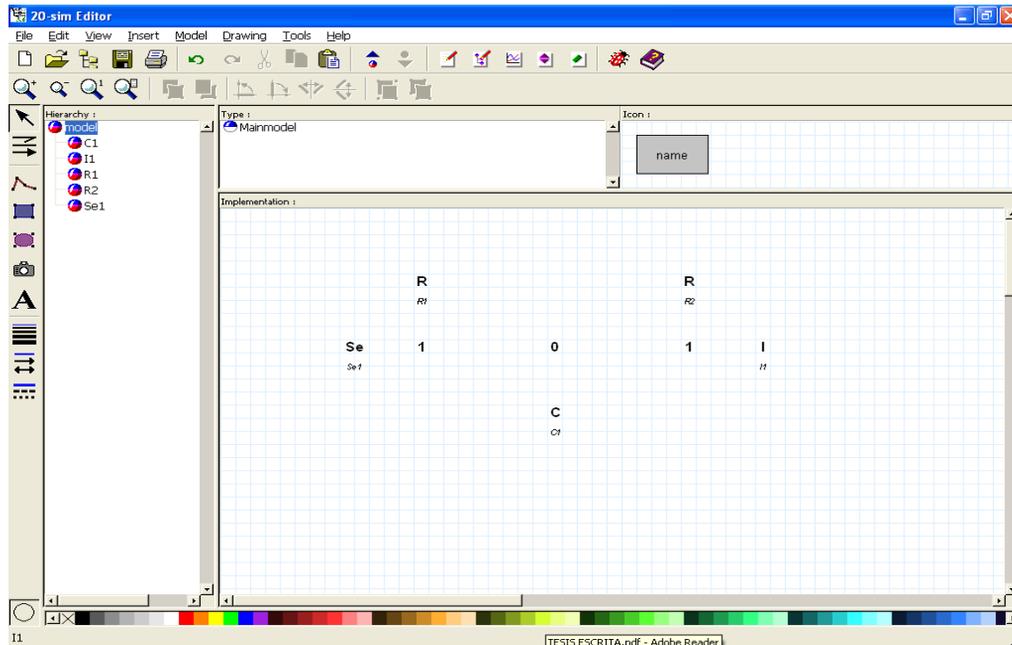


Figura 5.88. Inserción de los elementos restantes.

7. Ubicándose sobre un elemento, pulsando el botón derecho del ratón y seleccionando el comando *Show Name* se puede colocar el nombre del elemento en la posición que desee como se muestra en la Figura 5.89.
8. En la barra de herramientas a la izquierda de la ventana *Hierarchy*, haga clic en el botón *Connect*  para colocar el modo de conexión.
9. Seleccione el submodelo *Se* (clic con el ratón en el botón izquierdo en la parte superior del elemento) y extienda la conexión hacia la primera Unión 1 para crear el primer enlace. Haga este paso hasta totalizar las conexiones empleadas en su grafo de enlace, observe la Figura 5.90. Recuerde que el flujo de potencia positivo va desde la fuente hasta los elementos (de izquierda a derecha en este caso), como se ha realizado en ejercicios anteriores.

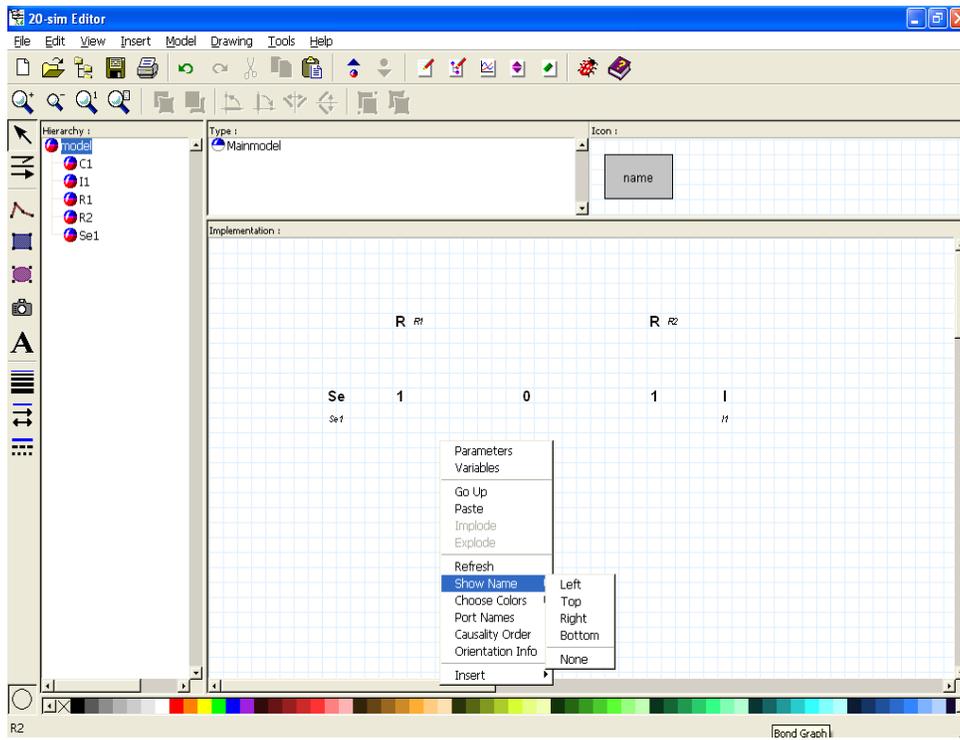


Figura 5.89. Cambio de posición de los nombres de los elementos.

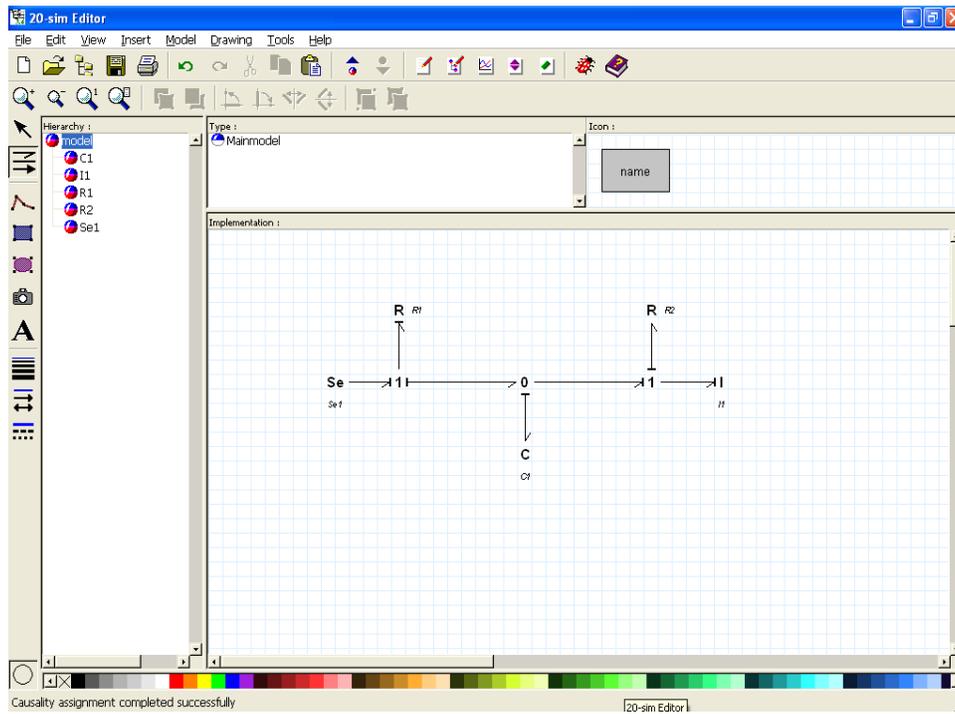


Figura 5.90. Creación de los enlaces (bonds) entre los elementos.

10. En la barra de herramientas, haga clic en el menú **Model** y seleccione el comando **Check Complete Model** para comprobar el modelo completo. Ver Figura 5.91.

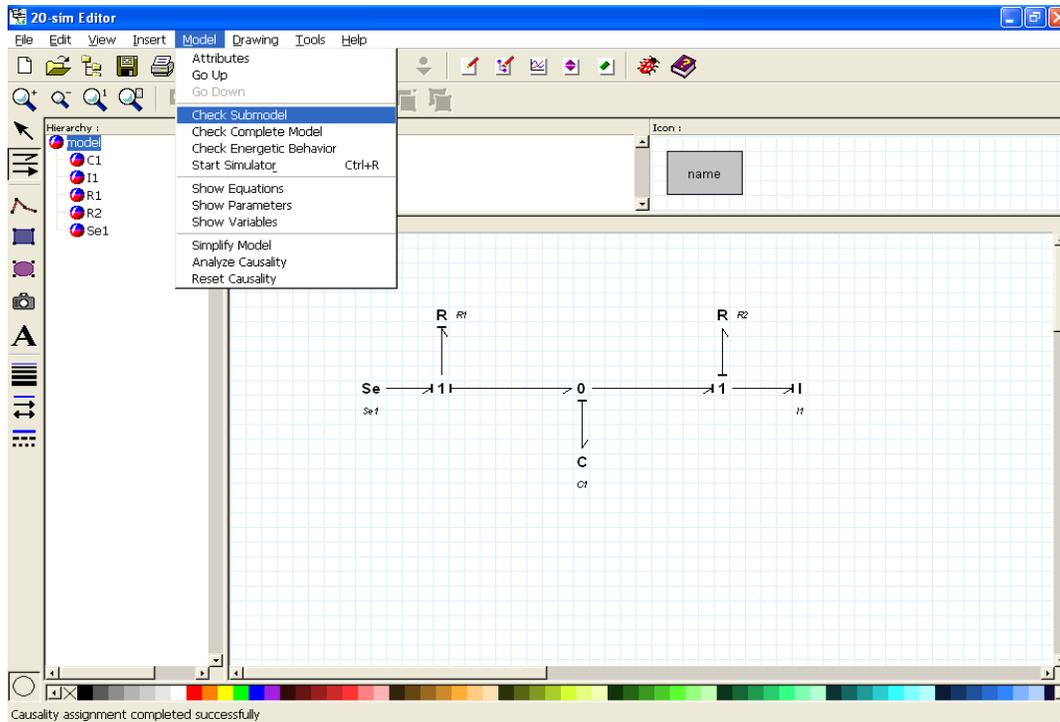


Figura 5.91. Comprobación del modelo creado.

Ahora el modelo será revisado completamente. Si el modelo es correcto una ventana (Figura 5.92) de mensaje aparece (en modo depuración) la cual muestra el número de ecuaciones, variables, etc. Si se encuentra algún error una ventana de mensaje aparecerá, mostrando los errores que ha encontrado el software 20-sim.

11. Si se produce algún error se debe corregir.
12. En el menú **File**, haga clic en **Save As** guarde el modelo en una carpeta temporal (por ejemplo c:/temp) con el nombre de CircuitoMixto.em o el que prefiera elegir.
13. En la barra de herramientas del editor seleccione el botón **Start Simulation** , se desplegará la ventana del simulador. (Ver Figura 5.93).

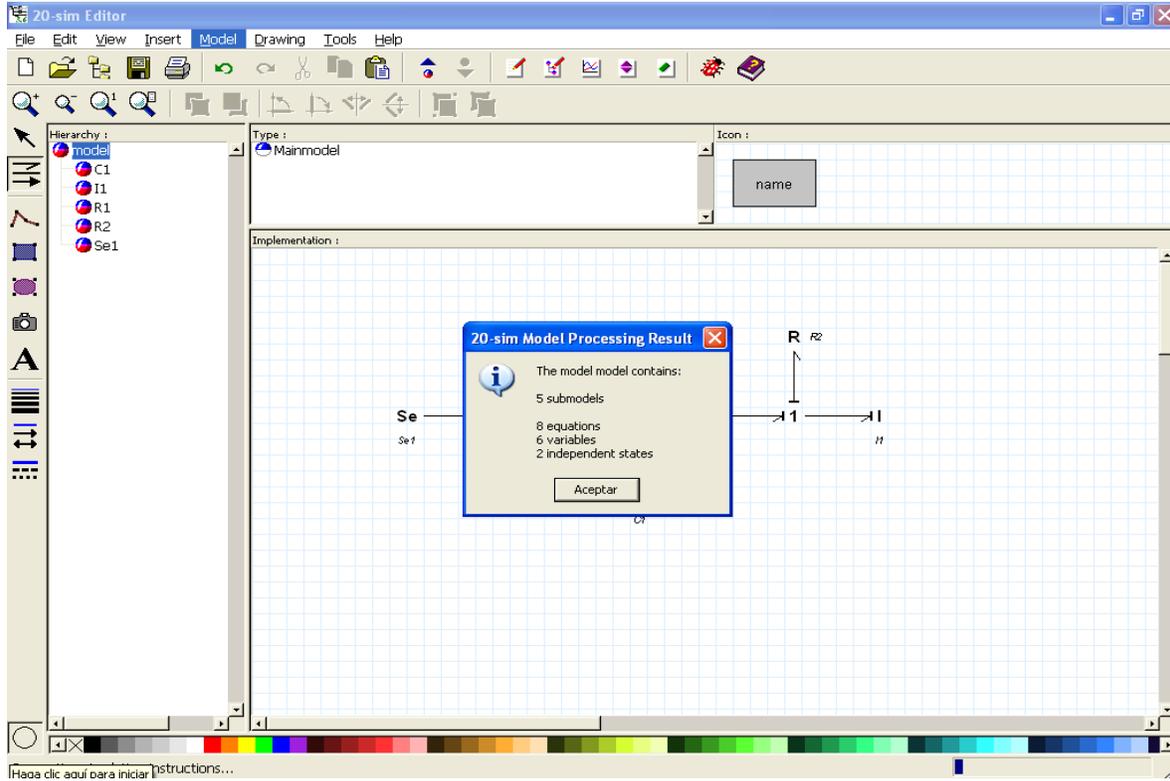


Figura 5.92. Ventana de comprobación exitosa.

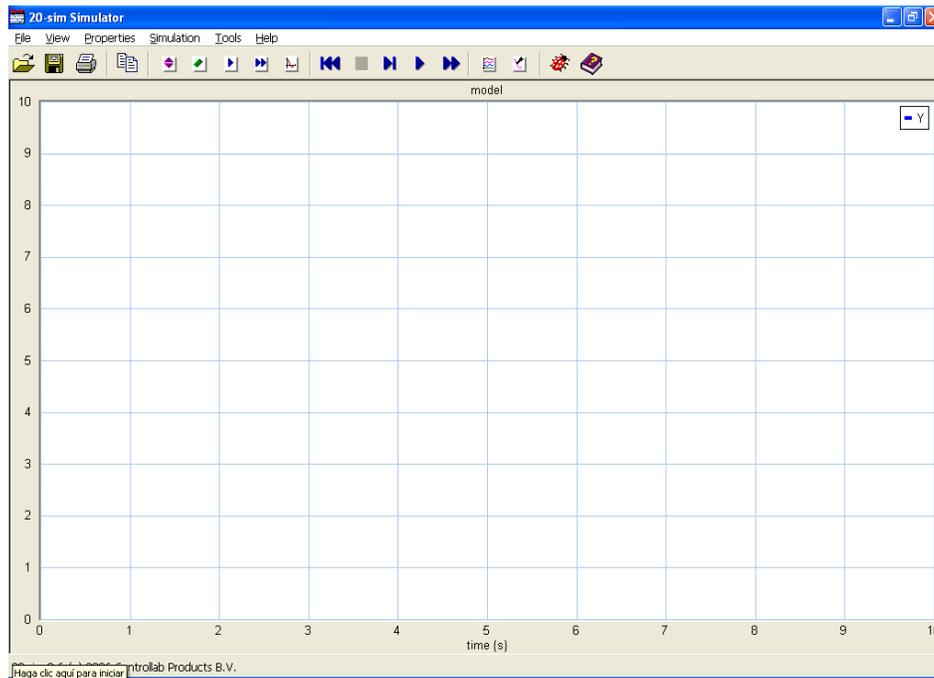


Figura 5.93. Ventana del Simulador.

14. En la barra de herramientas del simulador, haga clic en **Properties**, luego seleccione el comando **Parameters** y cambie los valores de los parámetros por defecto. Observe la Tabla 5.11 y las Figuras 5.94 y 5.95, en las cuales se muestran los valores de los elementos por defecto y los nuevos valores respectivamente.

Elemento	Valor
R1	1
R2	1
C	5
L	2
U	5

Tabla 5.11. Valores de los elementos del modelo.

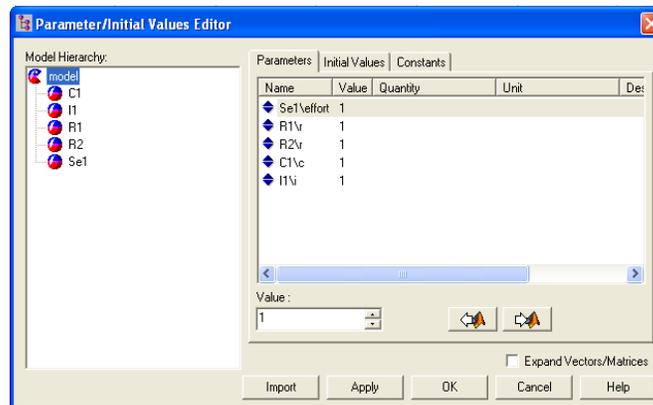


Figura 5.94. Editor de valores de los parámetros del modelo.

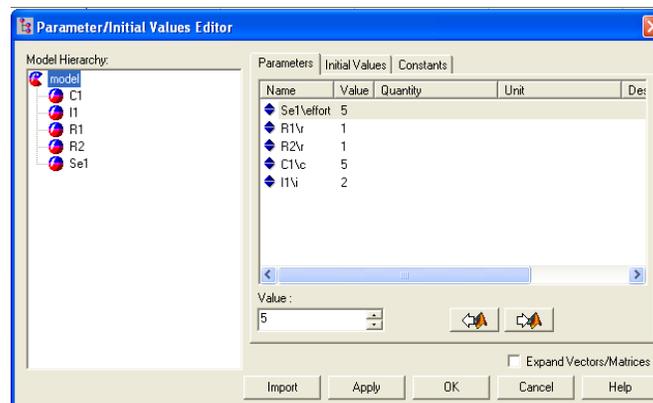


Figura 5.95. Edición de los valores de los parámetros del modelo.

15. Desde el menú **Properties**, seleccione el comando **Run** y cambie los valores por defecto de los parámetros de ejecución de la simulación, observe la Tabla 5.12 y la Figura 5.96.

Parámetro	Valor
Start	0
Finish	10
Method	Runge-Kutta 4
Step Size	0.1

Tabla 5.12. Valores de los parámetros de ejecución de la simulación.

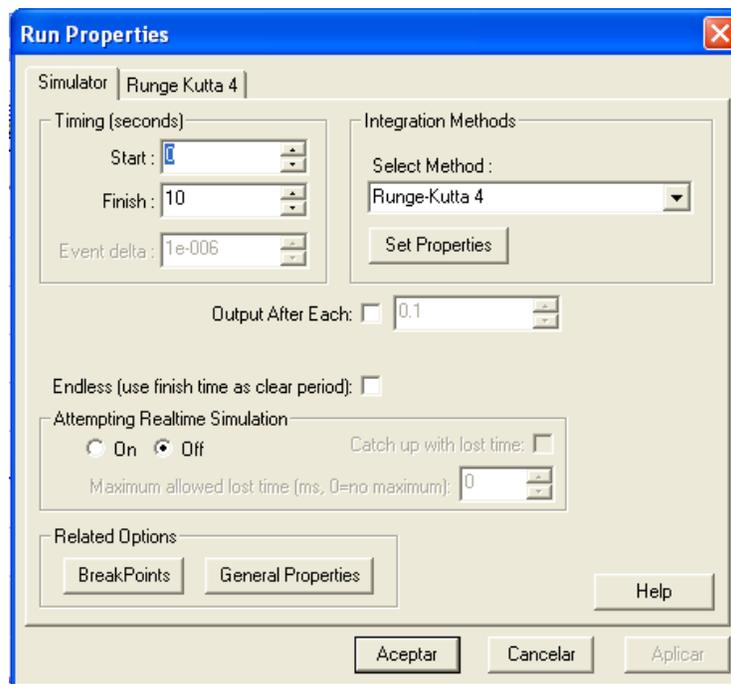


Figura 5.96. Edición de los valores de los parámetros de ejecución de la simulación.

16. En el menú **Properties**, seleccione el comando **Plot**. Se desplegará una ventana para editar las propiedades de la gráfica de la simulación.

17. Seleccione la pestaña **Plot Properties** para cambiar el título por defecto del modelo “*model*” al siguiente “*Circuito Mixto*”. Escriba el nuevo nombre en el espacio **Plot Title** en la parte superior de la ventana. Fíjese en la Figura 5.97.

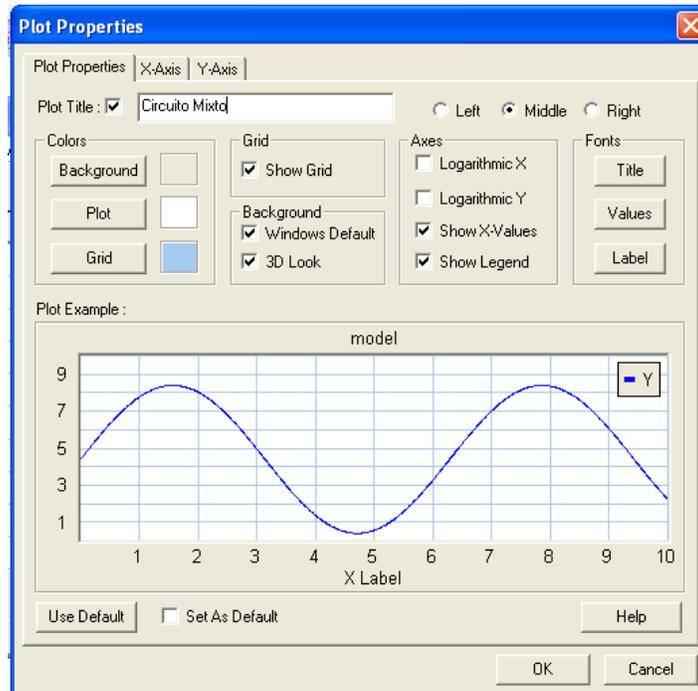


Figura 5.97. Edición de las propiedades de la gráfica en la simulación.

18. Ahora, en la misma ventana, seleccione la pestaña **Y-axis** y haga clic en **Choose** para elegir la variable a simular. En este caso se selecciona la variable **C

.e** la cual representa el voltaje del capacitor, seleccione **CI** la izquierda y luego **p.e** a la derecha como en la Figura 5.98. Presione **OK** para aceptar.

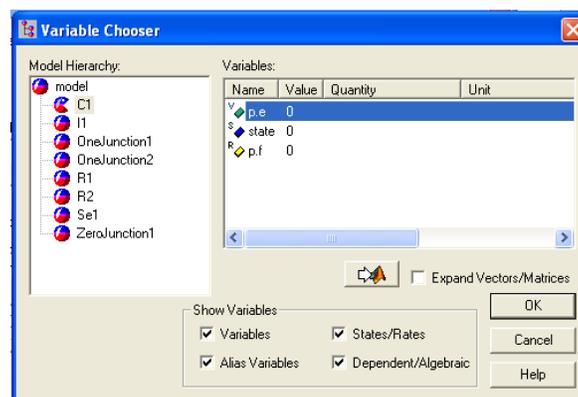


Figura 5.98. Elección de la variable a simular.



19. En el espacio **Label** coloque el nombre de la variable que se va a simular: **“Voltaje del Capacitor”**. Ver Figura 5.99.
20. Establezca los siguientes valores de la Tabla 5.11 en la zonas indicadas en la pestaña **Y-axis** (Ver Figura 5.99):

Tick Properties	
Tick Style	Circle Close
Min. Distance (pixels)	10
Color	Azul
Scaling	
Scaling	Manual
From	0
To	4

Tabla 5.13. Valores de los parámetros las propiedades de la gráfica.

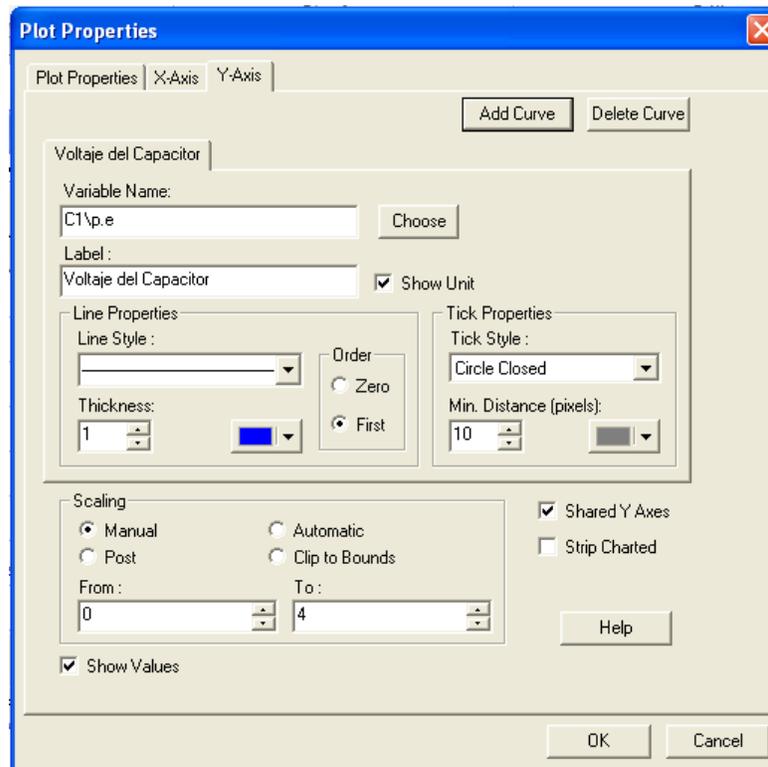


Figura 5.99. Editor de propiedades de la gráfica.



21. Cierre del editor *Plot Properties* haciendo clic en **OK**.
22. En el menú *Simulation*, haga clic sobre el comando **Run** para iniciar la simulación.
El resultado de la simulación debe ser similar a la Figura 5.100.

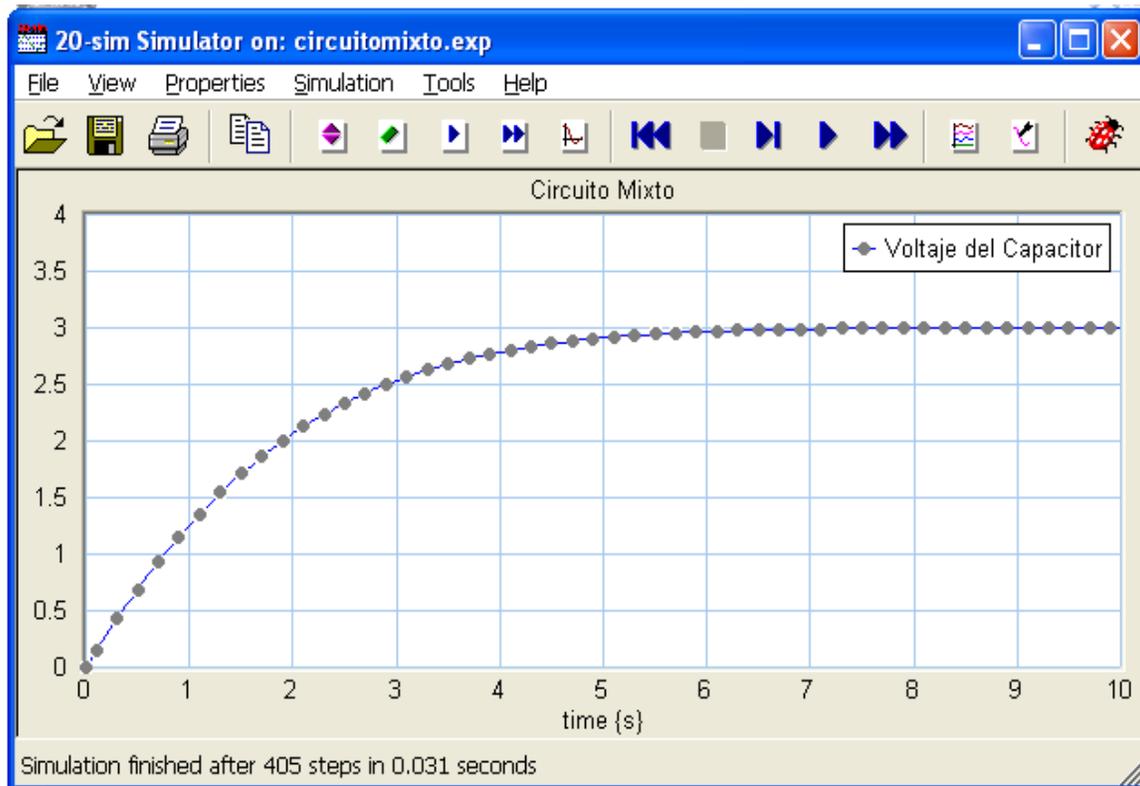


Figura 5.100. Gráfica del voltaje del capacitor.

Después de obtener esta primera grafica, se podrá repetir el procedimiento para obtener cada una de las modelaciones que se deseen del sistema a partir del paso 18, en cual se puede elegir otras variables para simular y observar su comportamiento a través de las gráficas que arroja el software.

Observe las siguientes figuras (5.101, 5.102, 5.103, 5.104, 5.105, 5.106, 5.107, 5.108) en las cuales se observan otras variables relacionadas al circuito tales como la corriente del capacitor, voltajes de los resistores y sus corrientes, voltaje del inductor, etc., y se ajustan los parámetros de apariencia de la gráfica, por ejemplo los límites en el eje Y, para una mejor visualización de los resultados de la simulación.

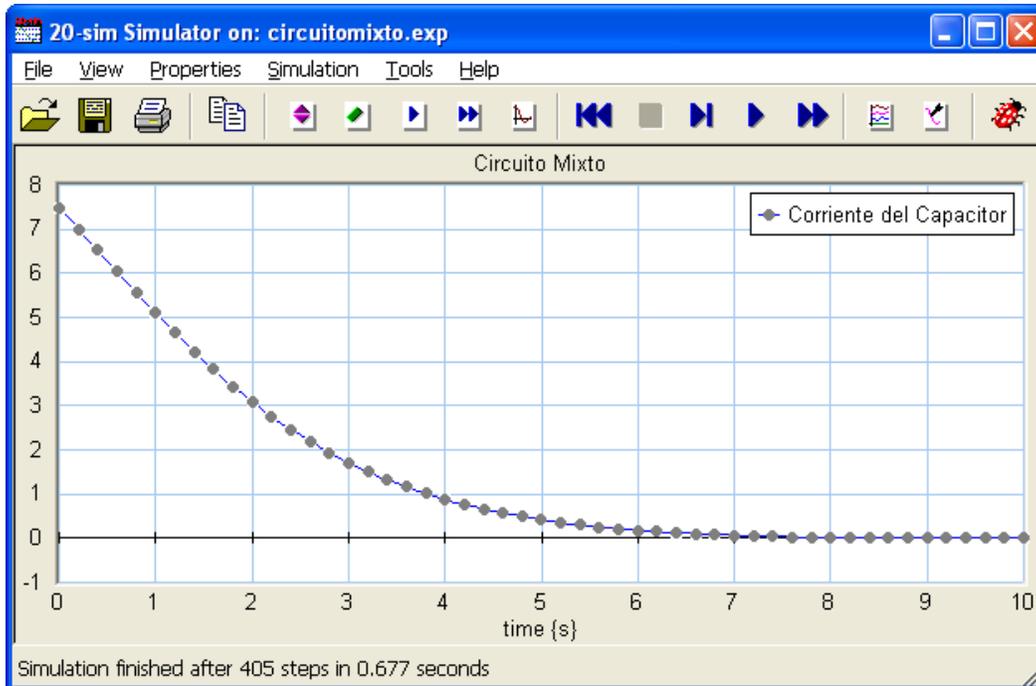


Figura 5.101. Gráfica de la corriente del capacitor.

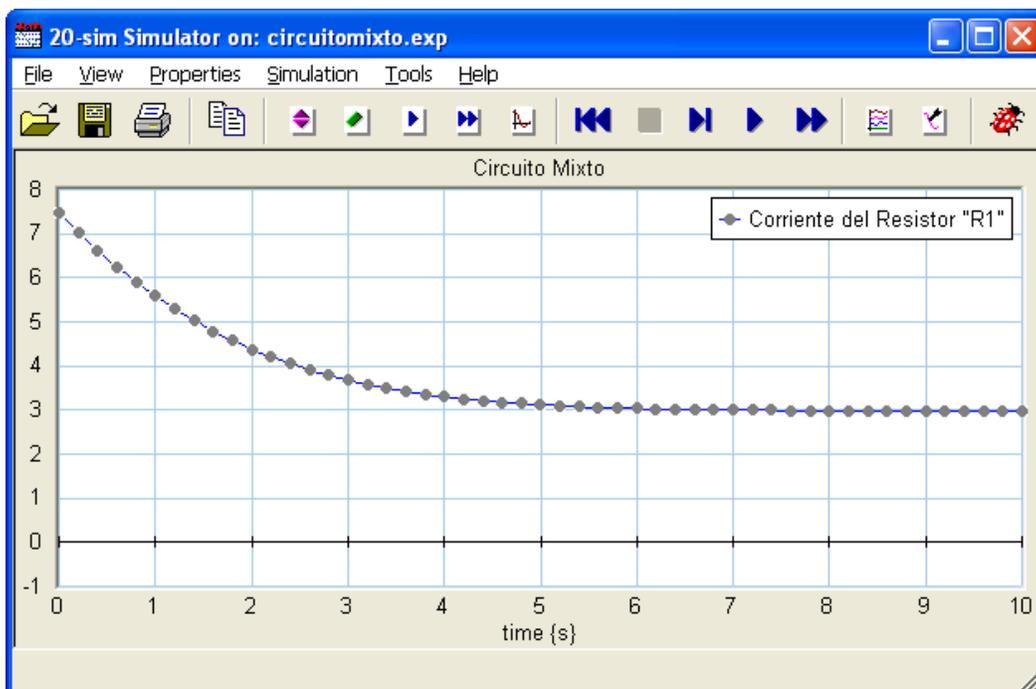


Figura 5.102. Gráfica de la corriente del resistor "R1".

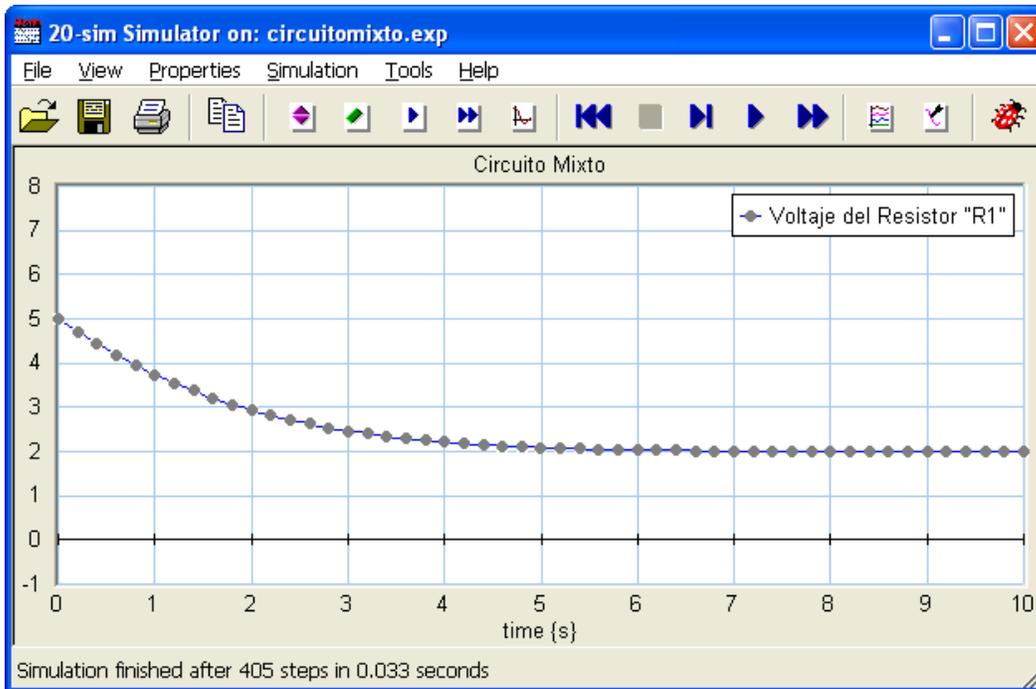


Figura 5.103. Gráfica del voltaje del resistor “R1”.

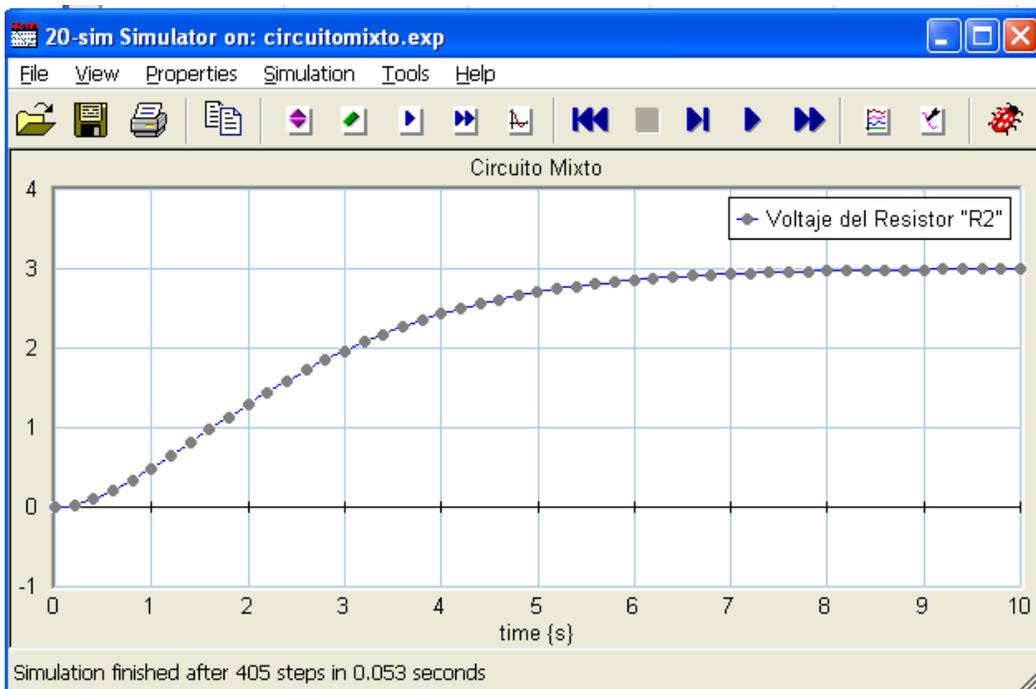


Figura 5.104. Gráfica del voltaje del resistor “R2”.

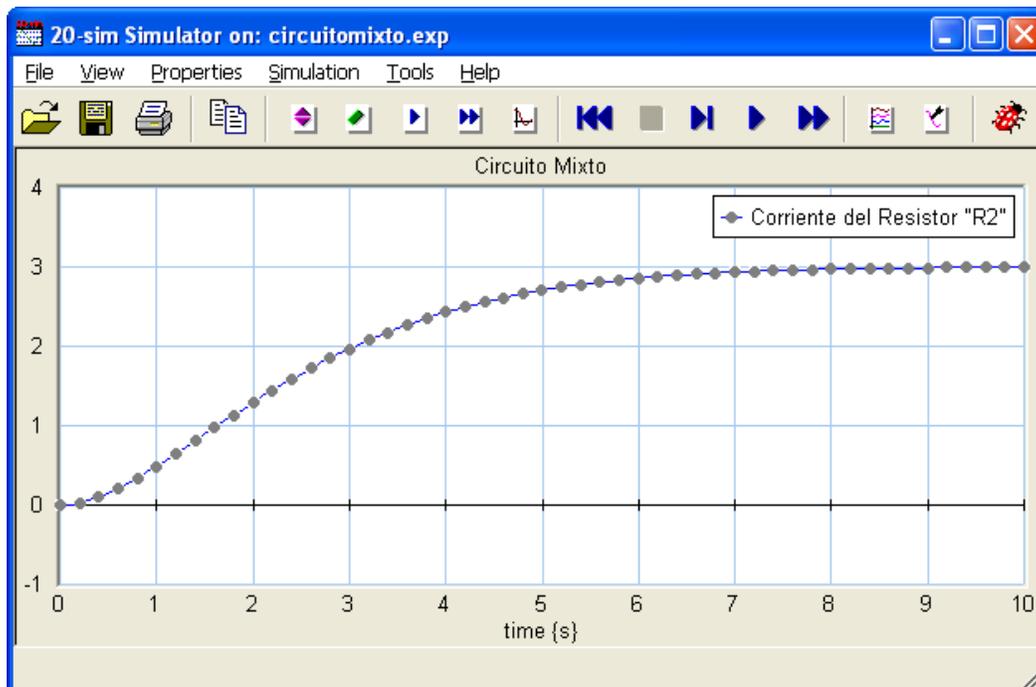


Figura 5.105. Gráfica de la corriente del resistor "R2".

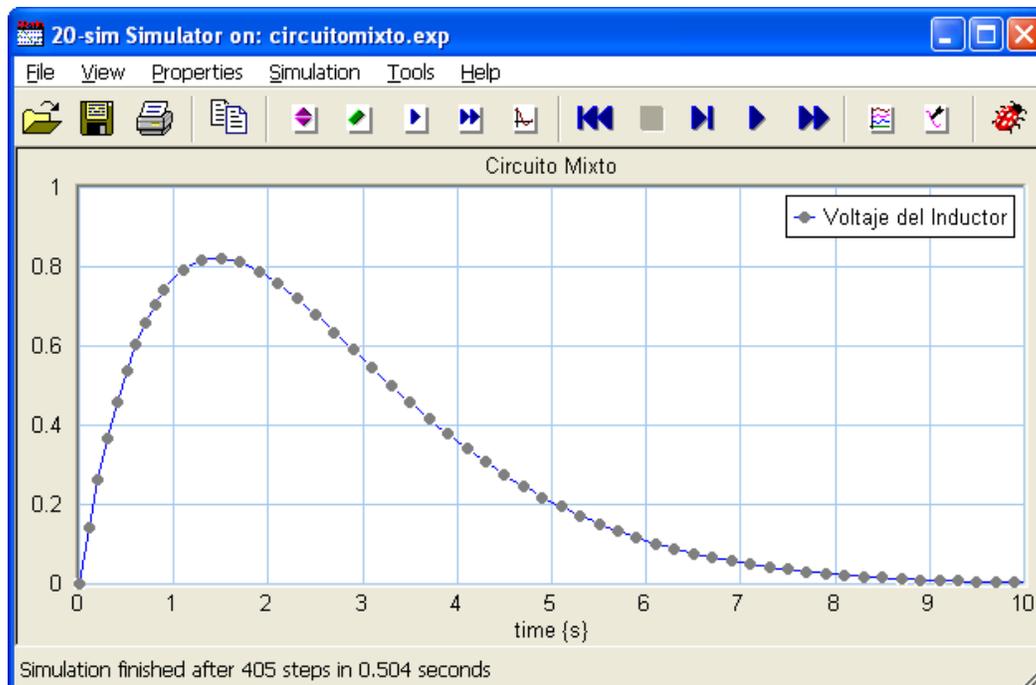


Figura 5.106. Gráfica del voltaje del Inductor.

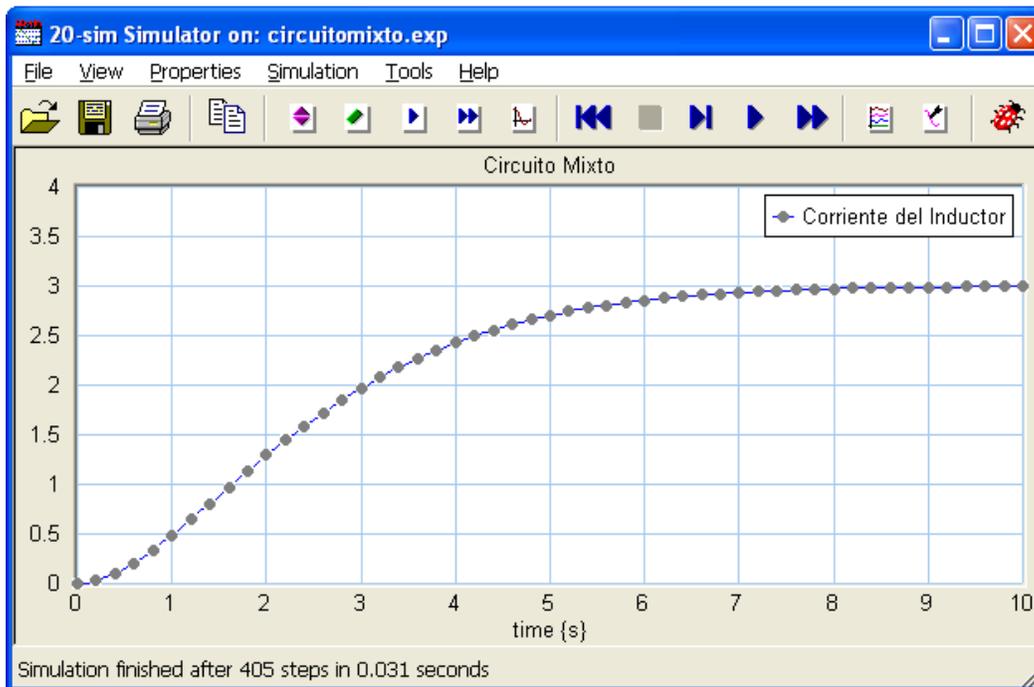


Figura 5.107. Gráfica de la corriente del inductor.

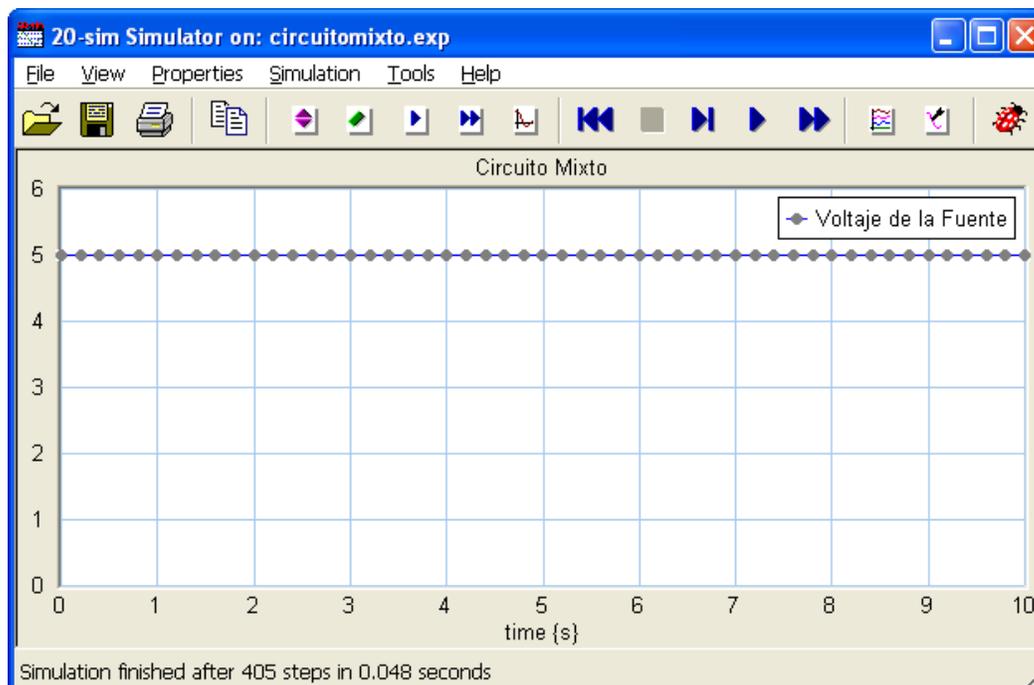


Figura 5.108. Gráfica del voltaje de la fuente.

23. Para guardar haga clic en **File**, luego **Save**.

5.8.4 MOTOR DE CORRIENTE CONTINUA (CC)

Un actuador mecánico muy difundido es el motor de CC. Provee directamente movimiento rotacional y, adecuadamente acondicionado, movimiento de traslación [28]. El circuito eléctrico de armadura y el diagrama mecánico rotacional, se muestran en la Figura 5.109:

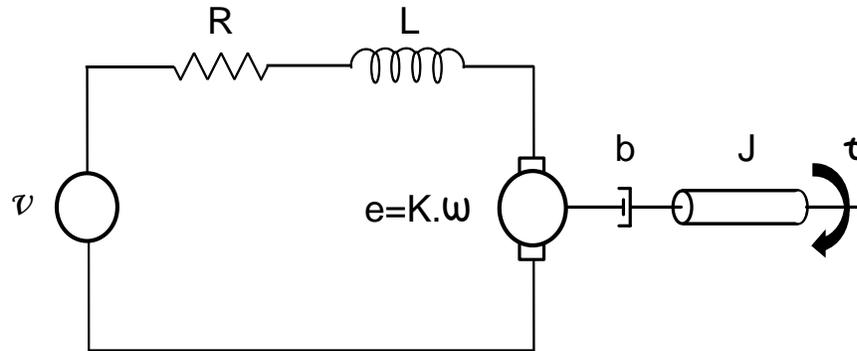


Figura 5.109. Esquema electro-mecánico de un motor de CC.

Donde,

J = Momento de Inercia del Sistema

b = Coeficiente de Roce

K = Constante de Fuerza Electromotriz

e = Fuerza Electromotriz

ω = Velocidad Angular

R = Resistencia de Armadura

L = Inductancia de Armadura

V = Fuente de Tensión

t = Par del Motor

Se considera que el flujo magnético es constante y su valor se integra dentro del valor de la constante de la fuerza electromotriz (K).

Y a continuación, en la Figura 5.110, se muestra el modelo bond graph del esquema del motor CC.

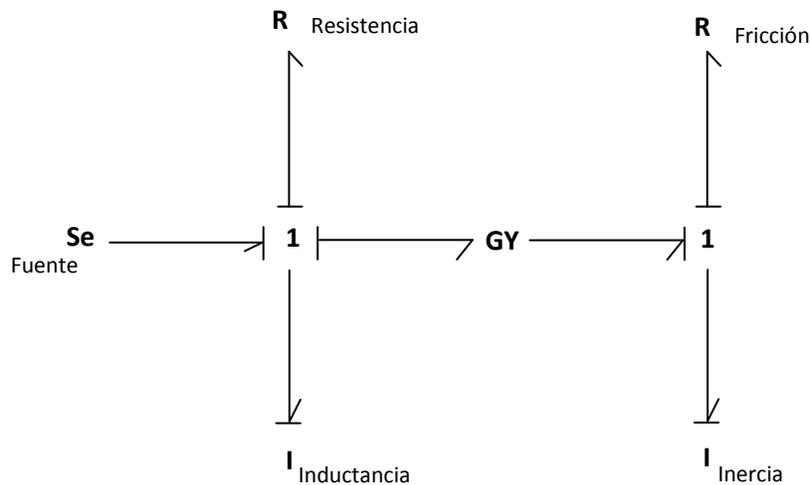


Figura 5.110. Bond Graph del motor de CC.

1. Realice los pasos del ejercicio anterior 5.8.3, desde el número 1 hasta el número 11. Tome en consideración los cambios pertinentes para reproducir el bond graph de la Figura 5.110. A continuación un breve resumen de las acciones que se concretan en este rango de pasos:
 - Inicio del software 20-sim, creación de un nuevo modelo y verificación del modo depuración. (Pasos 1 y 2).
 - Inserción de las uniones y elementos de la biblioteca de bond graph. Introduzca sólo los elementos y uniones que estén en el bond graph de la Figura 5.110. Cambie el nombre y la ubicación del nombre de cada elemento si lo prefiere. (Pasos 3,4,5,6 y 7)
 - Inserción de los enlaces que conectan a los elementos. Tome en cuenta la Figura 5.110 para observar el flujo de la potencia. (Pasos 8 y 9).
 - Comprobación del modelo, corrección de ser necesario. (Pasos 10 y 11).

Al finalizar esta serie de acciones su modelo esta ingresado en el software 20-sim y se debe mostrar como en la Figura 5.111.

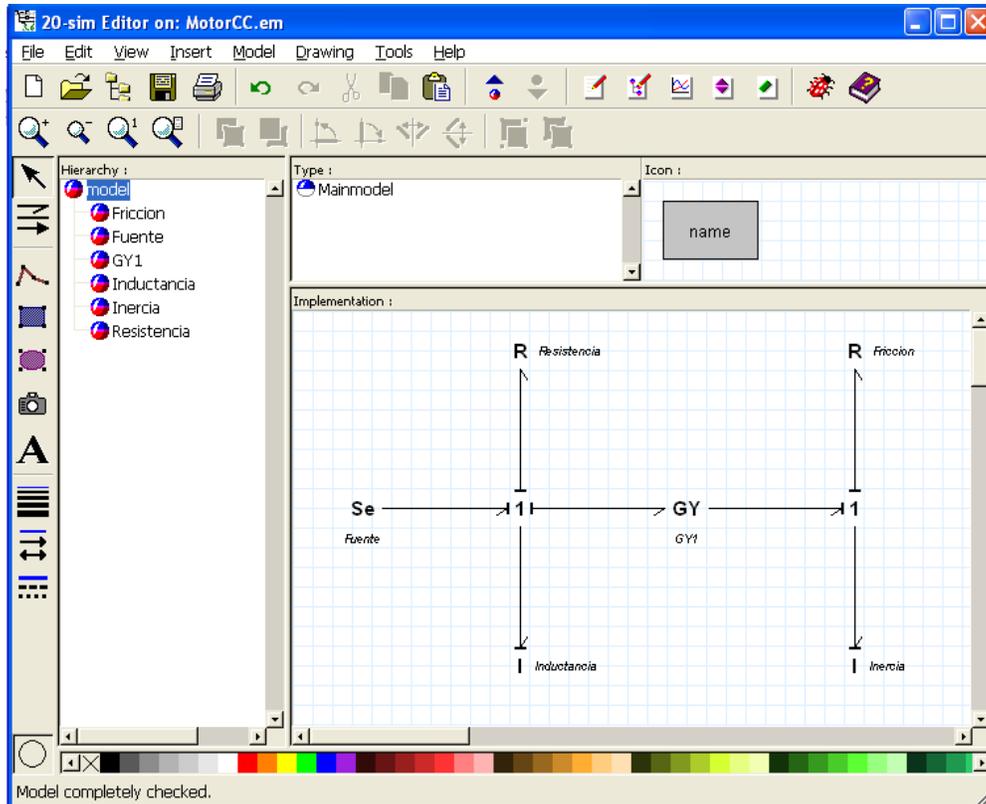


Figura 5.111. Inserción del modelo bond graph del motor de CC en 20-sim.

2. En el menú **File**, haga clic en **Save As** guarde el modelo en una carpeta temporal (por ejemplo c:/temp) con el nombre de MotorCC.em o el que prefiera elegir.
3. Una vez comprobado el modelo, en la barra de herramientas del editor, seleccione el botón **Start Simulation** , se desplegará la ventana del simulador.
4. En la barra de herramientas del simulador, haga clic en **Properties**, luego seleccione el comando **Parameters**, se abre la ventana de edición de los parámetros del sistema. Cambie los valores de los parámetros por defecto. Observe la Tabla 5.14 y la Figura 5.112, en las cuales se muestran los valores de los elementos y la ventana de edición de los mismos, respectivamente.

Elemento	Valor
Se (Fuente)	1
R (Resistencia)	1
I (Inductancia)	0.5
GY (K)	0.01
R (Fricción)	0.1
I (Inercia)	0.01

Tabla 5.14. Valores de los elementos del modelo.

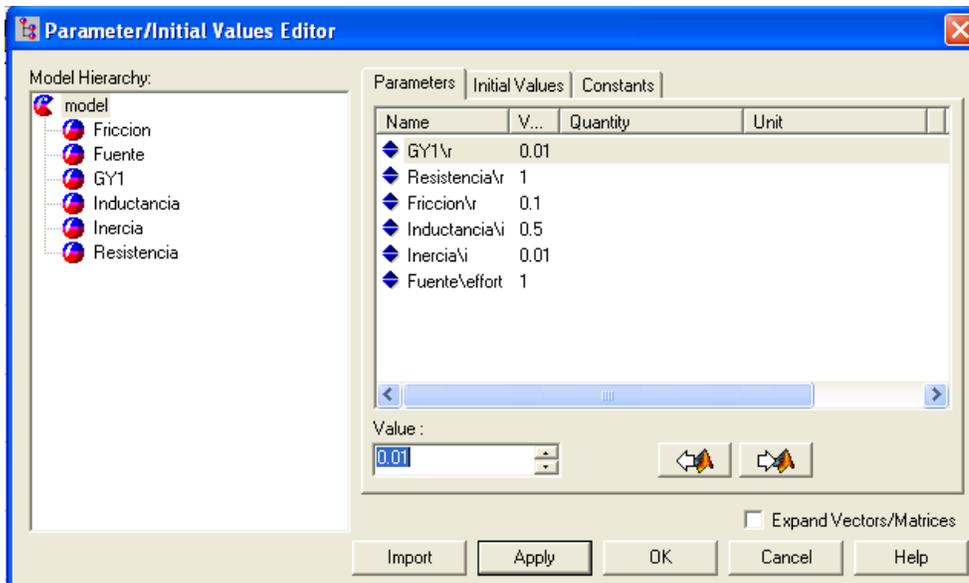


Figura 5.112. Edición de los valores de los parámetros del modelo.

- Desde el menú **Properties**, seleccione el comando **Run** y cambie los valores por defecto de los parámetros de ejecución de la simulación, observe la Tabla 5.15 y la Figura 5.113.

Parámetro	Valor
Start	0
Finish	5
Method	Runge-Kutta 4
Step Size	0.1

Tabla 5.15. Valores de los parámetros de ejecución de la simulación.

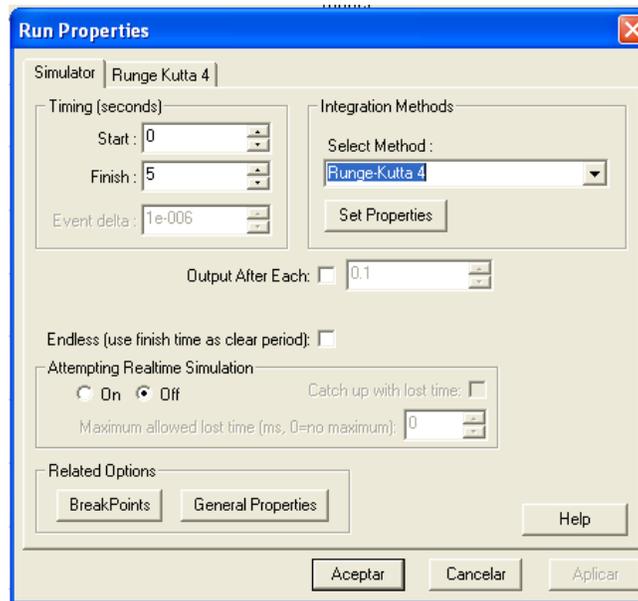


Figura 5.113. Edición de los valores de los parámetros de ejecución de la simulación.

6. En el menú **Properties**, seleccione el comando **Plot**. Se desplegará una ventana para editar las propiedades de la gráfica de la simulación.
7. Seleccione la pestaña **Plot Properties** para cambiar el título por defecto de la gráfica “**model**” al siguiente “**Motor CC**”. Escriba el nuevo nombre en el espacio **Plot Title** en la parte superior de la ventana. Fíjese en la Figura 5.114.

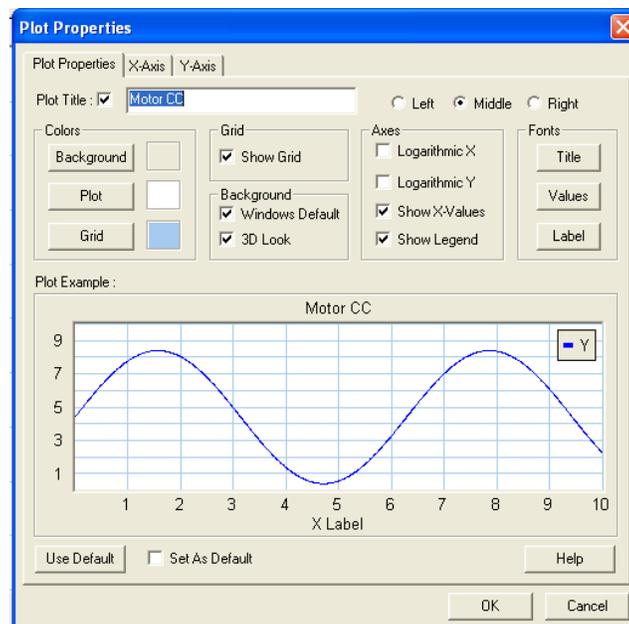


Figura 5.114. Edición de las propiedades de la gráfica en la simulación.

8. En la misma ventana, seleccione la pestaña *Y-axis* y haga clic en *Choose* para elegir la variable a simular. En este caso se selecciona la variable *Fricción**p.f* la cual representa la velocidad angular del motor (ω), seleccione *Fricción* a la izquierda y luego *p.f* a la derecha como en la Figura 5.115. Presione *OK* para aceptar.

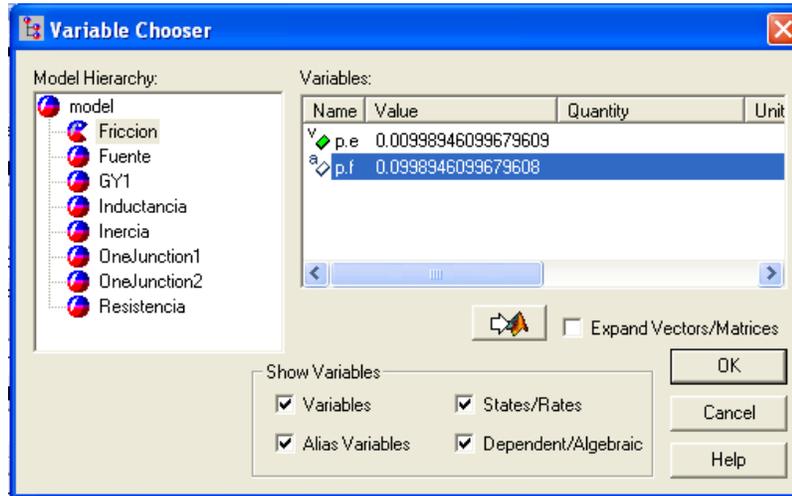


Figura 5.115. Elección de la variable a simular.

9. En el espacio *Label* coloque el nombre de la variable que se va a simular: *“Velocidad Angular del Motor”*. Ver Figura 5.116.
10. Establezca los siguientes valores de la Tabla 5.16 en la zonas indicadas en la pestaña *Y-axis* (Ver Figura 5.116):

Tick Properties	
Tick Style	Cross
Min. Distance (pixels)	10
Color	Negro
Scaling	
Scaling	Manual
From	0
To	0.12

Tabla 5.16. Valores de los parámetros las propiedades de la gráfica.

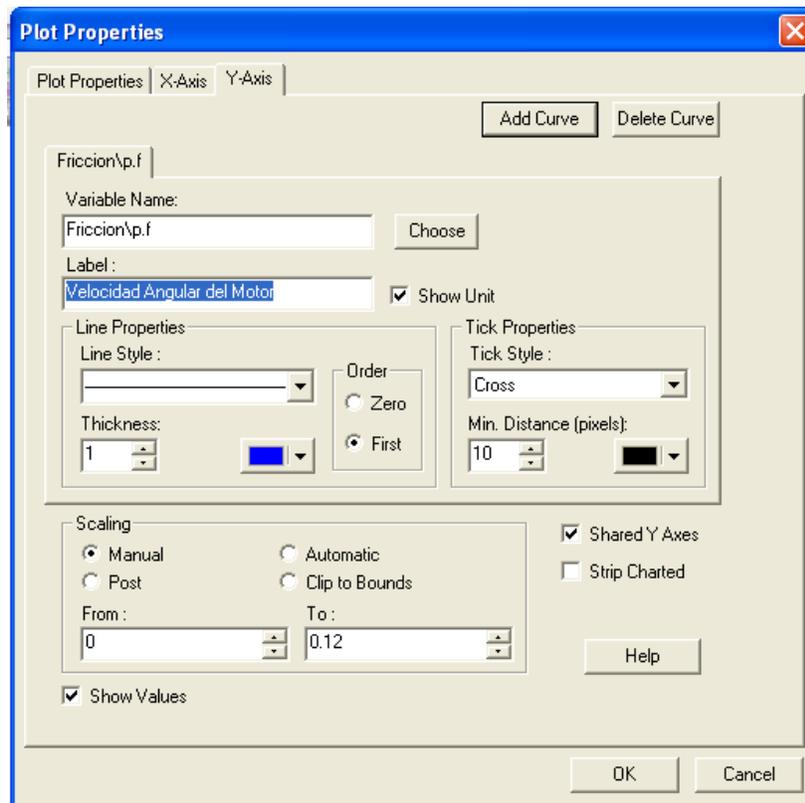


Figura 5.116. Editor de propiedades de la gráfica.

11. Cierre del editor *Plot Properties* haciendo clic en **OK**.
12. En el menú *Simulation*, haga clic sobre el comando **Run** para iniciar la simulación.
El resultado de la simulación debe ser similar a la Figura 5.117.

Después de obtener esta primera gráfica (Figura 5.117), se podrá repetir el procedimiento para obtener cada una de las modelaciones que se deseen del sistema a partir del paso 8, en cual se puede elegir otras variables para simular y observar su comportamiento a través de las gráficas que arroja el software. De esta forma se puede graficar otra variable de interés: el par del motor. En este nuevo caso, se selecciona la variable *Fricción\p.e* la cual representa el par del motor (τ) y se pueden hacer las modificaciones en los límites del eje X y el eje Y para una mejor visualización del resultado de la simulación (ver Figura 5.118).

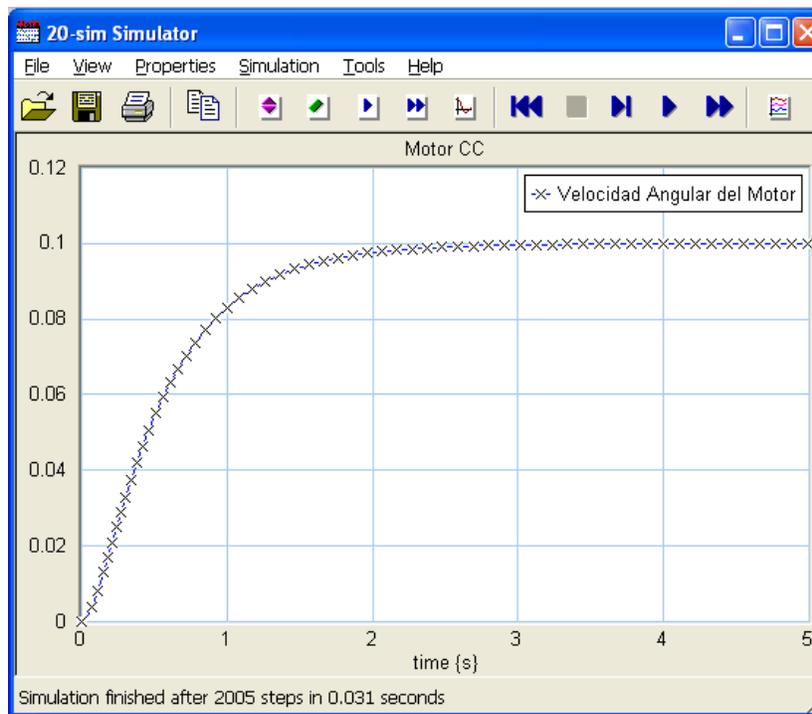


Figura 5.117. Gráfica de la Velocidad Angular del Motor.

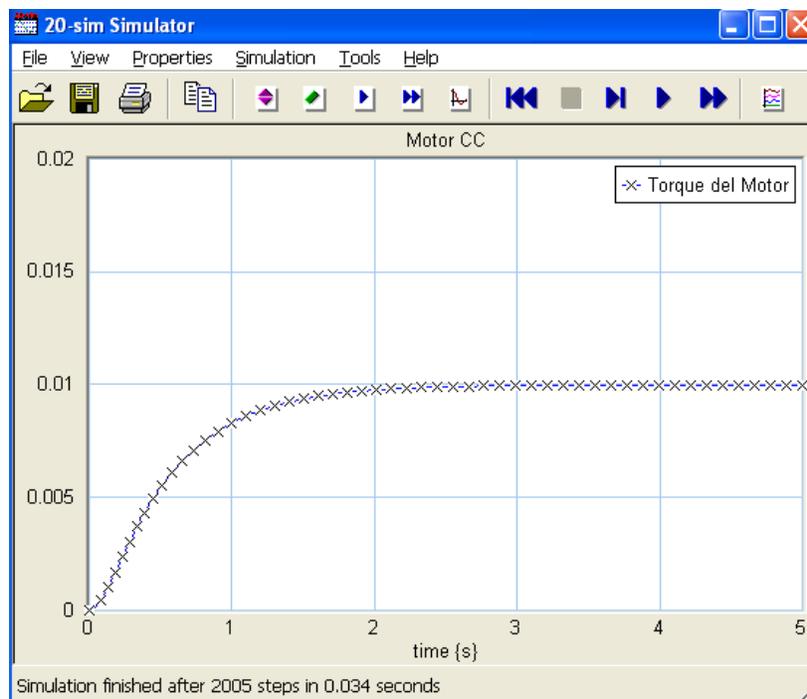


Figura 5.118. Gráfica del Par del Motor.

13. Para guardar haga clic en **File**, luego **Save**.



6.1 20-SIM VERSIÓN 4.3.0

En noviembre de 2012, se lanza al mercado la versión 4.3.0 del software 20-sim por la compañía Controllab Products para su venta. Tiene algunas características nuevas muy interesantes las cuales se describen a continuación [21].

6.1.1 INTRODUCCIÓN

El software 20-sim es un programa que permite modelar sistemas dinámicos y simular su comportamiento este se ejecuta bajo Microsoft Windows. Permite simular el comportamiento de sistemas dinámicos, tales como sistemas eléctricos, mecánicos e hidráulicos o cualquier combinación de estos, separando los diferentes dominios tecnológicos en submodelos independientes. En la actualidad se desarrolla comercialmente y es distribuido por Controllab Products [21].

20-sim proporciona una extensa librería la cual le permite crear modelos de una forma muy rápida e intuitiva, simularlos y analizar su rendimiento.

6.1.2 VERSIONES DEL 20-SIM 4.3.0

20-sim está disponible en tres versiones: Visor, Standard y Professional.

1. **Visor / versión de demostración:** Esta es una versión gratuita que le permite cargar y ejecutar modelos y evaluar el paquete. Guardar modelos no es posible en esta versión.
2. **Estándar:** Esta versión estándar de 20-sim. No tiene limitaciones pero sólo incluye la toolboxes del dominio de la frecuencia.
3. **Profesional:** Esta es la versión completa de 20-sim con todas las toolboxes.



6.2 REQUERIMIENTOS DE SOFTWARE Y HARDWARE

- Un PC con una conexión ethernet funcionando.
- La plataforma Windows (XP / Vista / 7)
- 64 MB de memoria
- 160 MB de espacio en disco duro
- CPU compatible x86

6.3 CARACTERÍSTICAS

6.3.1 MODELACIÓN

Los modelos se construyen en el Editor de 20-sim. Este editor contiene una librería de modelos extensa la cual permite construir modelos usando diagramas de bloques, diagramas icónicos, Bond Graph (grafos de enlace) o una combinación de estos. Varias toolboxes le ayudarán a crear modelos especiales:

- Cuadro de controles
- Mecatrónica Toolbox
- 3D Mechanics Toolbox

6.3.2 SIMULACIÓN

En el simulador se pueden ejecutar modelos y mostrar los resultados en gráficos y animaciones 3D [21].

6.3.3 ANÁLISIS

Se puede analizar los modelos usando la toolbox en el dominio del tiempo y la toolbox en el dominio de la frecuencia [21].



6.3.4 EXPORTACIÓN

Con la toolbox de generación de código puede exportar modelos en código C para su uso en 20-sim 4C, Matlab, Simulink y otros paquetes [21].

6.4 TOOLBOXES DEL SOFTWARE 20-SIM 4.3.0

6.4.1 TOOLBOX DOMINIO DEL TIEMPO

La toolbox dominio del tiempo le permite utilizar barridos de parámetros, optimización, ajuste de curvas para investigar y mejorar los sistemas.

6.4.2 TOOLBOX DEL DOMINIO DE LA FRECUENCIA

Con la toolbox de dominio de frecuencia se puede linealizar modelos de espacio de estado, crear sus propios sistemas lineales y mostrar el comportamiento de la frecuencia con diagramas de Bode y los de FFT [21].

6.4.3 CUADRO DE CONTROLES

La toolbox de control le permite diseñar sistemas de retroalimentación con una planta, un controlador y un pre-filtro lineal. Con un editor de filtros puede crear sus propios filtros lineales de acuerdo con sus especificaciones [21].

6.4.4 TOOLBOX MECÁNICA 3D

El editor de Mecánica 3D le ayuda a ensamblar modelos 3D arrastrando y llevando cuerpos, articulaciones y otros objetos en un espacio de trabajo 3D. El modelo resultante puede combinarse con los modelos de la biblioteca estándar de 20-sim [21].



6.4.5 HERRAMIENTAS DE ANIMACIÓN

Los resultados de la simulación en 20-sim pueden ser mostrados como una película mediante la toolbox de animación 3D. Cualquier variable de un modelo de 20-sim puede conectar a 3 objetos tridimensionales para mostrar la producción de animación.

6.4.6 TOOLBOX MECATRÓNICA

La toolbox Mecatrónica contiene el Asistente para perfiles de movimiento, el Asistente de Cámara y el Editor de Servo Motor [21].

6.5 NOVEDADES DEL LA VERSION 4.3.0

- La interfaz de 20-Sim se cambio para reflejar la nueva identidad corporativa, hay un nuevo logo.
- La biblioteca está incorporada dentro de la ventana del editor.
- Se puede agregar una imagen de fondo a los modelos gráficos y mostrar esta imagen durante la impresión y generación de documentos. Esto le permite generar fácilmente la documentación impresa de sus modelos, incluyendo sus números de documentación de la empresa del proyecto, fecha, datos de autor etc.
- Mejoras en la mecánica 3D.
- Autoguardado para guardar experimentos de simulación.
- Una función que permite chequear actualizaciones en línea.
- Auto-dimensionamiento de los iconos para ver modelos por defecto.
- Tiene menos iconos pero más avanzados.
- Se hace posible mover gráficos y curvas entre las ventanas del simulador con el menú derecho del ratón, así como también tiene funciones para análisis de datos.
- Arrastrar y soltar gráficos y curvas en una ventana.
- Todas las ventanas del simulador están enumeradas en el panel izquierdo.
- La opción de herramientas se agrega a la barra de menú de todas las ventanas del grafico.

6.6 DESARROLLO DE UN CIRCUITO RLC EN EL PROGRAMA 20-SIM 4.3.0

En la Figura 6.1 se presenta el circuito eléctrico Resistencia-Inductor-Capacitor (RLC) y en la Figura 6.2 se tiene el grafo de enlace (Bond Graph) de dicho sistema [21].

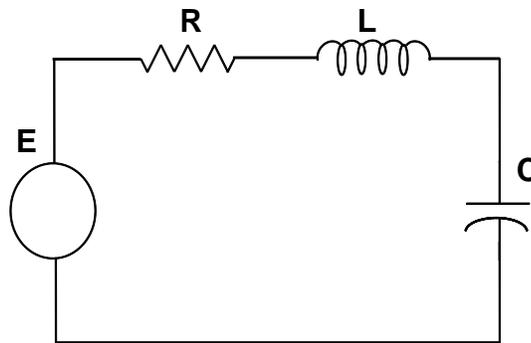


Figura 6.1. Circuito RLC.

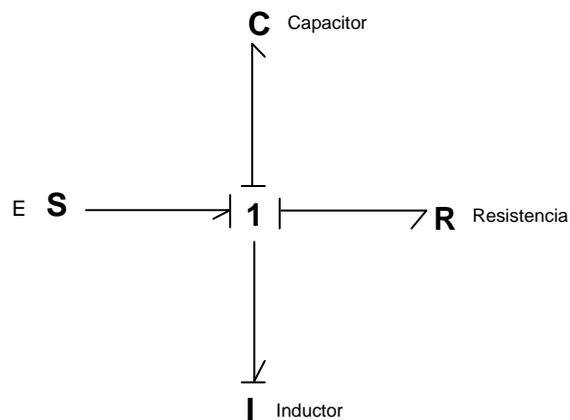


Figura 6.2. Bond Graph del Circuito RLC.

1. Abra 20-SIM y en el menú **File** seleccione **New** y luego **Graphical Model**.

Se abra el editor del 20-SIM, asegúrese de que 20-SIM este en modo depuración.

Esto se indica en la barra de herramientas con el botón **Mode** .

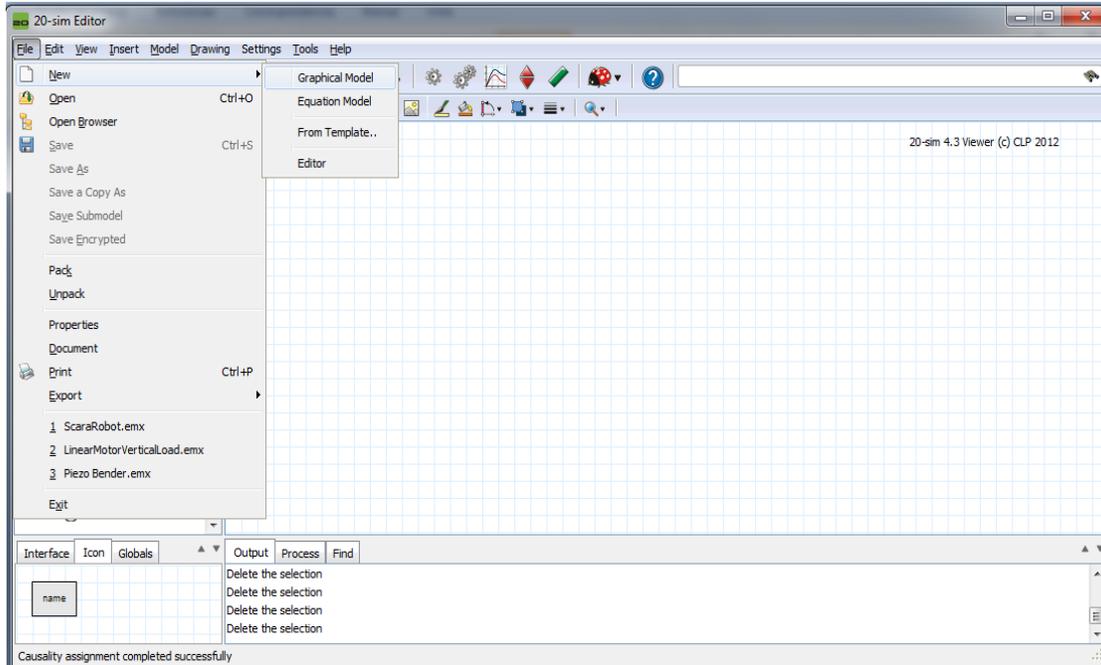


Figura 6.3. Selección del Espacio de Trabajo.

2. Si el 20-SIM está en modo rápido indicado por este botón  , haga clic hasta que cambie al modo depuración  .

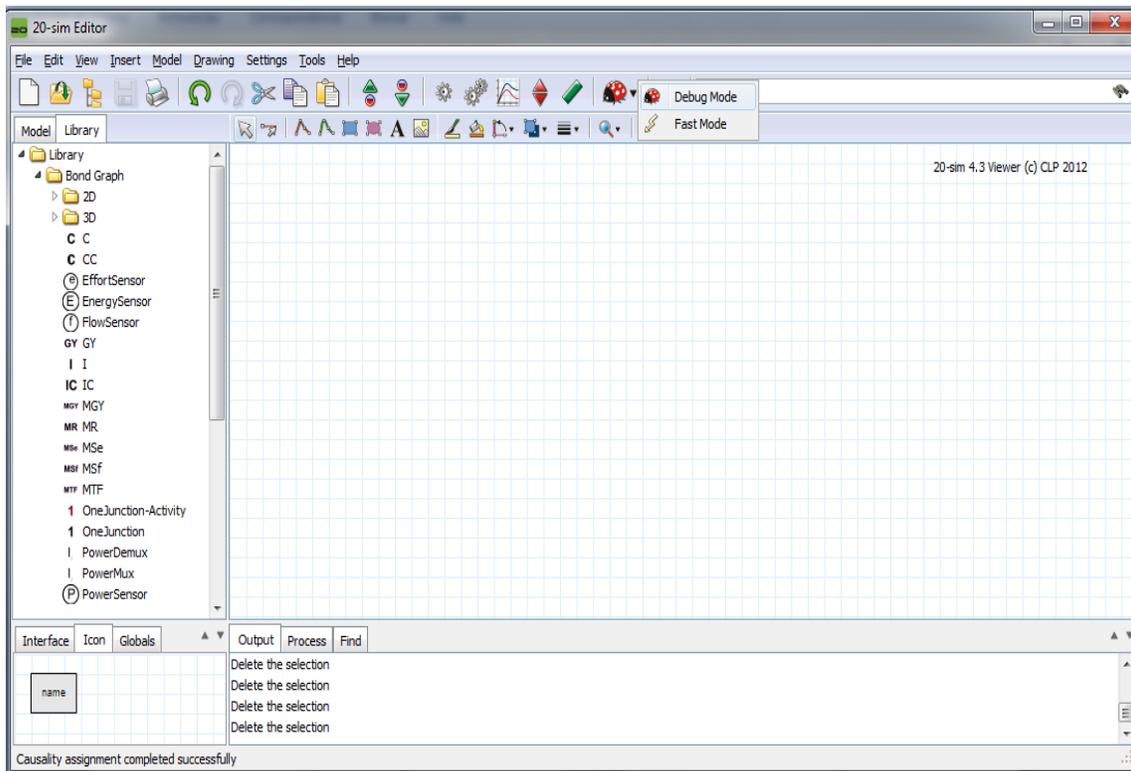


Figura 6.4. Selección del Botón Modo Depuración.



3. En la barra de herramientas del lado izquierdo del programa seleccione *library* se desplegaran varias carpetas, seleccionar la carpeta de *Bond Graph* la cual contiene los elementos que se requieren para construir el modelo.
4. En el menú del botón derecho del ratón o desde el menú *Insert*, seleccione *Knot* y después *OneJunction*. Se añade la unión 1, ver Figuras 6.7.

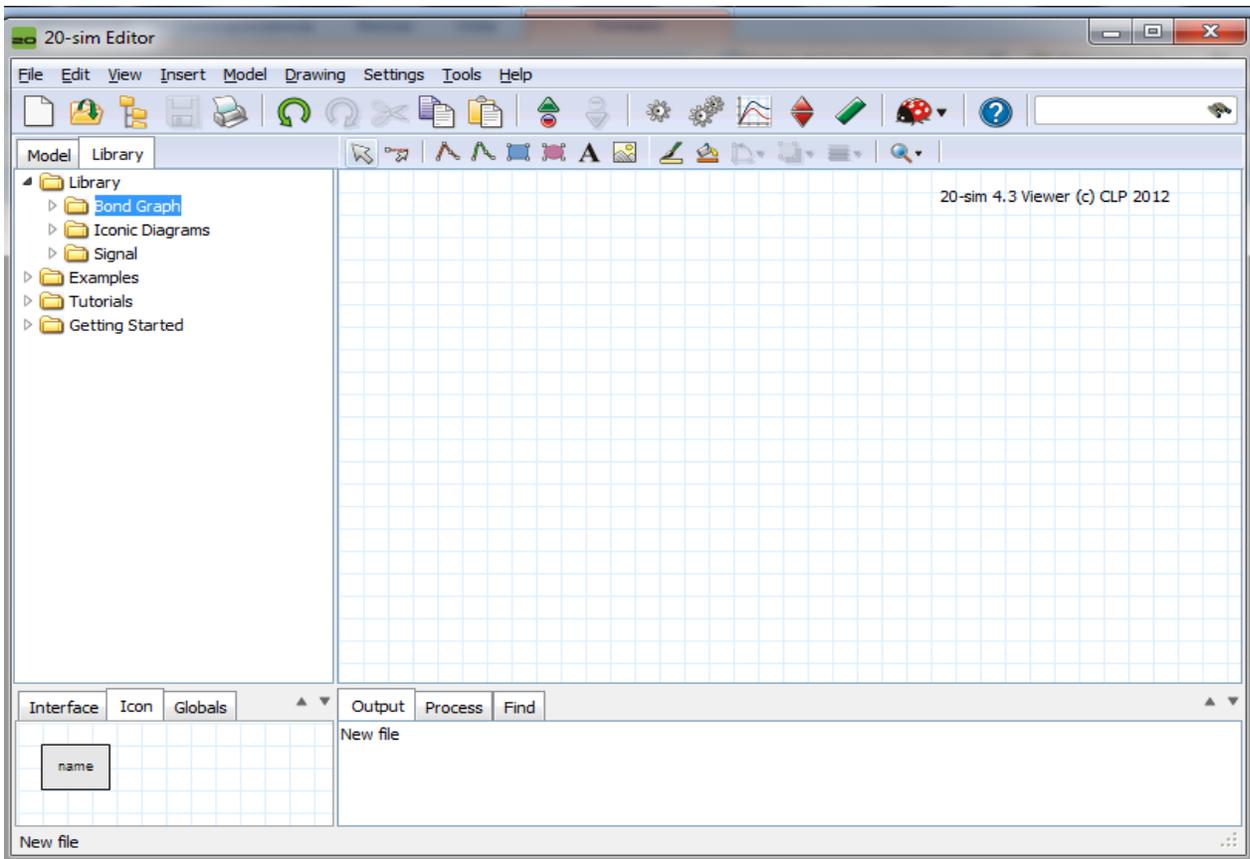


Figura 6.5. Selección de Librería de Bond Graph.

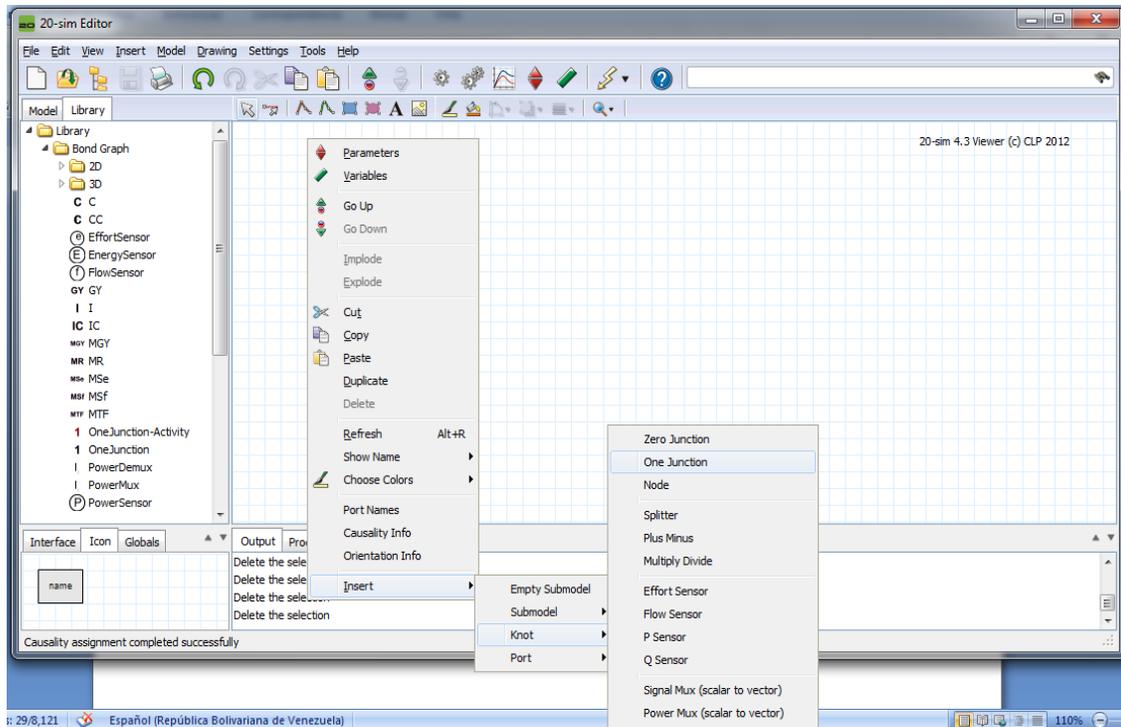


Figura 6.6. Selección de la Unión-1.

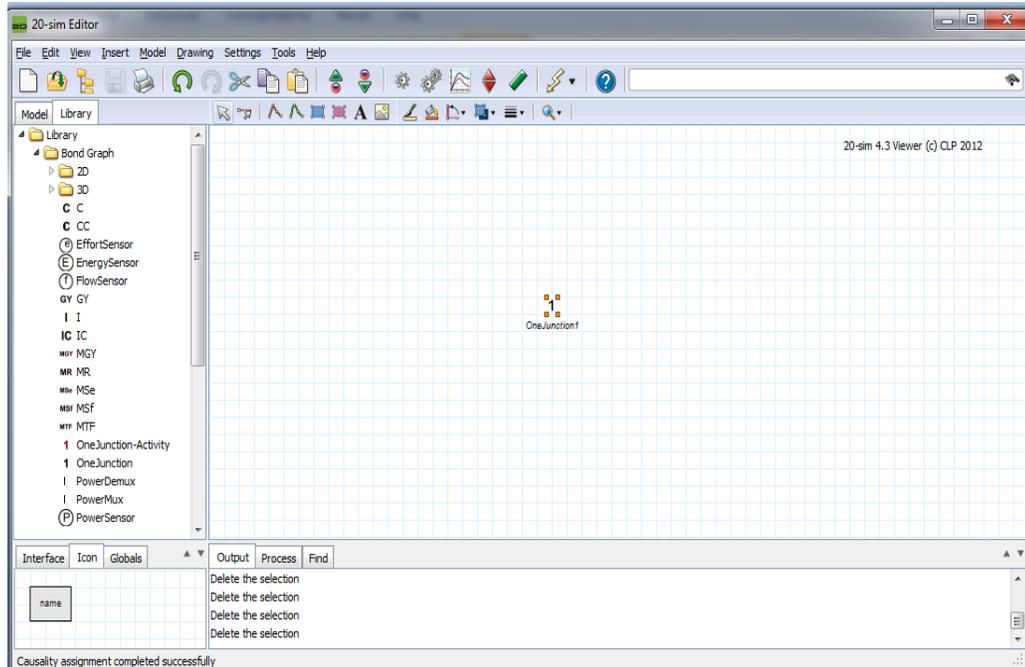


Figura 6.7. Inserción de la Unión-1.

5. Seleccionar el elemento *Se* de la librería de Bond Graph e insertarlo en la ventana de implementación.

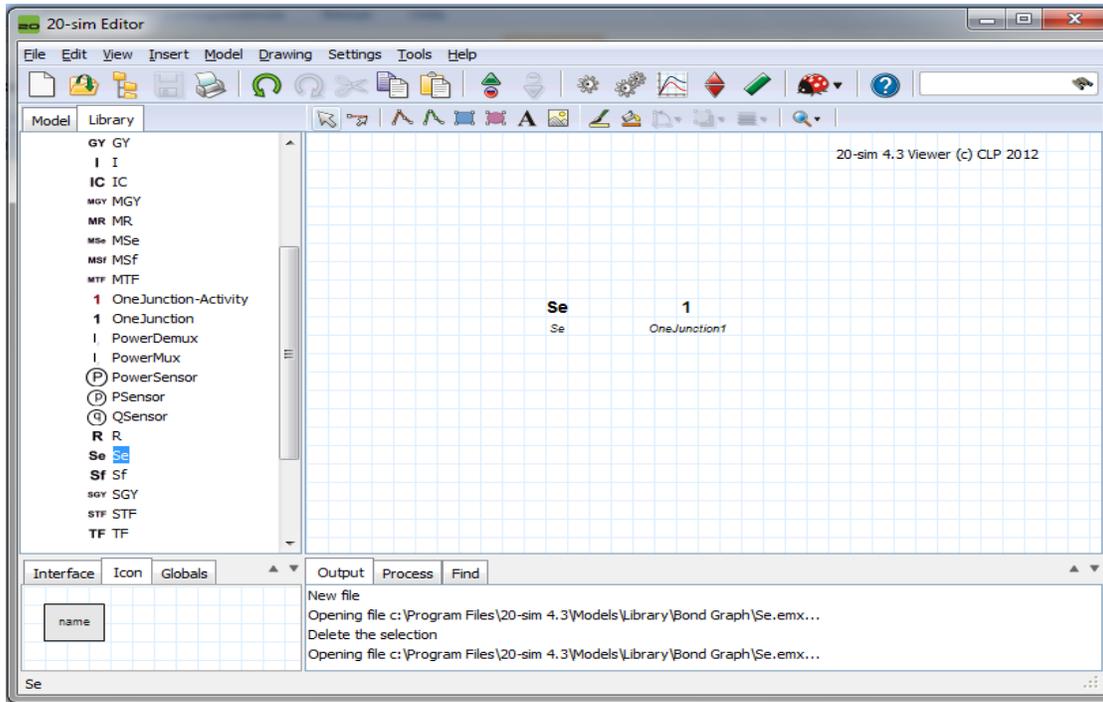


Figura 6.8. Inserción del elemento Se.

6. Seleccionar el resto de los elementos que conforman el modelo como se muestra en la Figura 6.9.

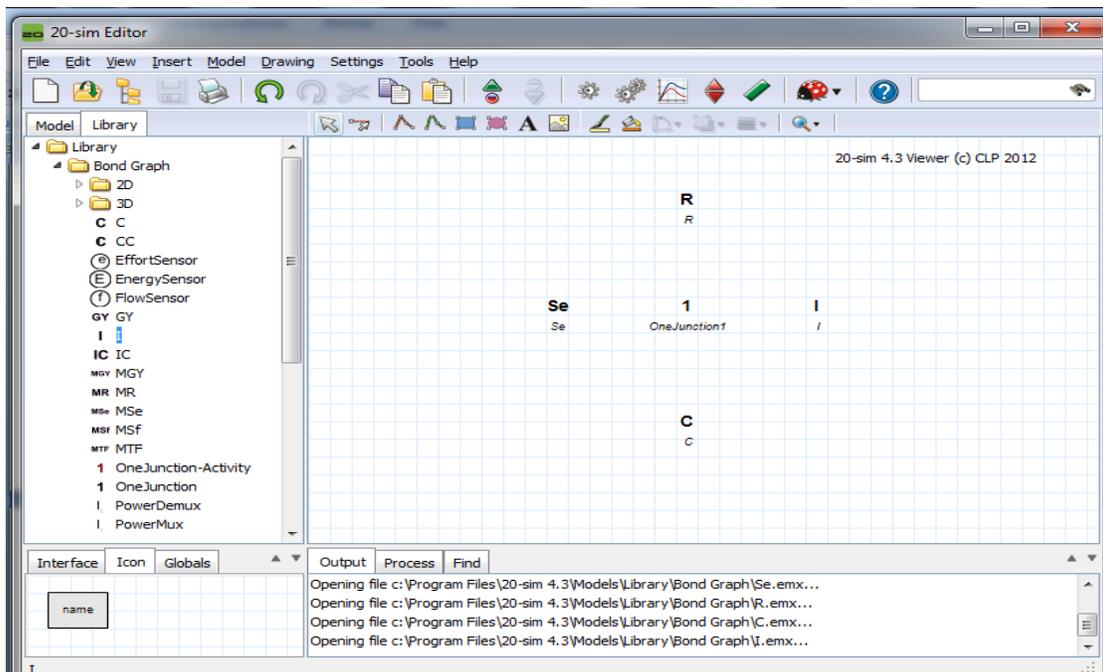


Figura 6.9. Inserción de elementos del modelo.

7. En la barra de herramientas haga clic en el botón  para colocar las conexiones.
8. Seleccione el elemento Se (clic con el ratón en el botón izquierdo sobre el elemento) y luego clic sobre la unión 1 para crear el primer enlace. Ver figura 6.10.
9. Proceda a crear los enlaces faltantes del modelo hasta completarlos y el modelo debe verse como el mostrado en la Figura 6.11.

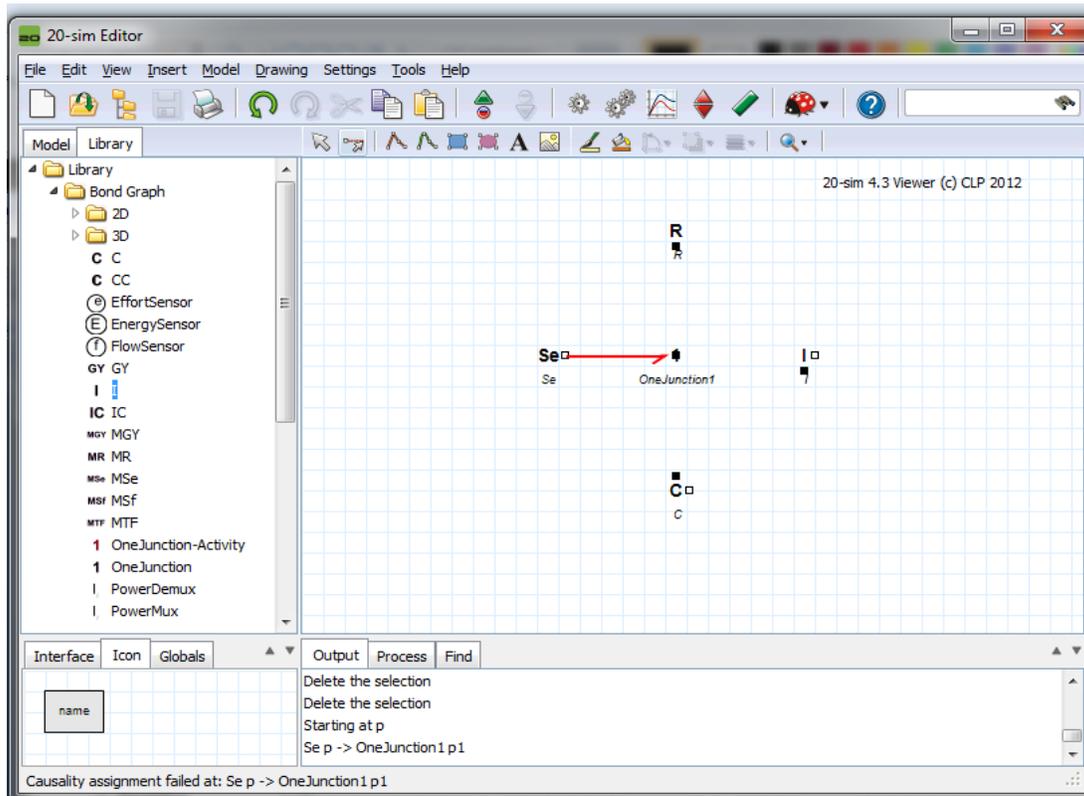


Figura 6.10. Enlace entre el elemento Se y la Union-1.

Ahora un enlace es visible en nuestro modelo, este es de color rojo lo que significa que la causalidad no ha sido asignada.

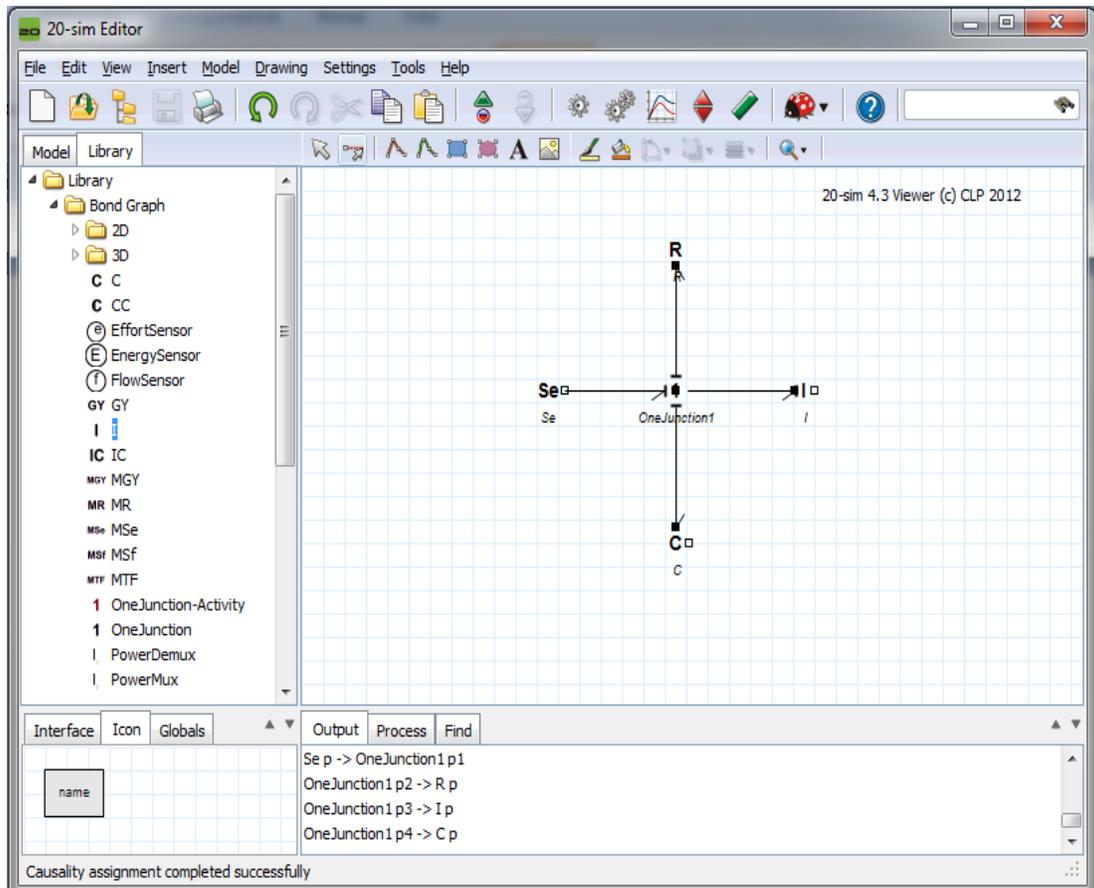


Figura 6.11. Modelo en Bond Graph del circuito RLC.

La Figura 6.11 muestra el modelo en Bond Graph del circuito RLC, el 20-Sim tiene la capacidad de asignarle la capacidad a cada elemento de forma automática.

10. En la barra de herramientas haga clic sobre el icono  el cual tiene la función de hacer la comprobación del modelo que se está ejecutando.

Si el modelo esta correcto aparecerá en la parte inferior del editor una subventana llamada **process** la cual mostrara el número de submodelos, numero de ecuaciones, numero de variables etc del sistema. Como se muestra en la Figura 6.12.

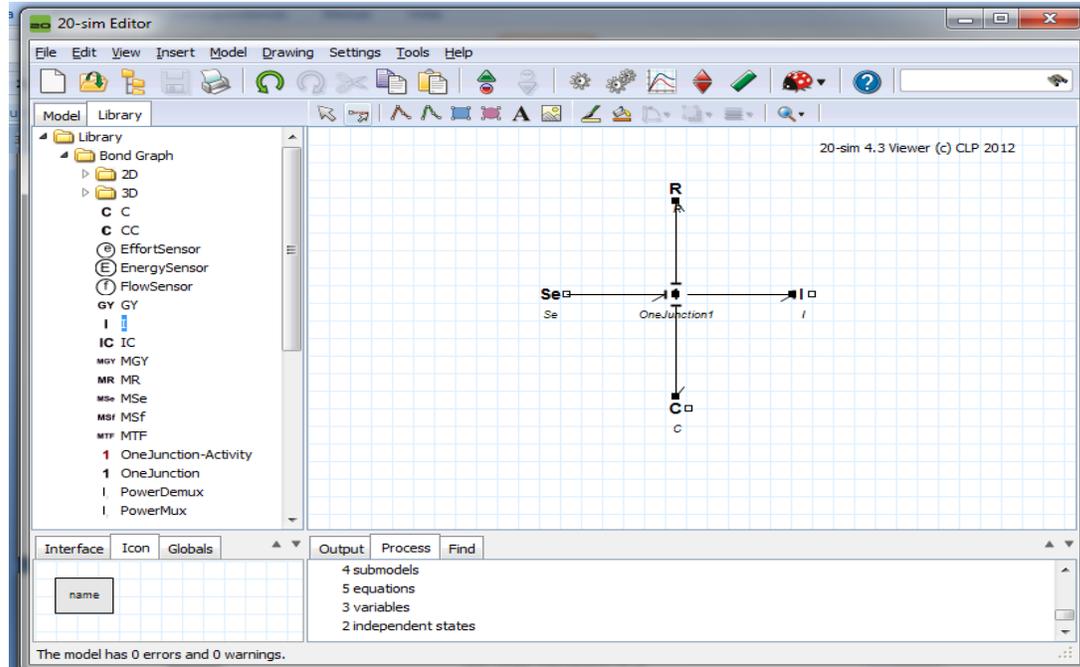


Figura 6.12. Subventana Process.

Si se produce algún error se debe corregir inmediatamente para continuar con el modelo.

11. En la barra de herramientas del editor seleccione el icono *Start Simulation* , esta acción despliega la ventana del simulador. Ver Figura 6.13

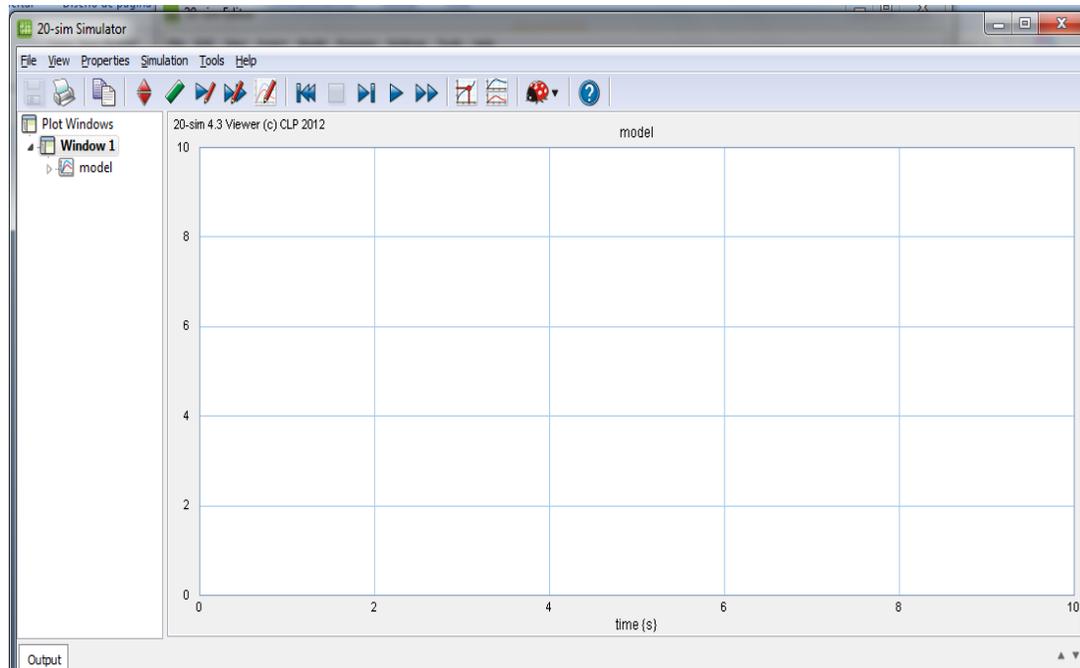


Figura 6.13. Ventana del simulador del 20-Sim 4.3.0.



Nota: Este modelo no podrá ser guardado ya que la versión de 20-Sim 4.3.0 es un demo el cual no permite guardar los modelos.

12. En la barra de herramientas del simulador haga clic en *Properties* luego seleccione el comando *parameters* y asigne valores a los elementos del modelo.

Elemento	Valor
R	0.67 Ω
L	1 H
C	5 F
Fuente	5 V

Tabla 6.1. Parámetros circuito RLC.

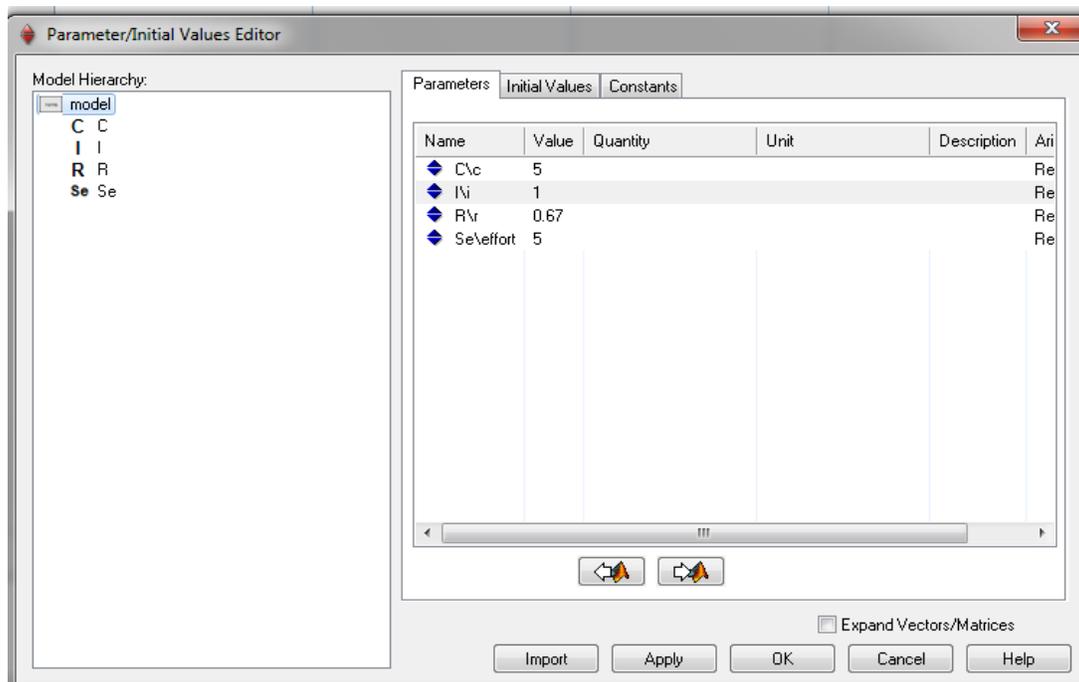


Figura 6.14. Edición de los valores de los Elementos del Modelo.

13. Desde el menú *properties* seleccione el comando **Run** y cambie los valores por defecto a los siguientes parámetros de simulación de la Tabla 6.2, observe la Figura 6.15.

Parámetro	Valor
Start	0
Finish	30
Method	Runge-Kutta 4
Step Size	0.1

Tabla 6.2. Valores de los Parámetros de Simulación del Modelo.

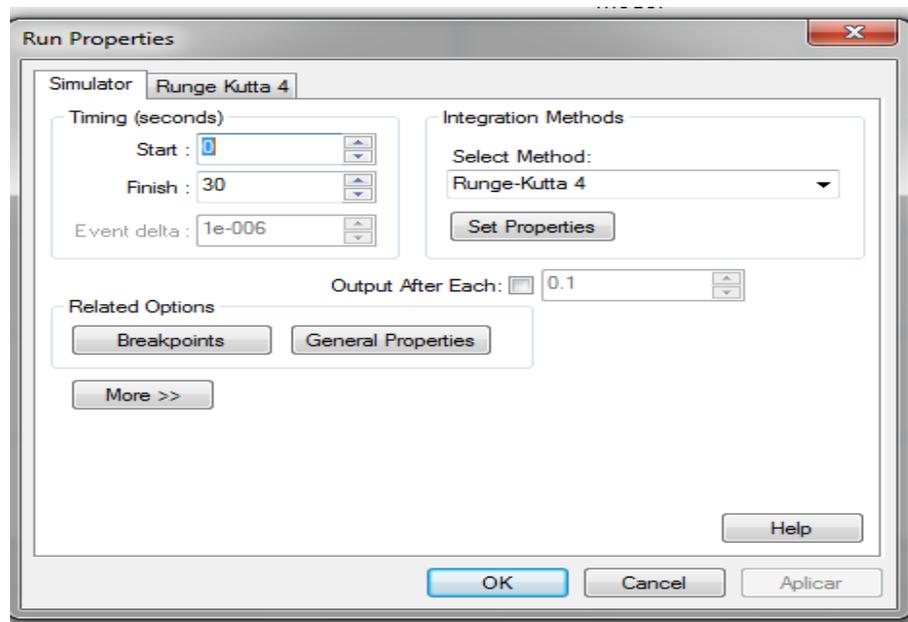


Figura 6.15. Edición de los valores de los parámetros (tiempo) de simulación modelo.

14. De la ventana del simulador seleccione el botón  y se desplegará la subventana *plot properties*, seleccione la pestaña **Y-axis** y haga clic en **Choose** para elegir la variable a simular. En este caso se elige la variable **I1\p.f**, la cual es la

corriente del inductor y también la corriente del circuito. Presione con el puntero sobre el submodelo **I1** a la izquierda, luego **p.f** a la derecha y finalmente presione **OK**. (Ver Figura 6.16 y 6.17).

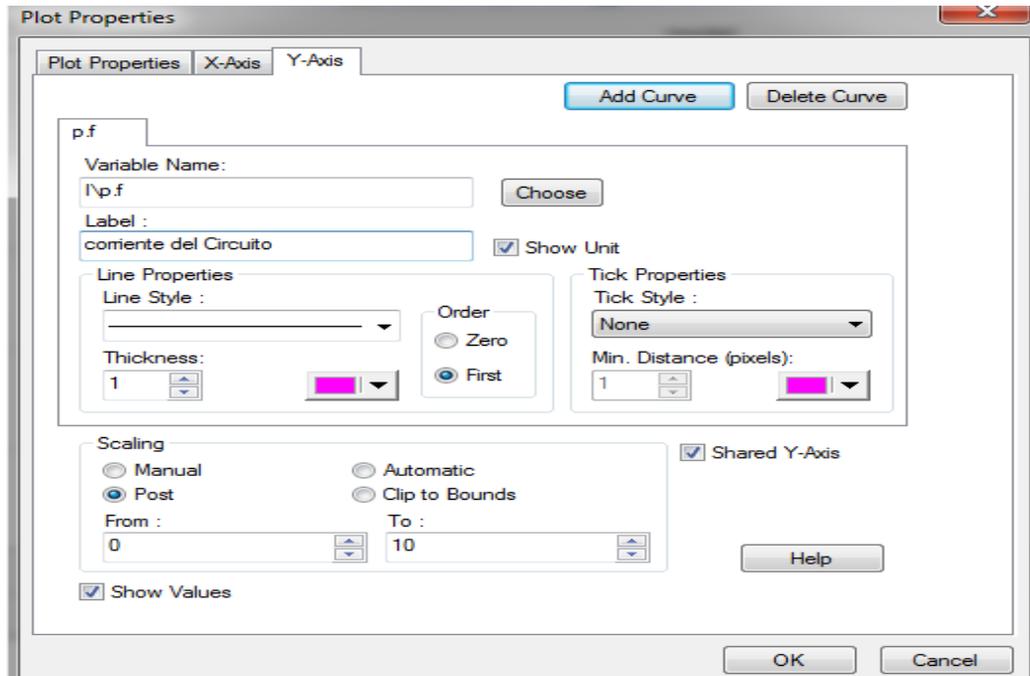


Figura 6.16. Subventana *plot properties*.

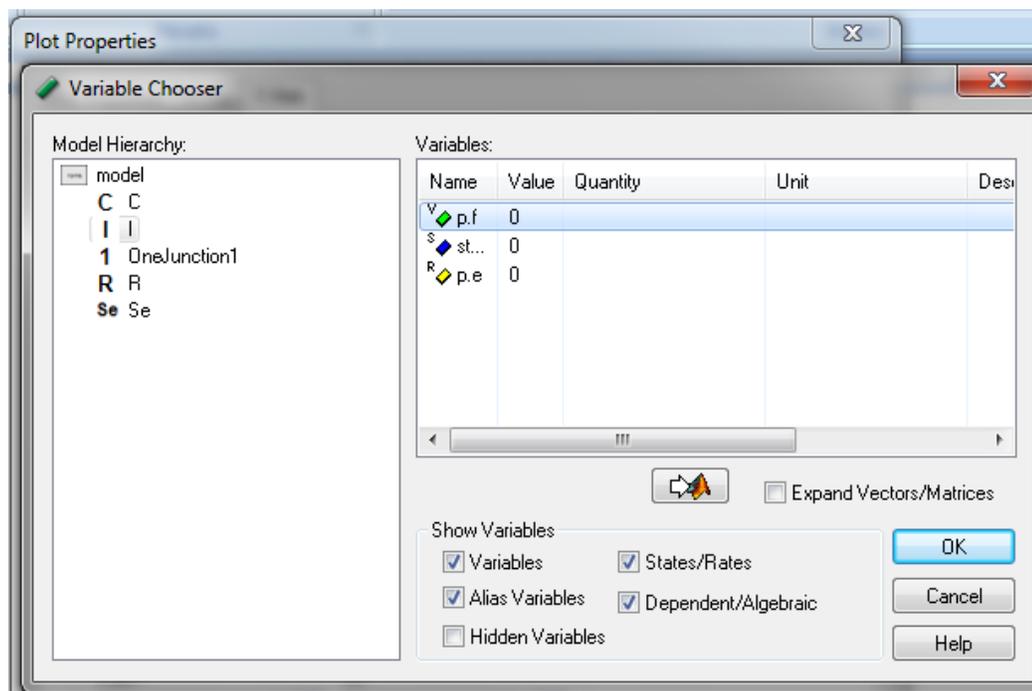


Figura 6.17. Elección de la Variable a Simular.



15. En el editor de *plot properties* establezca los siguientes valores que se muestran en la tabla 6.3.

Tick Properties	
Tick Style	None
Color	Azul
Scaling	
Scaling	Manual
From	-1
To	5

Tabla 6.3. Valores de los Parámetros de Simulación del Modelo.

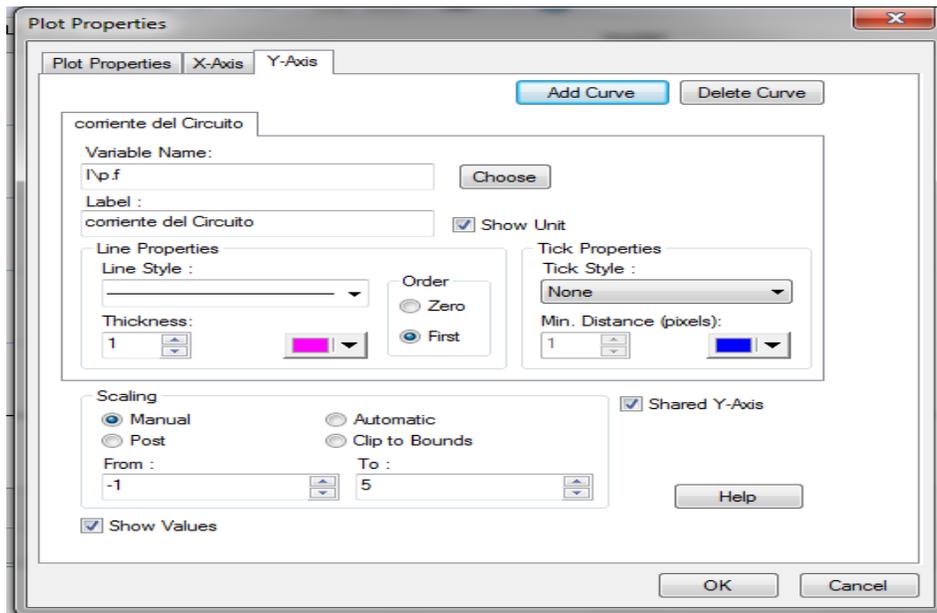


Figura 6.18. Identificación de Variable a Simular y parámetros del Grafico.

16. Cierre el editor *plot properties* haciendo clic en **OK**.

17. Presione el botón  para ejecutar la simulación del experimento. El resultado de la simulación debe ser similar a la Figura 6.19.

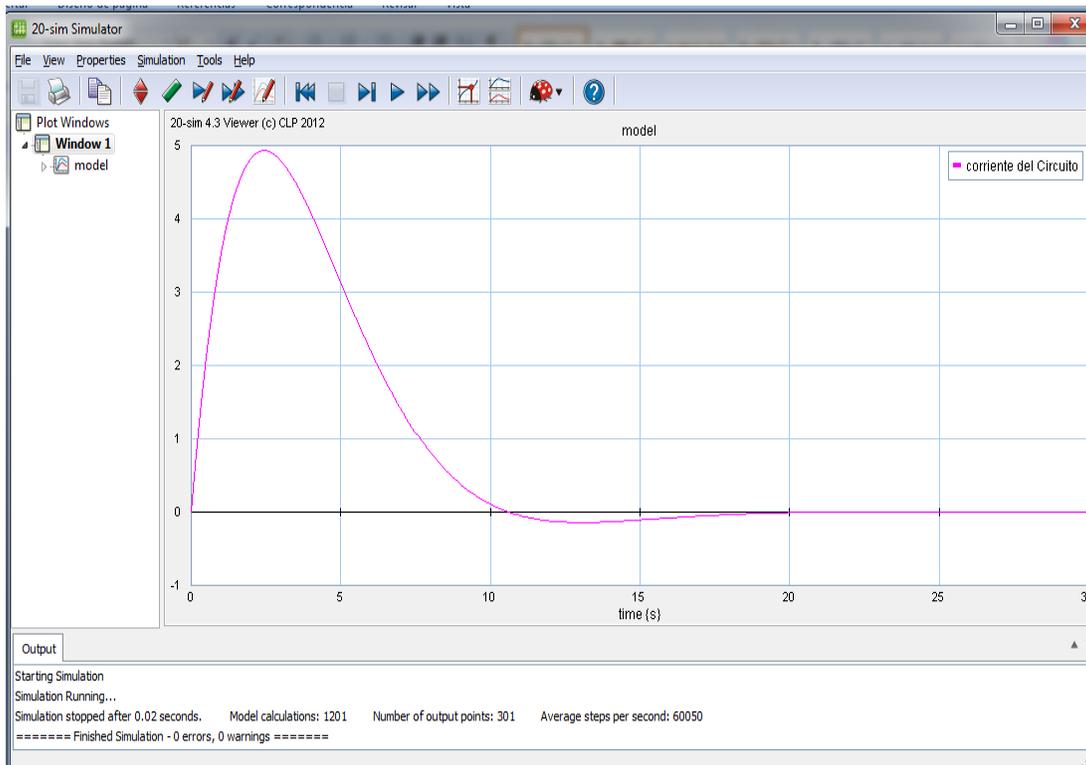


Figura 6.19. Corriente del Circuito RLC.

18. Ahora se grafican los diferentes voltajes de cada uno de los elementos (resistor, capacitor, inductor y fuente). Para llevar a cabo esta acción se modifica la variable a simular en cada caso, como en los pasos 13,14,15 y 16. En la siguiente Tabla 6.4 se indican las nuevas variables.

Elemento	Variable	Variable en 20-sim	Label
Resistor	Voltaje	R1\p.e	Voltaje del Resistor
Capacitor	Voltaje	C1\p.e	Voltaje del Capacitor
Inductor	Voltaje	I1\p.e	Voltaje del Inductor
Fuente	Voltaje	Se1\p.e	Voltaje de la Fuente

Tabla 6.4. Asignación de Variables Faltantes a Simular.



En las siguientes Figuras (6.20, 6.21, 6.22, 6.23) se muestran las gráficas de los diferentes voltajes de los elementos del circuito eléctrico.

Es posible cambiar las especificaciones de las gráficas según las preferencias del usuario, se incentiva a realizar diferentes cambios en el color de la gráfica, límites del eje Y, tipo de línea de la grafica, tipo de marca, los límites de tiempo de simulación (eje X) y tipo de método numérico a utilizar, etc. Cada una de las características anteriores hacen del software 20-sim un programa de modelación y simulación versátil, explícita, potente (respecto a los métodos de integración que contiene) y didáctica.

El uso del software 20-sim, desde el punto de vista académico, es una herramienta práctica que amplía el conocimiento del estudiante de ingeniería y carreras afines, en cuanto al rango de métodos aplicables en la resolución de problemas y estudio de sistemas físicos de diferentes dominios y combinaciones de los mismos.

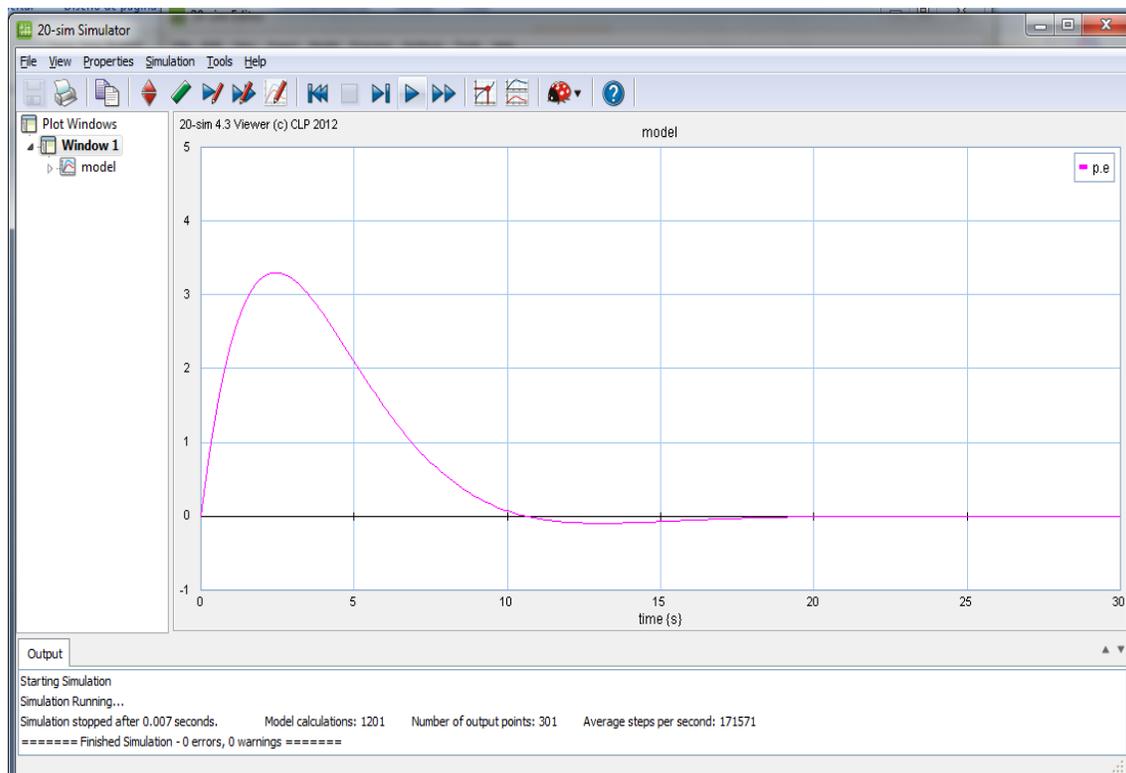


Figura 6.20. Voltaje del Resistor.

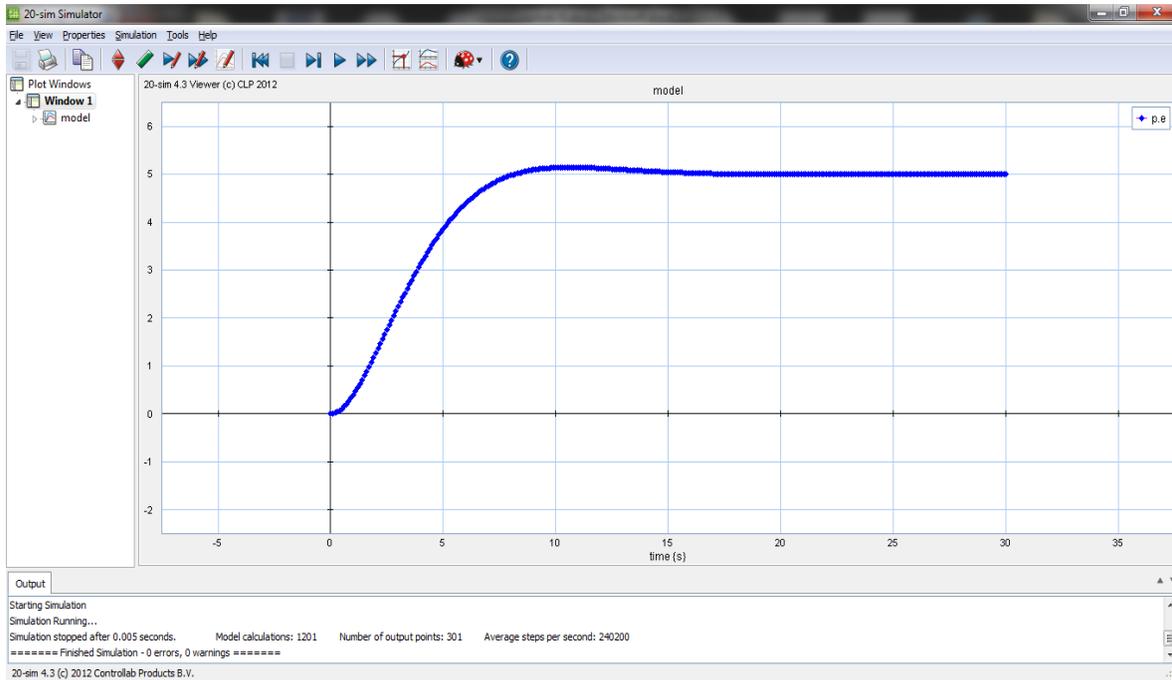


Figura 6.21. Voltaje del Capacitor.

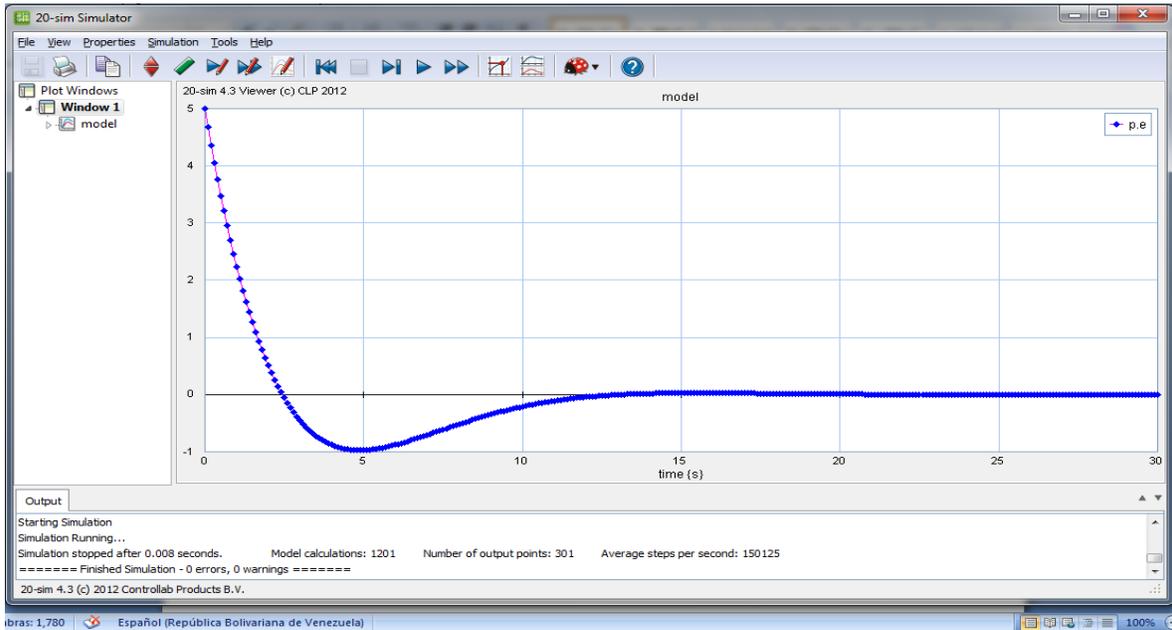


Figura 6.22. Voltaje del Inductor.

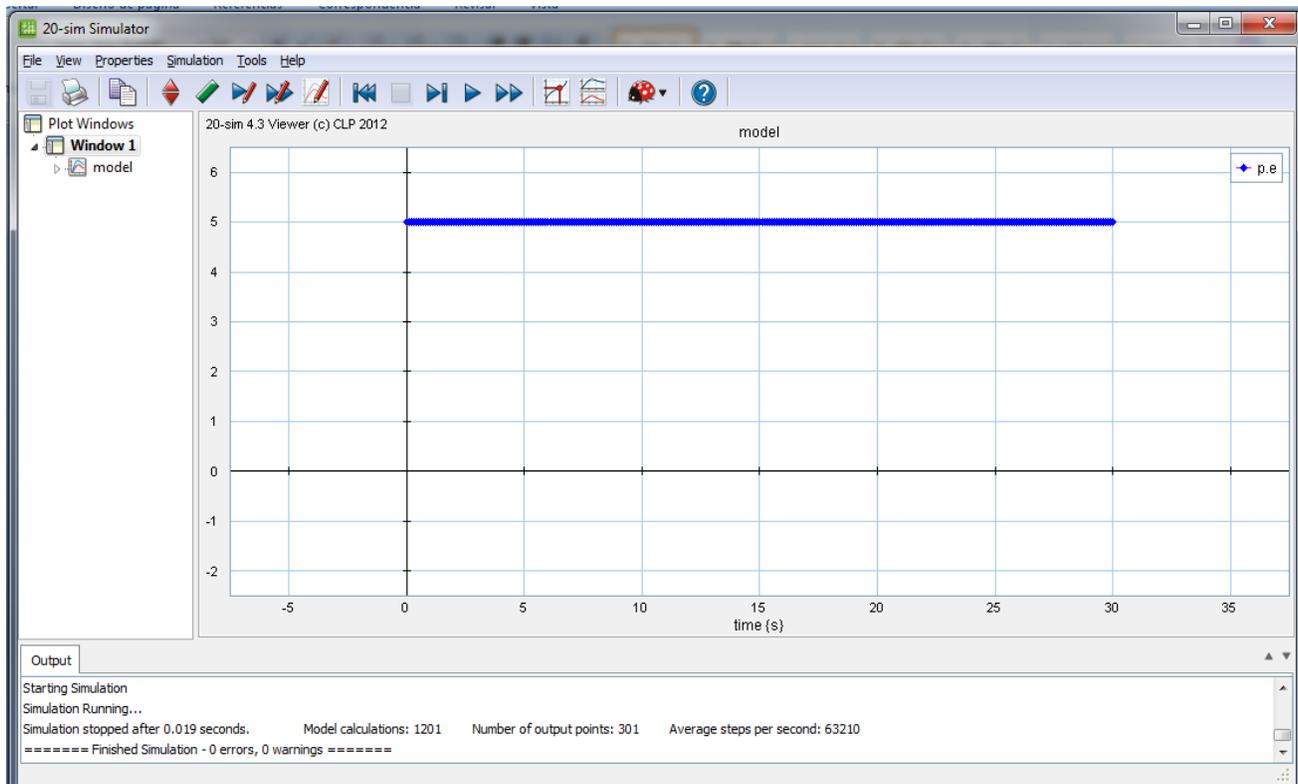


Figura 6.23. Voltaje de la Fuente.



MANUAL PARA EL USUARIO DEL SOFTWARE DIDÁCTICO

7.1. INTRODUCCIÓN

Este capítulo se dedica a la descripción del software didáctico (sitio web) para la enseñanza-aprendizaje de la técnica de Bond Graph y su uso en el software 20-sim para la simulación de sistemas físicos. La elaboración de este sitio web se concreta a través de la utilización del software de programación web Adobe® Dreamweaver® CS4 como recurso principal para el diseño de las páginas que contienen la información de este trabajo de grado, además se contó con las herramientas de animación web Adobe® Flash® CS4 para la creación del menú interactivo y Adobe® Fireworks® CS4 para la edición de imágenes, creación de los encabezados de las páginas y los botones de navegación.

7.2. REQUERIMIENTOS DEL SISTEMA PARA UNA VISUALIZACIÓN ÓPTIMA DEL SITIO WEB

Se debe disponer de la versión 23.0.1271.64 (o una versión posterior) del explorador de internet Google Chrome, el cual es un navegador web creado para mejorar la velocidad, la facilidad de uso y la seguridad y a través del mismo se permite la visualización óptima del sitio web.

7.3 ORGANIZACIÓN DEL SITIO WEB

La estructura del sitio web está organizada de forma jerárquica, como se explica en el esquema de la Figura 7.1, con una página de inicio y una página del menú principal de cuyos enlaces proceden todas las páginas que desarrollan la información de este trabajo de grado. Esta estructura permite crear páginas por niveles, de manera que existe una principal de la que cuelgan las subordinadas, lo cual admite la posibilidad de acceder a las páginas que desee el visitante sin necesidad de recorrer todo el sitio web.

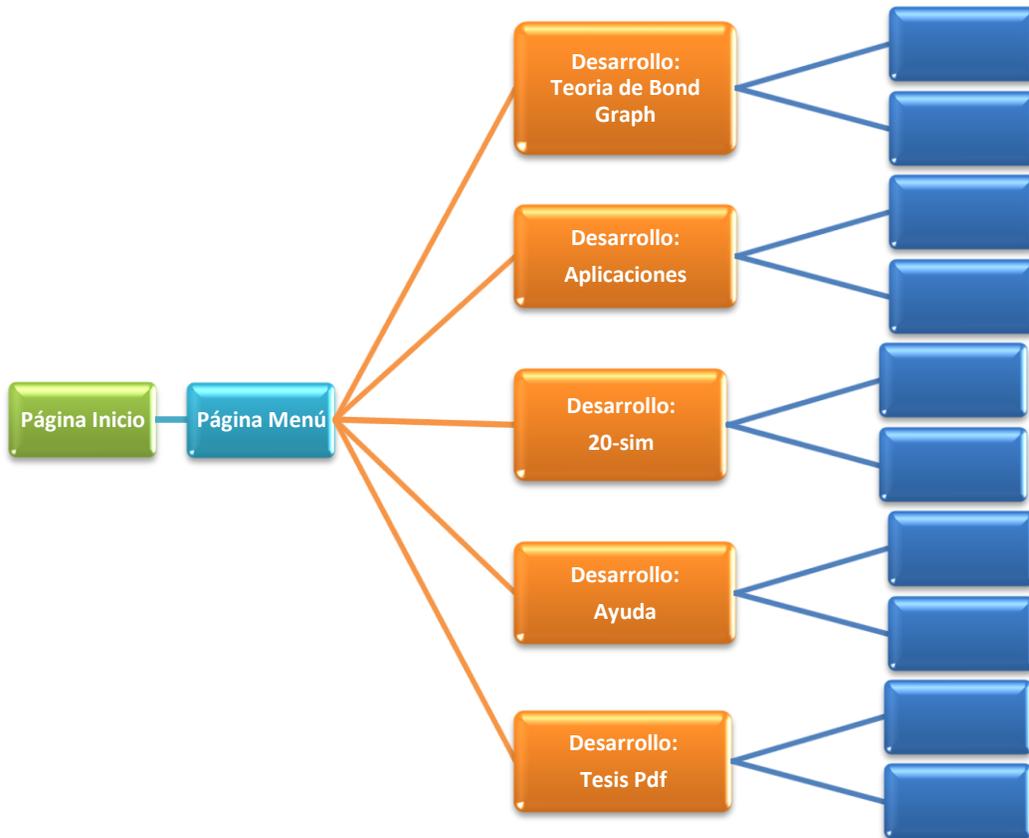


Figura 7.1. Estructura del software didáctico.

En la Figura 7.1 los cuadros azules a la derecha sin texto informativo, indican que existen páginas subordinadas que desarrollan el contenido de cada cuadro naranja.

- **Página Inicio:** contiene el título del trabajo de grado, los autores, tutor y el botón de acceso al menú principal. Ver Figura 7.2.
- **Página Menú:** contiene un cuadro interactivo que mediante imágenes asociadas al contenido, enlaza con las páginas del sitio web que desarrollan la tesis. Ver Figura 7.3.
- **Página Desarrollo:** contiene la teoría, ejercicios, aplicaciones y manuales desarrollados en la tesis, según la clasificación dada por los autores para la mejor captación y comprensión de la información presentada (ver Figura 7.4), cuya exploración y/o navegación se puede hacer de dos formas: a través de la Barra de



Navegación o por el uso de los Botones de Navegación, como se explica en la sección 7.4.



Figura 7.2. Página inicio.

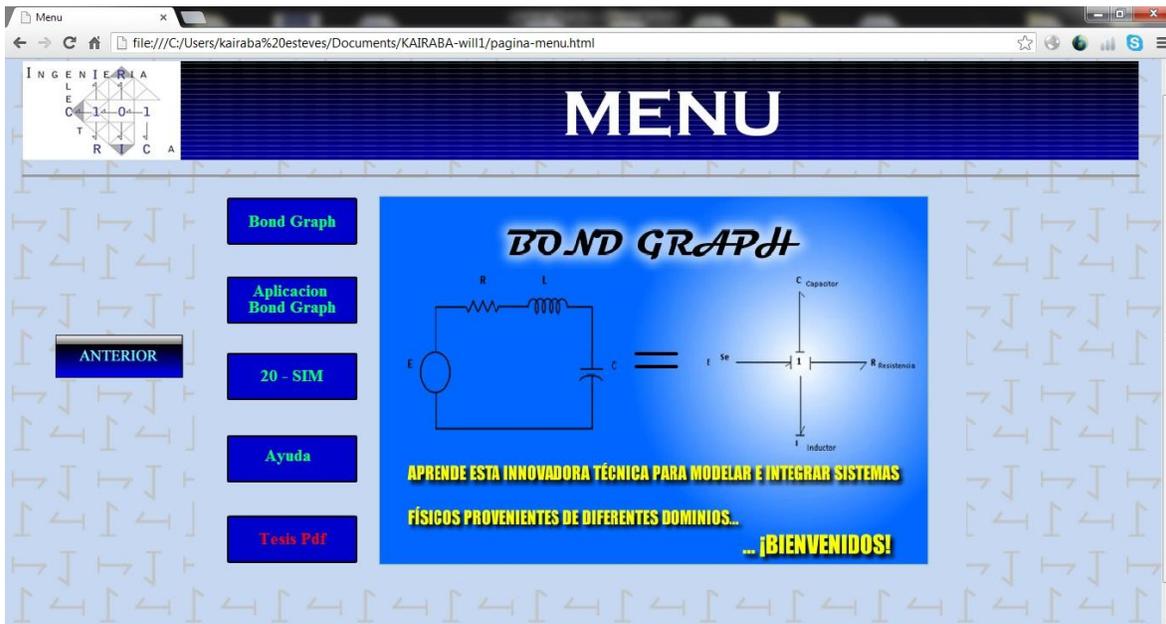


Figura 7.3. Página menú.

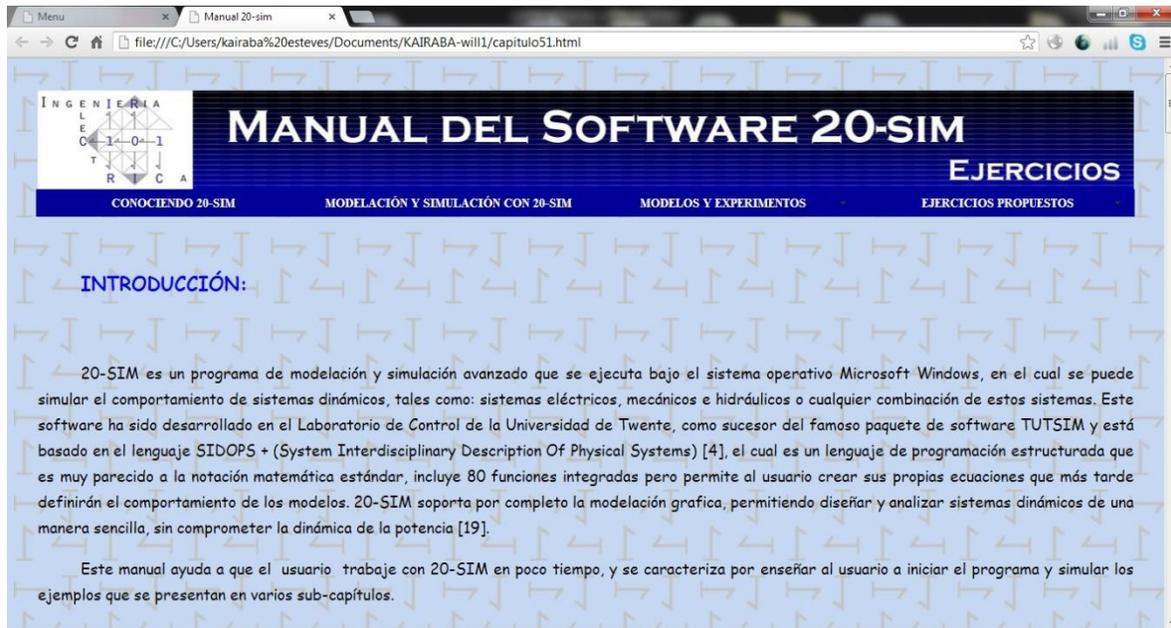


Figura 7.4. Página de desarrollo.

7.4. ELEMENTOS DE NAVEGACIÓN DEL SITIO WEB

Para la navegación del sitio web se dispone de dos herramientas, la Barra de Navegación y los Botones de Navegación.

- **Barra de Navegación:** son enlaces a otras páginas con el mismo nivel jerárquico desde la página donde el visitante se encuentra ubicado, éstos tienen como finalidad, facilitar y agilizar el recorrido del usuario en el sitio web y contiene los subtítulos distribuidos y ordenados de la información según la página de desarrollo donde se encuentra. La barra de navegación se encuentra ubicada debajo del encabezado de todas las páginas de desarrollo. Este medio fue creado a través de la herramienta Barra de menús de Spry, recurso del programa Dreamweaver®. Ver figura 7.5.
- **Botones de Navegación:** son enlaces que guían al visitante a través de las páginas del mismo y de diferente nivel jerárquico. En la tabla 7.1 se muestran los botones

utilizados y sus funciones. Los botones de navegación se encuentran en todas las páginas de desarrollo dos (2) veces, al tope de la página y al final del texto de las mismas, y fueron elaborados a través del programa Fireworks®.

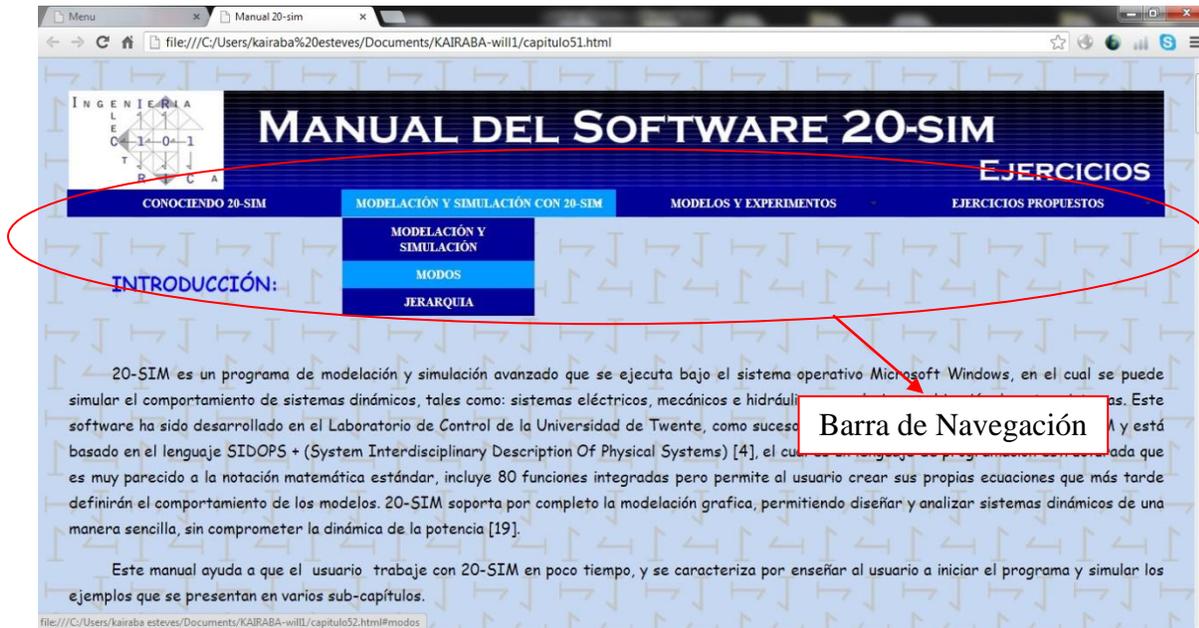


Figura 7.5. Barra de navegación.

BOTÓN	FUNCIÓN
	Conduce a la página anterior de la página actual en la que se encuentra el botón.
	Conduce a la página siguiente de la página actual en la que se encuentra el botón.
	Conduce a la página menú principal.
	Conduce a la página de ayuda al usuario del software didáctico.
	Conduce a la portada del software didáctico.

Tabla 7.1. Tabla de botones de navegación.



7.5. EXPLORACIÓN DE LAS PÁGINAS DEL MENÚ

Esta sección presenta un breve repaso de la Página Menú cuyas páginas subordinadas presentan el contenido de la información de este trabajo de grado. El objetivo es mostrar al usuario la página principal a la que conducen los botones del menú interactivo.

La Figura 7.3 muestra el menú interactivo. Si se presiona el botón *Bond Graph* (Figura 7.6), se despliega la página de desarrollo *Teoría de Bond Graph, Conceptos y Definiciones* (Figura 7.7) que contiene, como su nombre lo indica, toda la información teórica de la técnica de Bond Graph.



Figura 7.6. Botón *Bond Graph*.



Figura 7.7. Página de desarrollo *Teoría de Bond Graph, Conceptos y Definiciones*.



Si se presiona el botón *Aplicación Bond Graph* (Figura 7.8), se despliega la página de desarrollo *Aplicaciones de la Técnica de Bond Graph* (Figura 7.9) que contiene la aplicación de la técnica de Bond Graph a ejercicios de diferentes dominios físicos y muestra los procedimientos para aprender a modelar con la técnica de Bond Graph.



Figura 7.8. Botón *Aplicación Bond Graph*.



Figura 7.9. Página de desarrollo *Aplicaciones de la Técnica de Bond Graph*.

Si se presiona el botón *20-sim* (Figura 7.10), se despliega la página de desarrollo *Manual del Software 20-sim* (Figura 7.11) que contiene un manual del software de modelación y simulación 20-sim y muestra ejercicios propuestos a través de procedimientos para aprender a simular con la técnica de Bond Graph usando el software 20-sim.



Figura 7.10. Botón 20-sim.

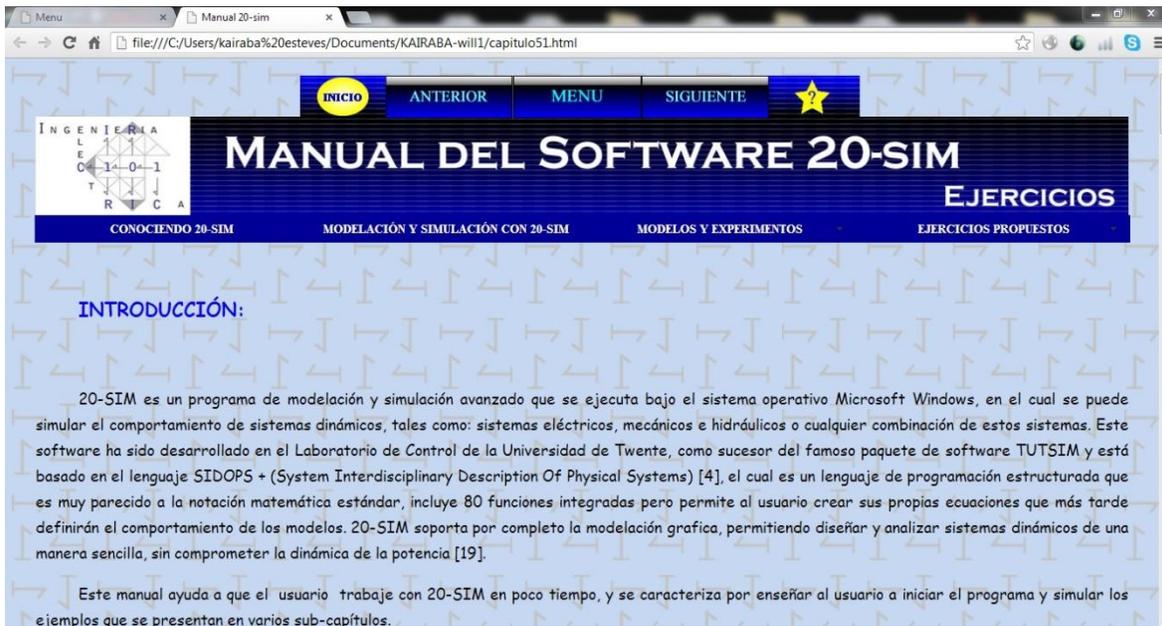


Figura 7.11. Página de desarrollo *Manual del Software 20-sim*.

Si se presiona el botón *Ayuda* (Figura 7.12), se despliega la página de desarrollo *Ayuda al Usuario* (Figura 7.13) que contiene todas las secciones de este capítulo, mostrando la distribución de la información en las diversas páginas que contiene y los elementos que sirven para la navegación y/o exploración dentro del sitio.



Figura 7.12. Botón ayuda.



Figura 7.13. Página de desarrollo. *Ayuda al Usuario*.

Si se presiona el botón *Tesis Pdf* (Figura 7.12), se despliega la página de desarrollo *Tesis.pdf* (Figura 7.13) que contiene los enlaces a todos los capítulos de la tesis en formato con extensión .pdf para permitir el acceso del usuario a toda la información del trabajo de grado.



Figura 7.14. Botón *Tesis Pdf*.



Figura 7.15. Página de desarrollo *Tesis.pdf*.



CONCLUSIONES Y RECOMENDACIONES

8.1 CONCLUSIONES.

- ↳ El diseño de un software didáctico para la enseñanza aprendizaje de técnicas de Bond Graph con aplicaciones en modelación y simulación de sistemas lineales en ambiente de programación 20-Sim, es una herramienta útil y práctica que abarca los objetivos especificados en este trabajo de grado.
- ↳ Este software didáctico engloba y sintetiza los diferentes enfoques de la técnica de Bond Graph desarrollada por los distintos autores (Paynter, Karnopp, Broenink, entre otros) señalados durante el estudio y avance del presente trabajo de grado.
- ↳ El contenido del trabajo incluye la modelación de sistemas lineales de diferentes dominios físicos expresados mediante el uso de procedimientos sistemáticos (serie de pasos) para la obtención de modelos en Bond Graph, que hacen sencillo el aprendizaje de la técnica aplicada a sistemas lineales.
- ↳ La información incluida en el Manual de 20-sim permite familiarizarse rápidamente con el software de modelación y simulación 20-sim, a través de ejemplos prácticos que muestran las características básicas y la capacidad del software para modelar y simular sistemas físicos provenientes de diferentes dominios.
- ↳ El software 20-sim es una excelente herramienta para la modelación y simulación de la técnica de Bond Graph aplicada a sistemas lineales como se comprobó en la modelación y simulación de los ejercicios de sistemas físicos lineales contenidos en el Manual de 20-sim desarrollado en este trabajo de grado.



- ↳ El software 20-sim cuenta con una función automática la cual asigna la causalidad a los diferentes elementos que conforman el modelo, lo cual ahorra tiempo y esfuerzo al modelador ya que la asignación manual de la causalidad puede tornarse en un proceso engorroso a medida que los modelos adquieren complejidad.

- ↳ La organización del software didáctico, fue diseñada de tal forma, que provee un ambiente de estudio interactivo, comprensible y de fácil navegación.

- ↳ El aprendizaje de la técnica de Bond Graph constituye un aporte importante para la actualización de los conocimientos que se imparten en la Facultad de Ingeniería ya que esta técnica incluye diferentes dominios físicos los cuales abarcan áreas como Electrónica, Mecatrónica, Robótica, Mecánica, entre otras, que son temas de relevancia en ingeniería.

- ↳ Los grafos de enlaces (bond graph) utilizan variables generalizadas, para describir el comportamiento de la dinámica de la energía y la potencia en un sistema físico; de tal forma que un mismo modelo de Bond Graph puede simular el comportamiento de varios sistemas físicos provenientes de diferentes dominios.



8.2 RECOMENDACIONES.

- ↳ Se recomienda tener conocimientos previos de sistemas lineales y sistemas de diferentes dominios físicos para facilitar la comprensión de la modelación con las técnicas de Bond Graph.
- ↳ Promover el uso de la técnica de Bond Graph y el software 20-sim para el modelado y simulación de sistemas de diferentes dominios físicos en la Facultad de Ingeniería de la Universidad de Carabobo.
- ↳ Revisar la ayuda del software didáctico antes de navegar en el sitio y/o en caso de alguna duda presentada al momento de la utilización del mismo.
- ↳ Incentivar la publicación de este software didáctico en Internet como un enlace anidado a un sitio web que pertenezca a la Facultad de Ingeniería con el objetivo de brindar este recurso informático a los interesados en el aprendizaje de las técnicas de Bond Graph.
- ↳ Se recomienda la descarga de la versión de prueba (demo) del software 20-sim con el fin de probar las habilidades, características y alcance del software 20-sim con respecto a la modelación y simulación de sistemas físicos utilizando la técnica de Bond Graph.
- ↳ Se recomienda el desarrollo de trabajos con enfoques más específicos dado a la gran variedad de aplicaciones que posee la técnica de Bond Graph orientada a sistemas físicos.



- ↳ Se recomienda hacer un estudio de la toolboxes mecánica 3D del 20-Sim con el fin de conocer la aplicación de esta para los Bond Graph.

- ↳ Se recomienda navegar con Google Chrome para tener una mejor visualización del software didáctico.

- ↳ Se recomienda instalar el lector de archivos con extensión .pdf Adobe Reader para poder tener acceso a los capítulos del Trabajo de Grado.



REFERENCIAS BIBLIOGRÁFICAS

- [1] GAWTHROP, P.; GERAINT, B. **“Bond-Graph Modeling”**. IEEE. Control Systems Magazine, pp 24-45. Abril 2007
- [2] AMELENDU MUKHERJE; RNJIT KARMAKAR; ARUN K. SAMANTARAY. **“Bond Graph in modeling, Simulation and fault Identification”**. CRC Press, 2.006. Manual.
- [3] KLEIJN. C, GROOTHUIS. M. A, DIFFER. H.G, CONTROLAB PRODUCTS B.V. **“20-sim 4.2 Reference Manual”**. Extraído desde:
<http://www.20sim.com/downloads/files/20sim42ReferenceManual.pdf>. 2011.
- [4] ROMERO REY. GREGORIO, **“Procedimientos optimizados utilizando métodos simbólicos para la simulación de sistemas dinámicos mediante Bond Graph”**. Tesis doctoral. Universidad Politécnica de Madrid. 2005
- [5] BOSNJAK S. ANTONIO M. **“Obtención y Simplificación de Ecuaciones de Estado Definitivas a partir de un Bond Graph”**. Informe de pasantías. Universidad Simón Bolívar. 1989.
- [6] BROENINK, JAN. **“Introduction to Physical Systems Modelling with Bond Graphs”**. University of Twente, Dept EE, Control Laboratory. 1999.
- [7] PAYNTER, H.M. **“The Gestation and Birth of Bond Graphs”**. Extraído desde:
<http://www.me.utexas.edu/lotario/paynter/hmp/Bondgraphs.html>.
- [8] DIAZ, JOSÉ; PERAZA, CESAR; ARTEAGA, FRANCISCO. **“Modelación y simulación de sistemas lineales utilizando Grafos de Enlace (Bond Graphs)”**. Centro de Investigación y Tecnología en Automatización Electrónica y Control (CITAEC). Universidad de Carabobo. 2005.



- [9] CELLIER, FRANÇOIS. “**Modelado Orientado a Objetos de Sistemas Físicos**” Dept. of Electrical and Computer Engineering University of Arizona. Estados Unidos. 2001.
- [10] MARQUEZ JOHANA, VARGAS DEYANIRA. “**Modelación y simulación de convertidores dc-dc y redes de atraso de fase usando técnicas de Bond Graph y el software 20-SIM**”. Trabajo de Grado, Escuela de Ingeniería Eléctrica. Facultad de Ingeniería. Universidad de Carabobo. 2010.
- [11] LABORATORI DE SISTEMES OLEOHIDRÀULICS I PNEUMÀTICS. “**Técnicas de modelación y simulación**” Universitat Politècnica de Catalunya. Ediciones UPC. 2002.
- [12] SAMANTARAY, ARUN K., BELKACEM OULD BOUAMAMA, “**Model-based Process Supervision**”. 1era Edición. Springer. 2008.
- [13] OGATA, KATSUHIKO. “**Ingeniería de Control Moderno**”, 3ª edición, Prentice Hall, 1.998.
- [14] KOFMAN, E., JUNCO, S. “**Un ambiente computacional para la modelización de sistemas dinámicos no lineales con Bond Graphs**”. VIII Reunión de trabajo en Procesamiento de la Información y Control -RPIC'99, Mar del Plata, Argentina. 1999
Extraído desde: http://www.fceia.unr.edu.ar/~kofman/files/paper_rpc99.pdf.
- [15] SAMANTARAY, ARUN K. “**About Bond Graphs**”. Artículo publicado y extraído en la web: www.bondgraphs.com/about.html. 2006.
- [16] MUÑOZ DELGADO, JOSE. “**Método de Análisis-Síntesis**”. Artículo publicado en el blog Matetam, extraído desde: <http://www.matetam.com/blog/entradas-jmd/metodo-analisis-sintesis>. 2009.



- [17] GONZALEZ AVALOS, GILBERTO. **“Técnicas de Control basadas en el Dominio Físico para Sistemas LTI MIMO. Aplicación a la Maquina Sincrónica”**. Tesis Doctoral, Universidad Autónoma de Nuevo León Facultad de Ingeniería Mecánica y Eléctrica. Mexico 2004.
- [18] PADILLA GARCIA, JOSE AARON. **“Dinámica y Estado estacionario de un Robot Manipulador en el Dominio Físico”** Tesis para Obtener Grado de Maestro en Ciencias en Ingeniería Eléctrica, Universidad Michoacana de San Nicolás de Hidalgo. 2009.
- [19] CONTROLLAB PRODUCTS B.V. **“Getting Started With 20 Sim”** manual version 3.6.03. noviembre 2004.
- [20] RIOS Z. VICTOR M, RODRIGUEZ S. HECTOR A, **“Diseño de Software Multimedia Didáctico para la Enseñanza - Aprendizaje de la asignatura Simulación”**. Trabajo de grado. Escuela de Ingeniería Eléctrica. Facultad de Ingeniería. Universidad de Carabobo. 2003.
- [21] CONTROLLAB PRODUCTS B.V. **“20-Sim The power in Modelling”**. En línea <http://www.20sim.com>.
- [22] GHELFI. GIOVANNI, **“Introducción a los Sistemas Lineales”**. Libro de texto. Escuela de Ingeniería Eléctrica. Facultad de ingeniería universidad de Carabobo. 2008.
- [23] CONTROLLAB PRODUCTS B.V. **“Using Port Names”** Archivo de ayuda del software 20-Sim. Versión 3.6.63. Noviembre 2004.
- [24] KARNOPP, DEAN C.; MARGOLIS, DONALD L.; ROSENBERG, RONALD C. **“System Dynamics, Modeling and Simulation of Mechatronic Systems”**. 4ta. edición. Wiley 2006.



- [25] NAKAMURA, SHOICHIRO. “**Análisis Numérico y Visualización Gráfica con Matlab**”. 1era. edición. Prentice Hall Hispanoamericana. 1997.
- [26] MATHEWS, JOHN H.; FINK, KURTIS D. “**Métodos Numéricos con Matlab**”. 3ra. edición. Prentice Hall. 2000.
- [27] DAMIC, VJEKOSLAV; MONTGOMERY, JOHN. “**Mechatronics by Bond Graphs**”. 1era. edición. Springer. 2002.
- [28] MESSNER, WILLIAM; TILLBURY, DAWN. “**Control Tutorials For Matlab and Simulink**”. Pearson Education, Inc. 2011. En línea:
<http://ctms.engin.umich.edu/CTMS/index.php?example=MotorSpeed§ion=SystemModeling>.
- [29] HERNÁNDEZ, MARÍA TERESA. “**Manual de Trabajos de Grado de Especialización y Maestría y Tesis Doctorales.**” Editorial: FEDUPEL. Universidad Pedagógica Experimental Libertador. 2006.



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA
DEPARTAMENTO DE SISTEMAS Y AUTOMÁTICA



**DISEÑO DE UN SOFTWARE DIDÁCTICO PARA LA ENSEÑANZA-
APRENDIZAJE DE TÉCNICAS DE BOND GRAPH CON APLICACIONES EN
MODELACIÓN Y SIMULACIÓN DE SISTEMAS LINEALES EN AMBIENTE
DE PROGRAMACIÓN 20-SIM**

KAIRABA ALEJANDRA ESTEVES SOLIS

WILLIANS DANIEL ZAPIAIN CASTILLO

Tutor: Ph.D. Francisco Javier Arteaga Bravo

RESUMEN

Este trabajo de grado tiene como objetivo fundamental proporcionar un recurso importante para la enseñanza-aprendizaje de la técnica de Bond Graph aplicada a sistemas lineales, orientada a la modelación de diferentes sistemas físicos y su simulación con el software 20-sim, plasmando la síntesis de la investigación en un software didáctico, interactivo y de fácil navegación.

El desarrollo de la investigación puede dividirse en cuatro partes principales. La primera, presenta la síntesis teórica del origen, fundamentos, variables y elementos de la técnica de Bond Graph; la segunda, muestra la aplicación de la técnica para la modelación de sistemas físicos provenientes de diferentes dominios, mediante el uso de procedimientos sistemáticos. Las características, herramientas y organización, del software 20-sim se presentan en un manual (tercera parte), que contiene adicionalmente, ejercicios de sistemas físicos lineales que permiten analizar y valorar la aplicabilidad y los alcances de la técnica de Bond Graph y el software 20-sim.

Por último, la cuarta parte agrupa las tres anteriores para integrar un software didáctico, elaborado en el ambiente de programación web Dreamweaver, que permita presentar de forma práctica el presente trabajo de grado con diseños agradables que incentivan a la enseñanza-aprendizaje de la técnica de Bond Graph.

Palabras Clave: Bond Graph, Modelación, Simulación, 20-SIM