



**UNIVERSIDAD CATÓLICA DE
SANTIAGO DE GUAYAQUIL
FACULTAD DE EDUCACIÓN TÉCNICA PARA EL
DESARROLLO**

**TESIS DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO(A) EN TELECOMUNICACIONES CON MENCIÓN EN
GESTIÓN EMPRESARIAL**

TEMA:

**APLICACIÓN DEL PROGRAMA SIMULINK DE MATLAB EN LA
ASIGNATURA DE ELECTRÓNICA I DE LA CARRERA DE INGENIERÍA
EN TELECOMUNICACIONES**

ALUMNOS:

**MARIO ANDRÉS CARRERA NOHLE
JENNIFER PAOLA VÉLEZ RUIZ**

DIRECTORA:

ING. LUZMILA RUILOVA

2011



TESIS DE GRADO

“APLICACIÓN DEL PROGRAMA SIMULINK DE MATLAB EN LA
ASIGNATURA DE ELECTRÓNICA I DE LA CARRERA DE INGENIERÍA EN
TELECOMUNICACIONES”

**Presentada a la Facultad de Educación Técnica para el Desarrollo, Carrera de Ingeniería en
Telecomunicaciones de la Universidad Católica de Santiago de Guayaquil**

REALIZADO POR:

MARIO ANDRÉS CARRERA NOHLE

JENNIFER PAOLA VÉLEZ RUIZ

Para dar cumplimiento con uno de los requisitos para optar por el título de:
Ingeniero en Telecomunicaciones con Mención en Gestión Empresarial

Ing. Luzmila Ruilova

Directora de Tesis

Ing.....

Vocal

Ing.....

Vocal

Ing. Héctor Cedeño Abad

Decano de la Facultad

Ing. Manuel Romero Paz

Director de Carrera

CERTIFICACIÓN

Certifico que el proyecto de grado titulado “Aplicación del programa Simulink de Matlab en la asignatura de Electrónica I de la Carrera de Ingeniería en Telecomunicaciones” desarrollado por Mario Andrés Carrera Nohle y Jennifer Paola Vélez Ruiz fue realizado, corregido y terminado, razón por la cual está apto para su presentación y sustentación.

Ing. Luzmila Ruilova
DIRECTORA DE TESIS

AGRADECIMIENTO

A Dios por permitirme culminar una etapa más de mi camino bajo su protección, a mis padres por guiarme a cada paso y enseñarme a escoger el camino más justo, por haber hecho de mí la persona que soy; a mis entrañables amigos por acompañarme en esta aventura llamada vida y a mi mejor amigo y compañero de tesis por ir de la mano conmigo en este largo camino y siempre ayudarme a salir adelante.

JV

Agradezco a Dios por sobre todas las cosas ya que me ha protegido cada día y por haber puesto en el camino a mi padre y a mi madre quienes han sido mi guía y ejemplo en cada aspecto de mi vida.

MC

DEDICATORIA

Esta tesis se la dedico con todo el amor y cariño a Dios que me permitió ver los días y las noches bajo su guardia y su infinito amor.

A mi madre y a mi padre quienes dedicaron su sudor, sacrificio y confianza todo este tiempo tratándome de dar lo que quizás para ellos a mi edad era muy difícil de conseguir, demostrando con ejemplos que el amor de los padres nunca acaba y no hay como medirlo.

A mis hermanas quienes siempre se preocuparon por saber sobre mi vida y brindarme sus experiencias para de ellas aprender y rescatar lo mejor de de ellas, convirtiéndolas en grandes amigas.

A mis amigos, a todos ellos, quienes me dieron la oportunidad de conocerlos, personas increíbles que me valoran por mi manera de pensar y ser, haciendo que los caminos que hemos recorrido y que espero sigamos recorriendo sean más llevaderos.

A mis maestros, personas de alta calidad humana quienes dedicaron cientos de horas a mi aprendizaje dándome las bases para poder pelear en un mundo donde el camino hacia el éxito está más cerca del que use su conocimiento para llegar a él.

MC

Esta tesis se la dedico a mis padres por su incondicional apoyo, al Ing. Manuel Romero por su inmensa dedicación, colaboración y ayuda desinteresada y a mis familiares y amigos por acompañarme en esta larga jornada.

JV

RESUMEN

La presente investigación tiene como propósito principal el realizar un análisis al pensum aplicado en la materia de Electrónica I de la Carrera Ingeniería en Telecomunicaciones de la Facultad Técnica para el Desarrollo de la Universidad Católica de Santiago de Guayaquil (UCSG), para luego poder ser aplicados en un programa (software) de simulación de dispositivos y circuitos electrónicos de tal forma que se conviertan en una herramienta tanto de enseñanza como aprendizaje para los docentes y alumnos de la Facultad.

Este procedimiento llevara a la unificación del área teórica y práctica de la materia, es decir poder aplicar los conocimientos adquiridos en el aula de manera rápida y fácil, lo cual permitirá al estudiante llegar a un conocimiento integral de la misma.

ÍNDICE

CAPÍTULO 1: GENERALIDADES DEL PROYECTO DE INVESTIGACIÓN

1.1 INTRODUCCIÓN	13
1.2 ANTECEDENTES	16
1.3 JUSTIFICACIÓN	19
1.4 PLANTEAMIENTO DEL PROBLEMA	20
1.5 HIPÓTESIS	21
1.6 OBJETIVOS	
1.6.1 OBJETIVO GENERAL	21
1.6.2 OBJETIVOS ESPECÍFICOS	21

CAPÍTULO 2: DISPOSITIVOS ELECTRÓNICOS

2.1 ELECTRÓNICA	22
2.2 DISPOSITIVOS ELECTRÓNICOS	22
2.2.1 DIODOS	22
2.2.1.1 Polarización directa de un diodo	25
2.2.1.2 Polarización inversa de un diodo	25
2.2.2 CONFIGURACIONES Y APLICACIONES DE LOS DIODOS	27
2.2.2.1 Rectificador de media onda	27
2.2.2.2 Rectificador de onda completa	29
2.2.2.3 Rectificador con derivación central	30
2.2.2.4 Puente de Diodos	31
2.2.2.5 Recortadores	31
2.2.2.6 Circuitos cambiadores de nivel	33
2.2.3 TRANSISTOR	33
2.2.3.1 Transistor de unión bipolar	34
2.2.3.2 Transistor de Unión Unipolar o de Efecto de Campo	34
2.2.3.3 Transistor de efecto de campo MOS	36

CAPÍTULO 3: PROGRAMA DE SIMULACIÓN SIMULINK	
DE MATLAB	39
3.1 INSTALACIÓN DE MATLAB	41
3.1.1 REQUERIMIENTOS DEL SISTEMA	41
3.1.2 ARCHIVOS MEX	41
3.1.3 PROCESO DE INSTALACIÓN Y ACTIVACIÓN	
DE LA VERSIÓN DE ESTUDIANTES	42
3.1.4 INICIO DE MATLAB	43
3.1.5 ARCHIVOS DE MATLAB	43
3.1.5.1 Scripts	43
3.1.5.2 Funciones	44
3.1.5.3 Tipos de funciones	46
3.1.5.4 Creación de interfaces graficas para usuarios	47
3.2 SIMULINK	51
3.2.1 SIMULINK, UNA HERRAMIENTA DE	
DISEÑO DE MODELOS.	52
3.2.2 INICIO EN SIMULINK	53
3.2.3 PREFERENCIAS DE SIMULINK	63
3.2.4 TIPOS DE PREFERENCIAS SIMULINK	63
3.2.4.1 Fuente	65
3.2.4.2 Sistema de modelos	65
3.2.4.3 Creación de un modelo simple	66
3.2.4.4 Recomendaciones	76
3.2.4.5 Ayuda de SIMULINK	76
3.2.4.6 Cambio de parámetros	77
3.2.4.7 Comentarios	78
3.2.4.8 Subsistemas	78
3.2.4.9 Funciones derivadas de MATLAB	79
3.2.4.10 Métodos de bloque	81
3.2.4.11 Tipos de métodos	81
3.2.4.12 Integrales y Derivadas	83

3.2.4.13 Usando Demos	84
CAPITULO 4: IMPLEMENTACIÓN DE PRÁCTICAS CON EL PROGRAMA DE SIMULACIÓN SIMULINK DE MATLAB	86
4.1 PROGRAMA ACTUAL DE LA ASIGNATURA ELECTRÓNICA I	86
4.2 IMPLEMENTACION DE LAS PRÁCTICAS	87
4.2.1 PRÁCTICA # 1: CONFIGURACIONES DE DIODOS CON ENTRADA EN DC	88
4.2.2 PRÁCTICA #2: RECTIFICADORES DE MEDIA ONDA POLARIZACIÓN DIRECTA	93
4.2.3 PRÁCTICA # 3: RECTIFICADORES DE MEDIA ONDA POLARIZACIÓN INVERSA	97
4.2.4 PRÁCTICA #4: RECTIFICADOR DE ONDA COMPLETA	101
4.2.5 PRACTICA #5 CIRCUITOS CAMBIADORES DE NIVEL	105
4.2.6 PRÁCTICA #6: MOSFET DECREMENTAL	108
4.3 PROGRAMA DE LA ASIGNATURA ELECTRONICA I CON LAS PRACTICAS RECOMENDADAS	111
CONCLUSIONES	113
RECOMENDACIONES	114
GLOSARIO	115
BIBLIOGRAFIA	116

ÍNDICE DE FIGURAS

CAPÍTULO 2: DISPOSITIVOS ELECTRÓNICOS

Figura 2.1: Diodo	23
Figura 2.2: Clasificación de diodos	26
Figura 2.3: Configuración de Diodos en Paralelo	27
Figura 2.4: Configuración de un rectificador de media onda	28
Figura 2.5: Tensión de entrada / rectificada	28
Figura 2.6: Rectificador de onda completa semiciclo positivo	29
Figura 2.7: Rectificador de onda completa semiciclo negativo	30
Figura 2.8: Rectificador con derivación central	30
Figura 2.9: Rectificador en Puente	31
Figura 2.10: Circuito recortador	32
Figura2.11: Circuito cambiador de nivel	33
Figura 2.12: Transistor bipolar	34
Figura 2.13: Configuración de un FET	35
Figura 2.14: Configuración de un MOSFET	37

CAPÍTULO 3: PROGRAMA DE SIMULACIÓN SIMULINK DE MATLAB

Figura 3.1: Respuesta de comando <i>type Rank</i>	45
Figura3.2: Ventana de Inicio rápido de GUIDE	48
Figura 3.3: Ventana GUIDE en blanco	49
Figura 3.4: Construcción de un modelo GUIDE	50
Figura 3.5: Ventana GUIDE en blanco	51
Figura 3.6: Forma de ingreso a Simulink	53
Figura3.7: Ventana del buscador de librerías de Simulink	54
Figura3.8: Elementos Librería <i>Sources</i>	55
Figura3.9: Librería <i>Sinks</i>	56
Figura3.10: Librería <i>Signal Routing</i>	57
Figura 3.11: <i>SimPowerSystems</i>	58
Figura3.12: Librería Fuentes Eléctricas	59
Figura3.13: Librería Elementos	60
Figura3.14: Librería Mediciones	61

Figura3.15: Librería Electrónica de Potencia	62
Figura3.16: Ventana de Preferencias Generales en Matlab	63
Figura3.17: Ventana de Preferencias de SIMULINK	64
Figura3.18: Ventana de Fuentes	65
Figura 3.19: Diagrama de bloques con una señal sinusoidal de entrada	67
Figura 3.20: Buscador de librerías SIMULINK en Windows	67
Figura 3.21: Selección de opción <i>Nuevo Modelo</i>	68
Figura 3.22: Nueva ventana de modelo	68
Figura 3.23: Ventana de bloques	69
Figura 3.24: Bloque de la Señal Seno en la ventana de trabajo	70
Figura 3.25: Bloques del modelo	71
Figura 3.26: Entrada/salida de bloque	71
Figura 3.27: Inicio de la conexión de bloques	71
Figura 3.28: Cambios del cursor en la conexión de bloques	72
Figura 3.29: Conexión de bloques terminada	72
Figura 3.30: Inicio de la conexión de la Línea de Rama	73
Figura 3.31: Arrastre del cursor para la conexión de la Línea de Rama	73
Figura 3.32: Conexión de Línea de Rama	74
Figura 3.33: Modelo de bloques	74
Figura 3.34: Configuración de parámetros	75
Figura 3.35: Visualización de Señales	75
Figura 3.36: Ventana <i>Ayuda SIMULINK</i>	77
Figura 3.37: Ventana de Parámetros de Bloque	77
Figura 3.38: Cambio de comentarios en bloques	78
Figura 3.39: Visualización de un subsistema	79
Figura 3.40: Programación en MATLAB	80
Figura 3.41: Diagrama en SIMULINK con una función MATLAB asignada a un bloque	80
Figura 3.42: Diagrama del bloque <i>Transferencia</i>	81
Figura 3.43: Ventana de parámetros	82
Figura 3.44: Simulación de función	82

Figura 3.45: Diagrama de bloques de funciones Integrales y Derivadas	83
Figura 3.46: Señales de salida	84
Figura 3.47: Ingreso a <i>Demos</i>	84
Figura 3.48: Panel de <i>Demos</i>	85

CAPÍTULO 4: IMPLEMENTACIÓN DE PRÁCTICAS CON EL PROGRAMA DE SIMULACIÓN SIMULINK DE MATLAB

Figura 4.1: Diodos en paralelo	89
Figura 4.2: Rectificador de media onda con entrada DC	90
Figura 4.3: Visualización de señales de voltaje y corriente	91
Figura 4.4: Visualización de señales de voltaje y corriente en el diodo	92
Figura 4.5: Visualización de señales de voltaje y corriente a la salida del circuito.	93
Figura 4.6: Configuración de un rectificador de media onda	94
Figura 4.7: Tensión de entrada / rectificada	94
Figura 4.8: Resultado de la práctica Rectificadores de media onda polarización directa	95
Figura 4.9: Visualización de señales de voltaje y corriente a la salida del circuito.	96
Figura 4.10: visualización de señales de voltaje y corriente en el diodo	97
Figura 4.11: Configuración de un rectificador de media onda	98
Figura 4.12: Tensión de entrada / rectificada	98
Figura 4.13: Resultado de la práctica Rectificadores de media onda polarización inversa	99
Figura 4.14: Visualización de señales de voltaje y corriente	100
Figura 4.15: Puente de Diodos	101
Figura 4.16: Resultado de la práctica Rectificador de onda completa	103
Figura 4.17: Visualización de señales de voltaje y corriente	104
Figura 4.18: Circuito cambiador de nivel	105
Figura 4.19: Resultado de la práctica Circuitos cambiadores de nivel	106
Figura 4.20: Visualización de señales de voltaje y corriente	107
Figura 4.21: Resultado de la práctica MOSFET Decremental	109

CAPÍTULO 1: GENERALIDADES DEL PROYECTO DE INVESTIGACIÓN

En este capítulo se desarrollan los aspectos fundamentales de la investigación, estableciéndose de esta manera los antecedentes de la misma que permiten justificar su realización, la definición del problema determinado en el análisis realizado, la hipótesis que permita su posible solución y los objetivos para conseguirlo.

1.1 INTRODUCCIÓN

El avance de la tecnología en nuestros días se debe principalmente a la creación de circuitos más pequeños y funcionales en todos los ámbitos de la vida del ser humano, desde equipos para facilitar la labor doméstica hasta equipos de comunicaciones sofisticados como terminales celulares, satélites, etc.

Este fenómeno tuvo inicio en el siglo XX cuando Thomas Edison introdujo las válvulas de vacío al mundo de la ciencia, lo cual impulsó al estudio más a fondo de los mismos creando variaciones que llevarían al hombre a contar con inventos como son los sistemas de computación, la telefonía y una amplia variedad de artefactos que facilitaron el avance de la tecnología y mejoraron la calidad de vida del hombre.

Basados en el principio de las válvulas, se produjeron un sin número de mejoras de las mismas las cuales fueron desde los amplificadores, tetrodos, pentodos, etc., hasta llegar al transistor el cual se convertiría en la base de la mayoría de los circuitos electrónicos debido a la mejora que brindan sus propiedades a los mismos y dando paso de esta manera a los circuitos integrados los cuales parten de la utilización de transistores para lograr el proceso definido según su clasificación.

La unión efectiva de estos elementos han llevado al ser humano a lograr un gran avance en la tecnología, por lo que esta investigación se basa en el análisis de las diferentes disposiciones que pueden llegar a darse con estos elementos para luego poder ser llevados a un software en el cual podremos simular su uso y diferentes tipos de conexiones.

El programa que se pretende usar es MATLAB (*MATrix LABORatory*, Laboratorio de Matrices); es un *software* matemático que ofrece un entorno de desarrollo integrado (IDE) con un lenguaje de programación propio (lenguaje M) que permite la simulación de circuitos tanto electrónicos como digitales convirtiéndose en una herramienta complementaria para establecer las posibles diferencias que pueden existir entre el comportamiento teórico y el comportamiento real de los circuitos.

MATLAB tiene como creador a Cleve Moler, el cual en 1984 lanzó la primera versión de este programa con el fin de fusionar subrutinas desarrolladas en Fortran para resolución de problemas de álgebra y análisis numérico, sin tener que escribir la programación en el lenguaje antes mencionado.

Este programa se basa en el lenguaje M, creado en 1970, siendo creado para permitir un acceso directo a los *software* Linkpack y Eispack, lo cual convierte a Matlab en una herramienta de última generación muy útil en lo que computación matricial se refiere.

Por esta razón MATLAB se proyecta como una potencial solución y más que nada como una herramienta de cálculo y modelado a la hora de resolver problemas complejos en los que es necesario aprovechar las amplias capacidades de proceso de datos de grandes computadores.

Es por esta razón que MATLAB cubre una amplia gama de áreas en las que se puede aplicar sus bondades, las cuales incluyen el diseño de todo tipo de sistemas en especial de control, así como el área de investigación médica y procesamiento de señales.

Las bondades de este programa van direccionadas específicamente a las necesidades de los usuarios debido a que permite personalizar herramientas y explorar datos lo cual le da la opción de tener un conocimiento más profundo del área en que se está desarrollando así como le permite crear elementos competitivos.

MATLAB ha llegado a desarrollarse de tal manera que en estos días cuenta con un sin número de aplicaciones que han resultado de gran ayuda para universidades y centros de investigación y

desarrollo. Estas aplicaciones incluyen programadores para procesadores digitales los que permiten la transferencia directa de líneas de comando al dispositivo. Otra aplicación de importante trascendencia es la que permite la creación de código VDHL (*VHSIC Hardware Description Language*).

Entre sus características fundamentales se puede resaltar las siguientes:

- El manejo de matrices,
- La representación de datos y funciones,
- La implementación de algoritmos,
- La creación de interfaces de usuario (GUI) y
- La comunicación con programas en otros lenguajes y con otros dispositivos *hardware*.

Muy aparte de las aplicaciones que el usuario puede crear, MATLAB incluye dos herramientas que lo hacen un programa muy atractivo y funcional.

- Simulink (plataforma de simulación multidominio)
- GUIDE (editor de interfaces de usuario - GUI).

MATLAB también incluye librería de herramientas (*toolboxes*) y paquetes de bloques (*blocksets*) en lo que Simulink se refiere, las cuales son conjuntos de opciones que posee el usuario para crear modelos, proyectos y simulaciones.

Al analizar la exigencia del mundo laboral que requiere profesionales que se enfrenten a retos tecnológicos constantemente es posible darse cuenta que las bondades de este *software* pueden ayudar fácilmente con la necesidad de los estudiantes de la carrera de Ingeniería en Telecomunicaciones de afianzar los conocimientos teóricos adquiridos en el aula por medio de la práctica de los mismos; por esta razón esta investigación se basará en brindar una herramienta de trabajo tanto para el Catedrático como para el estudiante de Electrónica I, llevándolos a cumplir con la meta específica de todo aquel dedicado al estudio, la cual es la excelencia académica.

1.2 ANTECEDENTES

La asignatura de Electrónica I es dictada en el Cuarto Ciclo de la Carrera de Ingeniería en Telecomunicaciones y por la gran importancia de los conocimientos que sobre este campo incluye esta materia, es considerada como común (aula conjunta) para las Carreras de Ingeniería Eléctrico-mecánica y de Ingeniería Electrónica en Control y Automatismo, todas ellas dictadas en la Facultad de Educación Técnica para el Desarrollo de la UCSG.

En esta asignatura se analizan los fundamentos de la Electrónica en función de la gran importancia que este campo del conocimiento tiene debido a que se ha convertido en algo prácticamente indispensable para la vida diaria, debido a los avances tecnológicos que se han producido, asimismo también cabe decir que es difícil encontrar a alguien que no haya oído mencionar la Electrónica, aunque en realidad pocos saben en qué consiste. La mayor parte de las personas ven como una necesidad básica los servicios de la electrónica y telecomunicaciones y considerando que los mismos no pueden funcionar sin los dispositivos electrónicos (celular, computadora, e internet), demuestra claramente lo importante que es la tecnología electrónica. Así por ejemplo, puede asegurarse que si en una empresa falta el internet o la informática en general, se produce un déficit tecnológico, puesto que se obstaculiza la empresa completa, debido a que la misma trabaja con sistemas informáticos, que permiten la realización de transacciones bancarias y comerciales, envíos de correo electrónico, etc.

La Electrónica ha causado la evolución a una nueva era: la era digital, esto ha provocado cambios en la manera de pensar de las personas, el estilo de interactuar y en definitiva sus costumbres a causa del desarrollo que se producido en la electrónica.

En el campo de las Telecomunicaciones, basta únicamente mencionar un par de ejemplos para demostrar la importancia que este campo posee en la vida moderna: los celulares y la televisión digital; además del crecimiento del internet y la forma en que éste a revolucionado al mundo moderno.

En la asignatura de Electrónica I se desarrollan temas acerca de los dispositivos básicos de todo circuito electrónico y sus materiales constitutivos. Así, el curso se inicia con una descripción de los materiales semiconductores como paso previo a la introducción de los dispositivos electrónicos, empezando desde el más sencillo, pero no por eso de singular importancia como es el diodo; en este punto del estudio se demuestra la importancia que reviste en la mayoría de las aplicaciones la utilización de elementos ideales y circuitos equivalentes que por lo general son mucho más fáciles de analizar y evaluar que los dispositivos reales, pero que sin embargo permiten obtener resultados que constituyen una excelente aproximación de los resultados efectivos; es importante notar en este punto que este concepto es la base para la utilización de programas simuladores para obtener resultados muy cercanos a los efectivos sin necesidad de utilizar dispositivos y circuitos reales.

Luego el curso continúa con el análisis de dispositivos un poco más complejos como los transistores bipolares de unión (BJT), los transistores de efecto de campo (FET) y los MOSFET (*Metal Oxide Semiconductor Field Effect Transistor*, FET de Oxido Metálico Semiconductor), todos ellos son estudiados en el campo dc, así como su respuesta en el campo ac.

Sin embargo, todo este análisis se lo realiza únicamente desde el punto de vista teórico y no incluye prácticas que permitan a los estudiantes corroborar los resultados obtenidos en la resolución teórica de los circuitos presentados en clase. Es importante recordar que en Cuarto Ciclo, en el que se dicta esta asignatura, los estudiantes no reciben clases de Laboratorio de Electrónica pues este corresponde al Quinto Ciclo junto a Electrónica II y por lo tanto el programa de dicho laboratorio debe abarcar los conocimientos que los estudiantes han recibido en las dos asignaturas teóricas.

Con estos antecedentes se establece la necesidad de incluir dentro del programa de Electrónica I alguna forma de brindar un complemento práctico para las clases teóricas de la asignatura. Por esta razón surge la idea de brindar a los estudiantes tales prácticas mediante un programa para simular dispositivos y circuitos electrónicos que sea al mismo tiempo fácil de utilizar y que permita obtener excelentes resultados.

Luego de analizar algunos programas simuladores, se consideró que el software más adecuado para implementar prácticas dentro de la asignatura de Electrónica I era MATLAB puesto que es una de las aplicaciones más útiles que existen para poner a punto métodos numéricos en distintas asignaturas de ingeniería, debido a que es una herramienta de alto nivel el desarrollo de programas numéricos con MATLAB puede requerir hasta un orden de magnitud menos esfuerzo que con lenguajes de programación convencionales, como Fortran, Pascal, C/C++, Java o Visual Basic.

MATLAB es un lenguaje de alta ejecución para desarrollo técnico, este integra tanto la visualización como la programación en un ambiente muy fácil de usar donde los problemas y soluciones son expresados en una notación matemática familiar (TP de Introducción a MATLAB/SIMULINK).

Los usos típicos de este programa incluyen:

- Matemáticas y computación
- Desarrollo de algoritmos
- Simulación y desarrollo de prototipos
- Análisis de datos, exploración y visualización
- Gráficos científicos y de ingeniería
- Desarrollo de aplicaciones, incluyendo el desarrollo de interfaces gráficas de usuario

Una característica importante de MATLAB es que su elemento de datos básico es un arreglo que no necesita dimensionamiento, esto provee una gran ventaja puesto que resuelve una gran cantidad de problemas técnicos especialmente a la hora de trabajar con formulaciones de matrices y vectores (Giner, 2008).

Esta característica se desprende del hecho que MATLAB fue creado originalmente para proveer un acceso fácil a programas basados en matrices los cuales fueron desarrollados por LINKPACK.

1.3 JUSTIFICACIÓN

Los avances en cualquier campo necesitan mucha investigación dado el número de pruebas que se tienen que realizar para ajustar a la realidad la solución para un problema ya sea éste acerca de la forma, funcionamiento, velocidad o comportamiento aplicado a un sujeto animado o inanimado; es aquí donde entra la gran importancia de conocer y entender la parte teórica de una materia y a partir de eso buscar la solución del problema planteado ya que fundamentalmente esa es la razón por la que es necesario prepararse para ser ingenieros.

Como ya se indicó la electrónica es una de las bases de cualquier avance tecnológico tomando como punto de referencia el hecho de que el desarrollo en este campo enfatiza en tratar de hacer los dispositivos más pequeños, rápidos y con mayores capacidades de procesamiento o almacenamiento, esto es lo que se puede ver en el día a día ya sea cuando se manipula un teléfono celular, el modem de algún proveedor de internet, un dispositivo de almacenamiento USB, y el resto de dispositivos que ayudan a realizar las actividades cotidianas.

Si por una parte es sumamente importante asimilar la teoría que imparten los catedráticos en una carrera lo es también poder poner en práctica lo que los estudiantes tratan de entender en las aulas, que quizás en algunos casos se puede decir que ver la situación aplicada ayuda a entender las palabras o el texto concerniente a una materia; por estas razones se justifica el uso de herramientas que faciliten las pruebas de lo aprendido en clases y de alguna manera simular lo que en algún momento se usará en el campo laboral, por lo que en una materia técnica como lo es Electrónica I, el realizar talleres de prácticas sobre lo aprendido en clases ayuda a alinear lo que se ve en realidad con lo que se conoce en la parte teórica.

Estando consientes de la importancia de la parte práctica de una materia y más aun si esta es una materia técnica como lo es Electrónica I y en lo que se basa el avance en este campo, considerando lo difícil que sería implementar siempre algo aprendido en clases ya que se tiene la posibilidad de cometer errores, se considera que la mejor manera de hacerlo sería simularlo usando un *software* apropiado especialmente para este fin, de esta manera se pueden realizar las

pruebas y cambios que sean necesarios para obtener los resultados esperados teniendo así el máximo rendimiento en la práctica.

Este trabajo de investigación plantea el trabajo conjunto tanto de la parte teórica impartida en clases con la práctica dada en talleres explicativos, buscando no solo alcanzar a entender que es lo que sucede en cuanto a la materia si no también en algún punto mejorar lo aprendido invirtiendo más tiempo tratando de innovar sobre el tema del que se está tratando ya que es así como los avances tecnológicos salen adelante.

El uso del MATLAB, al tener herramientas que permiten simular el comportamiento de un circuito electrónico ya sea básico o complejo permitiría mostrar lo que se imparte en la materia Electrónica I y así complementar el actual pensum pasando de ser una materia mayormente teórica a una también aplicativa que busca que los estudiantes visualicen que es lo que sucede en los componentes electrónicos o en los circuitos y así lograr que se involucren más en la materia no solo quedándose con lo aprendido en clases sino tratando de avanzar un poco más allá de texto ya que el programa permite hacer variantes que pueden afectar positivamente un circuito y de esta manera se va construyendo de a poco lo que toda carrera en todas las universidades en el mundo buscan, profesionales que vayan más allá de las aulas con la ayuda de las poderosas herramientas que están al alcance de todos actualmente aportando al desarrollo de nuevas soluciones generales que la electrónica plantea día a día.

1.4 PLANTEAMIENTO DEL PROBLEMA

La necesidad de reforzar la materia Electrónica I del Cuarto Ciclo de la Carrera de Ingeniería en Telecomunicaciones con demostraciones prácticas que ayuden al estudiante a comprender de una mejor manera los conocimientos teóricos recibidos, mejorando de esta manera el proceso enseñanza-aprendizaje.

1.5 HIPÓTESIS

El uso de un software que permita la posibilidad de simular e implementar los diferentes circuitos electrónicos impartidos en clase, dará al estudiante la opción de desarrollar el aprendizaje experimental y la aplicación de los conocimientos teóricos adquiridos previamente.

1.6 OBJETIVOS

Los objetivos planteados para este proyecto de investigación son los siguientes:

1.6.1 OBJETIVO GENERAL

Aplicar el software MATLAB como herramienta de carácter práctico para el programa académico de la asignatura de Electrónica I de la Carrera de Ingeniería en Telecomunicaciones.

1.6.2 OBJETIVOS ESPECÍFICOS

- Analizar el programa y syllabus de la asignatura Electrónica I y determinar las áreas del mismo que necesitan apoyo práctico.
- Aprender el manejo del programa MATLAB.
- Determinar las herramientas de MATLAB necesarias para desarrollar los temas decididos.
- Determinar las prácticas que serán realizadas.
- Presentar un nuevo programa y syllabus para la asignatura de Electrónica I con la incorporación de las prácticas recomendadas.

En el siguiente capítulo se desarrollaran los conceptos fundamentales incluidos en el programa académico de la asignatura Electrónica I y los dispositivos tales como el diodo, los transistores BJT, JFET y MOSFET.

CAPÍTULO 2: DISPOSITIVOS ELECTRÓNICOS

A continuación, se procederá a realizar una breve descripción de los conceptos principales en el estudio de la asignatura de Electrónica I y los cuales son parte vital de esta investigación.

2.1 ELECTRÓNICA

La electrónica es una de las tantas ramas de la ingeniería, la cual enfoca su estudio a la creación de diferentes dispositivos así como su funcionamiento, siendo los más comunes los circuitos electrónicos. Estos circuitos poseen como característica que deben poder transportar energía eléctrica para desempeñar sus funciones las cuales abarcan transmisión, almacenamiento, y generación de datos.

Estas funciones abarcan todo lo concerniente a procesamiento de información, lo cual incluye modulación y demodulación de señales, el llevar señales a niveles donde puedan operar también conocido como amplificación, así como operaciones lógicas las cuales son de gran importancia en equipos de última tecnología, como computadoras, equipos de rastreo, etc.

2.2 DISPOSITIVOS ELECTRÓNICOS

Los circuitos electrónicos están constituidos por dispositivos electrónicos conectados apropiadamente entre sí. Estos dispositivos pueden ser activos o pasivos.

Las resistencias, los condensadores y las bobinas se consideran componentes pasivos, mientras que los activos incluyen las baterías, los generadores y los transistores (CODETEL, 2009).

2.2.1 DIODOS

Los diodos son dispositivos electrónicos que poseen dos terminales permitiendo de esta manera el flujo de electrones a través del mismo en un solo sentido; e impidiendo el paso de la corriente en sentido contrario.

En nuestros días el tipo de diodo más utilizado es el diodo semiconductor, este se encuentra formado por una pieza de cristal semiconductor conectada a los dos terminales, de ahí el origen de su nombre.

Cabe mencionar que existen parámetros dentro de los cuales el diodo puede operar, produciendo de esta manera una curva característica que representa al diodo (I-V) y que presenta dos áreas bien definidas: una región de no conducción localizada antes de que el diodo alcance un determinado potencial, denominado voltaje de umbral, donde para aplicaciones prácticas se puede considerar al dispositivo como un circuito abierto por la alta resistencia que presenta entre sus terminales, y por encima de este umbral puede considerarse como un circuito cerrado con una resistencia eléctrica muy pequeña, constituyendo la segunda región denominada de conducción.

Por este motivo se utiliza los diodos, dispuestos en configuraciones características como rectificadores, debido a que pueden llegar a suprimir partes de una señal de entrada establecida, para el uso de la parte necesaria, este es el principio vital de la conversión de corriente alterna a continua.

La figura 2.1 muestra un modelo característico de un diodo, así como su símbolo correspondiente y la dirección de propagación de la corriente.

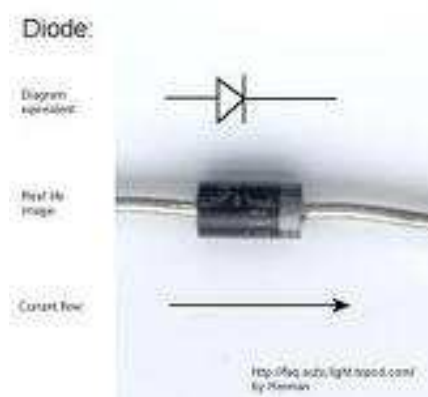


Figura 2.1: Diodo

Fuente: <http://www.dtic.upf.edu/~jlozano/interfaces/interfaces8.html>

Como ya se indicó, el diodo semiconductor tiene como elemento principal un compuesto semiconductor, de todos los que existen en la naturaleza el más utilizado para este propósito es el silicio.

Cabe recalcar que para que un elemento semiconductor sea apropiado para formar parte de un diodo debe poseer un alto nivel de impurezas (dopado) debido a que éstas son las que originan una región que contiene portadores de carga negativos (electrones), un material semiconductor con estas características es llamado de tipo n y una región que contiene portadores de carga positiva (huecos), llamada tipo p. Los terminales del diodo se unen a cada región. La importancia del funcionamiento del diodo radica en la unión PN, la cual es el límite dentro del cristal de estas dos regiones.

La corriente que va a recorrer el diodo es transportada por el cristal semiconductor desde el lado **n** representado por el cátodo del mismo hacia el ánodo. Como se comentó anteriormente, el diodo no permite el flujo de corriente en sentido contrario, es decir en dirección del ánodo al cátodo.

Al realizar la unión de estas áreas, es decir **n** y **p**, se crea una región llamada de agotamiento luego de la aparición de cargas fijas en ambas zonas, al conjunto de estas cargas se les llama corriente de difusión. Esta región de difusión comienza a llegar a su fin, cuando una nueva zona llamada de agotamiento comienza a incrementar su anchura lo cual logrará una estabilización en el sistema.

Este fenómeno se presenta debido a que en cada área se albergan iones con signos contrarios los cuales llegan a formar un campo magnético que actúa sobre los electrones que quedaron libres en el área **n**, haciendo de esta manera que el flujo de los mismos se detenga.

Luego de haber alcanzado el equilibrio, el ancho de la región de agotamiento es casi siempre del orden de 0,5 micras pero cuando uno de los cristales está mucho más dopado que el otro, la zona de carga espacial es mucho mayor (Cid, 2007).

Los diodos poseen la particularidad que al ser sometidos a un diferencial de tensión, pasan a un estado de polarización, teniendo la posibilidad ésta de ser directa o inversa. A continuación, la diferencia entre los tipos de polarización.

2.2.1.1 Polarización directa de un diodo

La particularidad de esta polarización es que la forma en que está dispuesto el diodo permite que la barrera en la zona de unión se atenúe y permita el tránsito de electrones a lo largo del área de unión.

Para que exista esta polarización la conexión del ánodo y cátodo deben ser en el polo positivo y negativo respectivamente. En resumen, se puede decir que el diodo polarizado directamente conduce electricidad, teniendo la configuración correspondiente.

2.2.1.2 Polarización inversa de un diodo

La polarización inversa en un diodo consiste en la interrupción del flujo de electrones a través el mismo, este proceso se lo realiza mediante la conexión del cátodo y ánodo a los polos positivo y negativo respectivamente, esta disposición permite que la zona de barrera en la unión de las áreas presente una mayor concentración de carga lo que impide el tránsito de corriente.

Aunque el diodo este polarizado de esta manera, debido a causas de temperatura se crean espacios entre los electrones, este fenómeno es llamado par electrón-hueco, del cual se genera una corriente de muy bajo amperaje llamada de saturación inversa, así como una corriente superficial del diodo; éstas tienen en común que son de valores muy pequeños por lo que se desprecia en procesos de cálculos o análisis de funcionamiento del dispositivo.

La zona de ruptura es un estado del diodo en el cual se presentan altos valores de tensión y corriente originando de esta manera valores elevados de potencia lo cual produce un aumento en la temperatura de la unión **pn**, este fenómeno se da al polarizar el diodo de forma inversa lo cual genera que el mismo conduzca la corriente inversa de saturación cuyo valor alcanza niveles

constantes, pero en la practica se necesita un valor de tensión determinado así como del efecto avalancha para que el diodo normal o de unión abrupta llegue a este estado, produciendo un aumento de la corriente. Dentro de la clasificación de los diodos existen otros tipos en los que este estado se debe a otros factores, estos son:

- **Efecto avalancha** (diodos poco dopados). Como se había explicado anteriormente al polarizar un diodo inversamente se generan pares electrón-hueco lo cual da paso a la corriente inversa de saturación, por otra parte si se aplica una tensión inversa muy elevada, es decir mayor a 6V, los electrones del flujo se acelerarán haciendo que los mismos choquen entre si y así provocando la liberación de los mismos, de esta manera se genera una avalancha de estos electrones lo que se traduce en una corriente grande.
- **Efecto Zener** (diodos muy dopados). Cuanto más dopado está el material, menor es la anchura de la zona de carga por lo que puede generar que el propio campo sea capaz de arrancar electrones de valencia incrementándose la corriente. Este efecto se produce para tensiones de 4 V o menores. En casos como el diodo Zener, si se aplican tensiones inversas dentro del rango entre 4 y 6 V, el fenómeno de ruptura se puede dar por ambos efectos (Diodo, 2011).

En la figura 2.2 se representan algunos de los tipos de diodos más conocidos.



Figura 2.2: Clasificación de diodos

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

2.2.2 CONFIGURACIONES Y APLICACIONES DE LOS DIODOS

Luego de explicar la composición y funcionamiento de los diodos es importante analizar las configuraciones en las cuales se pueden disponer a los diodos para diferentes aplicaciones de la tecnología, las cuales se detallan a continuación:

Estas configuraciones se dividen tanto en conexiones en serie o en paralelo de los diodos cuando existe una entrada de voltaje directo, como se puede apreciar en la figura 2.3:

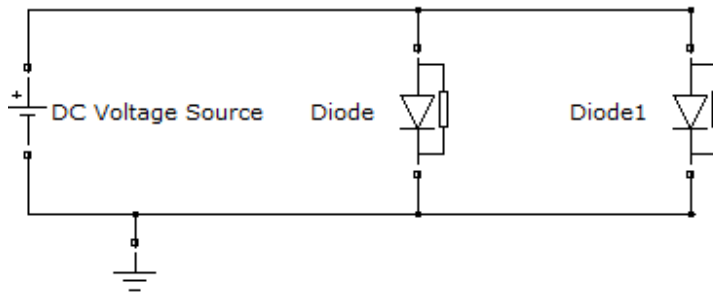


Figura 2.3: Configuración de Diodos en Paralelo

Fuente: *Print Screen* de Matlab Versión 7.8.0.347 (R2009a)

Por otro lado existen las configuraciones que se pueden implementar cuando la entrada de voltaje del circuito es alterna o sinusoidal, estas son:

2.2.2.1 Rectificador de media onda

El rectificador de media onda es un circuito electrónico el cual se usa para la eliminación del área ya sea positiva o negativa de una onda determinada siendo esta de corriente alterna, este tipo de circuitos permite que la corriente alterna de ingreso luego del paso por el mismo llegue a ser corriente directa a la salida.

Como ya se había tratado, los diodos permiten el paso de la corriente en una sola dirección por lo que dependiendo de la polarización que se aplique a estos, se comportarán como circuito abierto o permitiendo el paso de la corriente.

En el caso de esta práctica utilizada a modo de ejemplo, el diodo va a permitir el paso de la corriente, y teniendo en cuenta que el diodo necesita un voltaje característico para su polarización (germanio: 0,3V y silicio: 0,7V) el voltaje de entrada a la salida se verá reducido en este valor.

$$\circ V_o = V_i - V_D \rightarrow V_o = V_i - 0,7 \text{ (Silicio)} \quad \text{Ecuación 2.1}$$

$$\circ V_o = V_i - V_D \rightarrow V_o = V_i - 0,3 \text{ (Germanio)} \quad \text{Ecuación 2.2}$$

La característica de la configuración mostrada en la figura 2.4 es que suprime la parte negativa de la señal sinusoidal de entrada, así como la conversión de estado alterno a directo de la corriente de entrada.

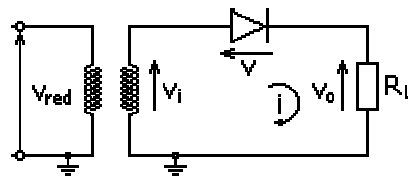


Figura 2.4: Configuración de un rectificador de media onda

Fuente: http://es.wikipedia.org/wiki/Archivo:Tensi%C3%B3n_rectificada_media_onda.png

Las formas de onda resultantes se muestran en la figura 2.5:

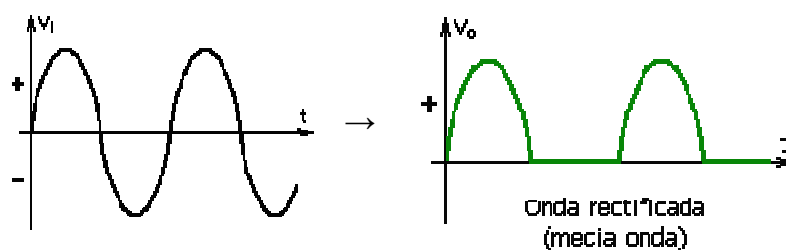


Figura 2.5: Tensión de entrada / rectificada

Fuente: http://es.wikipedia.org/wiki/Archivo:Tensi%C3%B3n_rectificada_media_onda.png

2.2.2.2 Rectificador de onda completa

Los rectificadores de onda completa son de mucha utilidad debido a que la señal que distribuyen todas las empresas eléctricas en cualquier lugar del mundo es de tipo alterna y por otro la mayoría de los equipos que se utilizan en los hogares funcionan con corriente directa, por lo que se necesita de un circuito que regule este problema.

En primera instancia se debe analizar que el voltaje entregado es de 120/220 V y los equipos funcionan en un intervalo de 12 a 15 V por lo que es necesario llevar el voltaje entrante a estos niveles por medio del uso de transformadores, los cuales a su vez generan una oscilación de la señal de semiciclo positivo y otro negativo, por lo que se necesitan los diodos para corregir este problema como se detalla a continuación:

Para el semiciclo positivo el diodo se polariza directamente, y como se había explicado anteriormente, trabajará como cortocircuito en este estado, haciendo la función de un rectificador de media onda, en la figura 2.6 se analiza este caso:

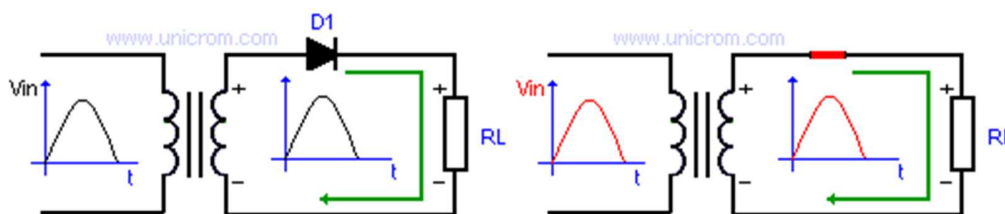


Figura 2.6: Rectificador de onda completa semiciclo positivo

Fuente: http://www.unicrom.com/Tut_rectificador_media_onda.asp

Por otro lado cuando el secundario del transformador entrega la corriente durante el semiciclo negativo, estará recorriendo el diodo en forma opuesta por lo que generará que se comporte como un circuito abierto, interfiriendo la entrega de corriente a la salida del circuito.

Este caso se aprecia en la figura 2.7 a continuación:

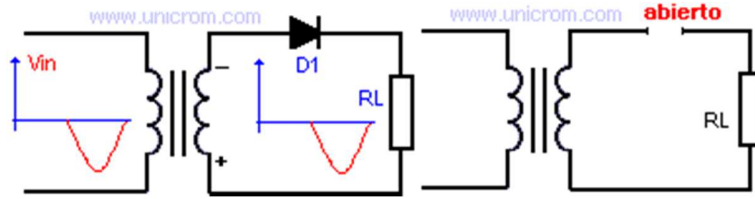


Figura 2.7: Rectificador de onda completa semiciclo negativo

Fuente: http://www.unicrom.com/Tut_rectificador_media_onda.asp

No es posible tener a un equipo electrónico con estas variaciones de ciclo debido a que pudiera llegar a dañarse o solamente no funcionaria de forma correcta. Por lo que existen configuraciones que permiten tener el mejor uso de estas propiedades de los diodos, como las que se detallan a continuación:

2.2.2.3 Rectificador con derivación central

Este tipo de rectificador permite el uso de la señal entrante ya sea durante el semiciclo positivo o el negativo, esto se debe a como están colocados los diodos y las resistencias, como se puede observar en la figura 2.8, al ingresar el semiciclo positivo el diodo D2 queda como un circuito abierto por lo que D1 se comportaría como un cortocircuito permitiendo el paso de la corriente y creando un voltaje de salida en RL.

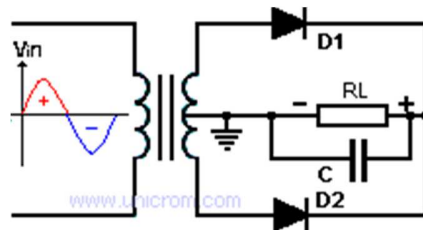


Figura 2.8: Rectificador con derivación central

Fuente: http://www.unicrom.com/Tut_rectificador_onda_completa.asp

Por otro lado, al ingresar el semiciclo negativo, D1 queda como circuito abierto y D2 conduce, de igual manera permitiendo el paso de la corriente por ese ramal del circuito hasta llegar a RL y generando un voltaje de salida. Debido a la forma en que se realiza la conexión la corriente

siempre va transitar de la misma manera por RL lo cual genera un voltaje unipolar, necesario para el funcionamiento de los equipos eléctricos y electrónicos.

2.2.2.4 Puente de Diodos

El rectificador de puente de diodos, presenta una configuración que necesita de cuatro diodos dispuestos para que trabajen de la siguiente manera:

- Cuando el circuito está siendo alimentado por el semiciclo positivo la corriente circulará por los diodos D1 y D2 permitiendo la formación de un voltaje de salida en R.
- Cuando el semiciclo negativo se presenta en el circuito, los diodos que conducirán son D3 y D4 generando así mismo un voltaje de salida en RL.

Luego de este proceso se puede obtener una señal unipolar que servirá para poner en funcionamiento los equipos eléctricos o electrónicos en el hogar. Cabe mencionar que debido a la caída de voltaje que presentarán ambos diodos, el voltaje que ingresó será disminuido en la tensión necesaria para la polarización. Esta configuración de diodos se muestra en la figura 2.9.

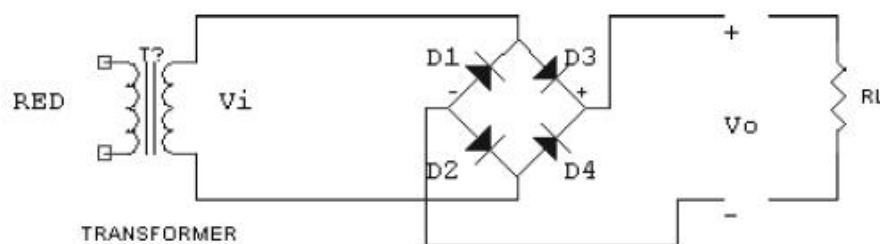


Figura 2.9: Rectificador en Puente

Fuente: [http://www.info-](http://www.info-ab.uclm.es/labeledc/Solar/elementos_del_pc/fuentes_de_alimentacion/fwCB.htm)

[ab.uclm.es/labeledc/Solar/elementos_del_pc/fuentes_de_alimentacion/fwCB.htm](http://www.info-ab.uclm.es/labeledc/Solar/elementos_del_pc/fuentes_de_alimentacion/fwCB.htm)

2.2.2.5 Recortadores

Otro tipo de configuración de los diodos son los circuitos recortadores, los mismos que se caracterizan por entregar una señal de salida la cual es una porción de la señal de entrada, esta porción se basa en los parámetros que se le aplique al circuito.

En este caso también se ve afectado por los semiciclos de la señal de entrada para lo cual, en el caso del semiciclo positivo se dan dos opciones:

- Si el voltaje de entrada es menor que el voltaje DC, el diodo se verá polarizado inversamente por lo que se comportará como circuito abierto, permitiendo que el voltaje de salida en R o sea igual al de entrada.
- Si el voltaje de entrada es mayor al voltaje DC, el diodo se comporta como un cortocircuito, por lo que el voltaje que se percibirá a la salida será igual al de DC.

Por otro lado en el semiciclo negativo, el voltaje de salida no dependerá del valor que tenga el de entrada, esto se da debido a que el diodo estará recibiendo el ciclo negativo de la señal con lo que este se comportará como un circuito abierto, por lo que el voltaje de salida será igual al de DC.

Un circuito de este tipo se muestra en la figura 2.10.

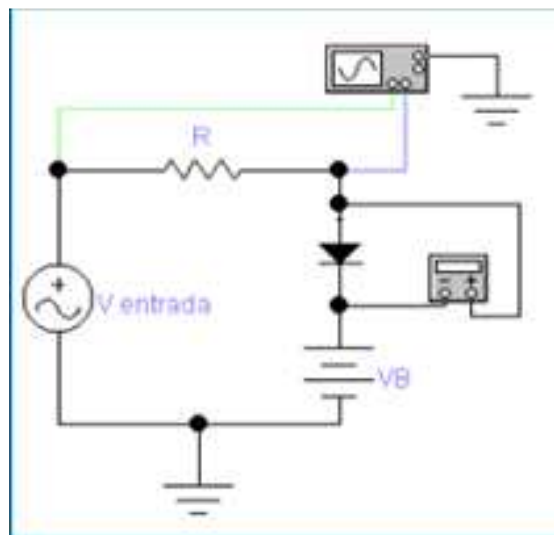


Figura 2.10: Circuito recortador

Fuente: <http://gafuhe.galeon.com/tarea4.htm>

2.2.2.6 Circuitos cambiadores de nivel

Los circuitos cambiadores de nivel se caracterizan por llevar a la señal a un voltaje determinado, para lograr esto se valen del uso de un capacitor en la composición del circuito, como se muestra en la figura 2.11.

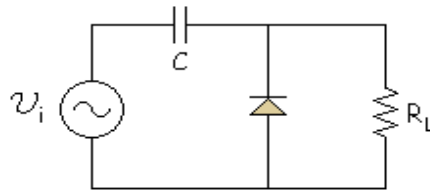


Figura2.11: Circuito cambiador de nivel

Fuente: http://www.sc.ehu.es/sbweb/electronica/elec_basica/tema4/Paginas/Pagina19.htm

Este capacitor funciona de la siguiente manera:

- En el semiciclo negativo: el diodo se comportará como cortocircuito permitiendo el paso de la corriente, con lo el capacitor se procederá a cargar
- En el semiciclo positivo: el diodo trabajará como circuito abierto, por lo que procederá a descargarse el capacitor, dándole de esta manera el voltaje con que se cargó a la salida, por lo que la señal de entrada se verá incrementada en el valor que provenga del capacitor dando como resultado la señal de salida.

2.2.3 TRANSISTOR

El transistor es un dispositivo electrónico semiconductor que cumple funciones de amplificador, oscilador, conmutador o rectificador (Transistor, 2011).

A continuación se presentarán los principales tipos de transistores.

2.2.3.1 Transistor de Unión Bipolar

El transistor de unión bipolar o BJT, tiene como base de fabricación, elementos que sean semiconductores como lo son el Silicio y el Germanio. Para poder generar la configuración del transistor se procede a contaminar de forma controlada tres zonas, dejando dos de las mismas del mismo tipo, por ejemplo NPN o PNP.

Estas configuraciones se refieren a que la letra intermedia corresponde a la característica de la base y las otras al emisor y colector, lo que las hace diferentes entre sí es la cantidad de contaminación que recibe cada una.

Cabe recalcar que la zona N poseerá electrones y la zona P huecos, los cuales se consideran como cargas positivas, aunque en realidad implican la ausencia de carga negativa. En la figura 2.12 se muestra un modelo de un transistor bipolar de unión.



Figura 2.12: Transistor bipolar

Fuente: http://www.electronicagonzalez.com/3401_Transistor+Bipolar+NPN+Original.php

2.2.3.2 Transistor de Unión Unipolar o de Efecto de Campo

El transistor de unión unipolar, es también conocido como transistor de efecto de campo de unión o JFET (Transistor, 2011).

Su conformación está dispuesta por una barra de silicio de tipo N o P, la única diferencia entre estas dos posibles configuraciones es que para lograr el JFET de tipo N es necesario disponer al transistor de un contacto óhmico en los terminales.

El FET se caracteriza por ser un conjunto de transistores en los cuales su común denominador es que su comportamiento se basa en el campo eléctrico, el cual controla el paso de electrones por el canal de un material semiconductor, por lo que se puede decir que controla la corriente basándose en la tensión que se aplica al dispositivo.

Los FET son conocidos por su función como resistencia controlada por medio de una diferencia de potencial determinada. Poseen tres terminales, llamados: compuerta (*gate*), drenaje (*drain*) y fuente (*source*). La puerta es el terminal equivalente a la base del BJT.

Como ya se había planteado el transistor de efecto de campo funciona como un interruptor el cual es controlado por una tensión determinada, donde el voltaje que se aplica a la puerta permite o no el flujo de corriente entre el drenaje y la fuente (Camarena & Flores, 2009).

La figura 2.13 presenta el símbolo del FET utilizado en los circuitos electrónicos, con su respectiva configuración.

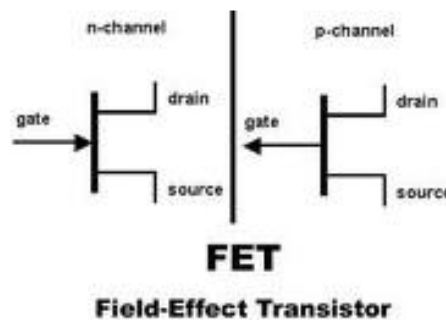


Figura 2.13: Configuración de un FET

Fuente: <http://integrated-thoughts.blogspot.com/2011/05/bjt-vs-fet-jfet-mosfet.html>

Como se indicó anteriormente, los transistores bipolares se dividen en NPN y PNP, de la misma manera los FET también presentan una clasificación la cual es de dos tipos: canal n y canal p, esto depende de la aplicación de una tensión positiva en la puerta, lo cual pondrá al transistor en estado de conducción o no conducción.

El transistor de efecto de campo de unión, como su nombre lo indica basa su funcionamiento en el campo eléctrico que se forma a partir de los valores de tensión aplicados a los terminales, básicamente al voltaje entre S (fuente) y G (compuerta), por lo que se puede decir que los valores de salida del transistor dependerán directamente de los de entrada del mismo, los cuales lograrán que se defina la curva característica en la que se delimitarán las tres zonas en las cuales puede trabajar el transistor: corte, activa y saturación.

2.2.3.3 Transistor de efecto de campo MOS

El transistor de efecto de campo MOS (MOSFET), donde MOS se refiere a un semiconductor de Óxido Metálico, en el cual se basa el funcionamiento de este tipo de transistores, los cuales presentan una compuerta metálica que se presenta aislada del canal del MOS por una capa de óxido.

La microelectrónica actual se encuentra basada en el uso de este transistor debido a sus características, por lo que los transistores MOSFET son los elementos principales de los circuitos integrados disponibles en el mercado.

El MOSFET presenta una construcción basada en un sustrato de material semiconductor dopado, el cual se debe dividir en tres partes: dos áreas opuestas separadas por una región formada por un dieléctrico, todo esto recubierto por una capa de material conductor. Para realizar esta división es necesaria la ejecución de procesos de difusión de dopantes los cuales generan las diferentes áreas mencionadas.

Los transistores MOSFET se clasifican en dos tipos fundamentales dependiendo de cómo se haya realizado el dopaje:

- Tipo nMOS: Sustrato de tipo p y difusiones de tipo n.
- Tipo pMOS: Sustrato de tipo n y difusiones de tipo p.

La figura 2.14 muestra la configuración de los MOSFET canal N y canal P.

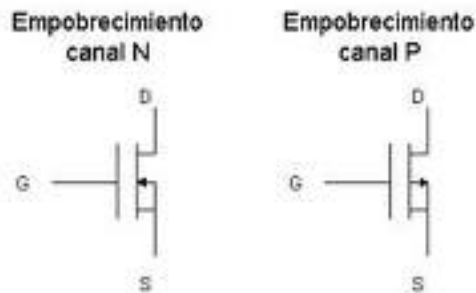


Figura 2.14: Configuración de un MOSFET

Fuente: <http://myspaceelectronic.blogspot.com/2010/09/transistores.html>

Así como en los JFET los terminales o áreas de difusión tienen las mismas denominaciones: fuente (*source*) y drenaje (*drain*), y el conductor entre ellos es la compuerta (*gate*).

Otra característica del transistor MOSFET es que posee tres estados de funcionamiento:

- **Estado de corte:** se da al aplicar una tensión a la compuerta que sea de igual valor que la que posee el sustrato. En este estado el MOSFET no conduce, es decir no existe corriente entre la fuente y el drenaje.
- **Conducción lineal:** en este estado existe una diferencia de potencial entre el drenaje y la fuente lo que produce la existencia del flujo de una corriente por lo que el transistor se encuentra conduciendo.

Para llegar a este estado, el transistor debe pasar por la polarización de la puerta con una tensión negativa (pMOS) o positiva (nMOS) lo cual genera una separación entre la fuente y el drenaje, en este espacio aparecen tanto electrones, en dopaje pMOS, como huecos, o si está dopado en nMOS, lo que da paso a la diferencia de potencial antes mencionada. Es también importante conocer que el transistor pasa a trabajar como una resistencia controlada por la tensión aplicada a la puerta.

- **Saturación:** este estado se da luego de que el voltaje entre la fuente y el drenaje superan un valor determinado, por lo que el canal de conducción de la compuerta desaparece en el área del drenaje. Pero aunque se de este fenómeno la corriente entre drenaje y fuente no desaparece ya que la misma se da por campo eléctrico y por eso no es afectada en la interrupción, de esta manera ya no depende de la diferencia de potencial entre estos terminales.

En el siguiente capítulo se presentará el programa de simulación SIMULINK de MATLAB, sus fundamentos y características principales que serán aplicadas para el posterior desarrollo de las prácticas o demostraciones de circuitos electrónicos en la asignatura de Electrónica I.

CAPÍTULO 3: PROGRAMA DE SIMULACIÓN SIMULINK DE MATLAB

MATLAB es un lenguaje de programación de alto rendimiento, el cual integra computación, visualización y programación en un contexto fácil de usar donde los problemas y las soluciones pueden ser expresados de manera más amigable para los usuarios.

Es un sistema interactivo que se basa en arreglos de elementos que no requieren dimensionamiento. Esto permite resolver muchos problemas de programación, especialmente los que tienen matrices y vectores, en una fracción de tiempo de la que le tomaría a alguien escribir un programa ingresando un escalar en un lenguaje no interactivo como lenguaje C o Fortran.

El nombre MATLAB proviene de *Matrix Laboratory* (Laboratorio de Matrices). Originalmente fue escrito para proveer una manera fácil de acceder al software de matrices desarrollados por los proyectos LINPACK y EISPACK (Introducción al Matlab).

Hoy en día, MATLAB incluye las librerías de LAPACK y BLAS, incorporando lo último en software para programación de matrices.

MATLAB y Simulink son los principales paquetes de *software* técnico tanto para la industria como para la educación. La versión de estudiantes de MATLAB y Simulink posee las mismas características que la versión profesional de este *software*, sin limitaciones, y la completamente funcional al igual que la versión profesional de MATLAB, con la capacidad de hacer modelos hasta de 1000 bloques. La versión para estudiantes da un acceso inmediato a cálculos computarizados de alto rendimiento haciendo modelos y simulaciones.

Este programa permite enfocarse en el curso del trabajo y aplicaciones en lugar de los detalles de la programación. Permite resolver varios problemas numéricos en fracciones de tiempo comparado con lo que toma escribir un programa de lenguaje de programación de bajo nivel como el lenguaje C, C++ o Fortran. Además, ayuda a entender y aplicar los conceptos en aplicaciones que van desde la ingeniería y las matemáticas a la química, la biología y la economía.

Este programa de simulación es interactivo y permite modelar y analizar sistemas dinámicos, tales como controles, procesamiento de señales, comunicaciones y otros sistemas complejos. La Herramienta de Símbolos Matemáticos, también incluida en la versión de estudiantes, se basa en Maple 8, una herramienta simbólica matemática de propósito general capaz de realizar cálculos simbólicos, algebraicos y de programación algebraica (Matlab).

Los productos de MATLAB se utilizan en una amplia gama de industrias, incluyendo automotriz, aeroespacial, electrónica, telecomunicaciones, medio ambiente, periféricos informáticos, finanzas y medicina. Más de un millón de técnicos profesionales de empresas tecnológicas, laboratorios de investigación gubernamentales, instituciones financieras, y en más de 3500 universidades, confían en MATLAB y SIMULINK como las herramientas fundamentales para sus trabajos científicos y de ingeniería.

Los usos más frecuentes de MATLAB son:

- Matemáticas y computación
- Desarrollo de Algoritmos
- Adquisición de datos
- Realización de modelos, simulaciones y prototipos
- Análisis, exploración y simulación de datos
- Gráficos de tipo científico y para la ingeniería
- Desarrollo de aplicaciones, incluyendo interface graficas para los usuarios.

MATLAB ha evolucionado con el paso de los años gracias a los aportes de muchos usuarios. En las universidades es una herramienta que sirve para introducir e instruir en las clases avanzadas de matemáticas, ingeniería y ciencias. En la industria, MATLAB es una herramienta para la investigación, desarrollo y análisis de mayor productividad.

3.1 INSTALACIÓN DE MATLAB

Esta sección describe los requerimientos de sistema necesarios para instalar la versión de estudiantes de MATLAB y SIMULINK en una computadora con Windows. También brinda parte por parte las instrucciones para instalar y activar el software

3.1.1 REQUERIMIENTOS DEL SISTEMA

Los requerimientos para instalar el *software* de MATLAB en una computadora son los siguientes:

- Windows XP (Service Pack 1 or 2) o Windows 2000 (Service Pack 3 or 4)
- Procesador Intel Pentium III o posterior, procesador Celeron, o procesadores de la familia Intel Xeon; AMD Athlon / Duron / Opteron; o procesadores compatibles
- 256 MB RAM (se recomienda 512 MB RAM)
- Espacio mínimo en el disco de 500 MB
- 16-, 24-, o 32-bit OpenGL tarjeta aceleradora de gráficos
- CD-ROM o disco DVD drive
- E-mail (requerido), acceso a internet (recomendado) para la activación del producto.

3.1.2 ARCHIVOS MEX

Los archivos MEX son subrutinas enlazadas dinámicamente las cuales MATLAB puede cargar y ejecutar automáticamente. Estas proveen de un mecanismo por el cual se puede llamar a sus propias subrutinas creadas en lenguaje C y Fortran a MATLAB como si fueran desarrolladas en modo de función.

3.1.3 PROCESO DE INSTALACIÓN Y ACTIVACIÓN DE LA VERSIÓN DE ESTUDIANTES

Esta lista resume los pasos básicos para el procedimiento de instalación. Se puede realizar la instalación y activación simplemente siguiendo las instrucciones que van apareciendo en el programa de instalación de MATLAB, esto lleva a los siguientes procesos:

- Cerrar todas las copias que se estén ejecutando de MATLAB.
- Insertar el CD de instalación de MATLAB y Simulink en el CD-ROM, para comenzar el programa de instalación ejecutando el archivo setup.exe desde el CD.
- Leer la pagina de Bienvenida y luego darle click en **SIGUIENTE**.
- Ingresar el nombre de la institución educativa donde se usará el programa, se da click en **SIGUIENTE**.
- Revisar el acuerdo de licencia del programa y si se está de acuerdo con los términos, se selecciona **SI** y se da click en **SIGUIENTE**.
- Revisar las políticas de uso de la versión para estudiantes, si se está satisfecho con los términos, se selecciona **SI** y se da click en **SIGUIENTE**.
- Se escoge el modo de instalación. La instalación **Típica** instala todos los productos; la instalación **Personalizada** permite seleccionar que productos se instalarán. Luego de seleccionar el tipo de instalación, se da click en **SIGUIENTE**.
- El cuadro de diálogo de **Selección de Carpeta** permite especificar el nombre del archivo donde se quiere instalar MATLAB. También se puede dejar que el programa de instalación instale el *software* en una carpeta por defecto. Si la carpeta no existe, el instalador la crea. Para continuar con la instalación, se da click en **SIGUIENTE**.
- El cuadro de diálogo de **Confirmación** permite reafirmar las opciones de instalación seleccionadas. Para cambiar alguno de estos parámetros se puede dar click en el botón **ATRÁS**. Para proceder con la instalación se da click en **Instalar**.

3.1.4 INICIO DE MATLAB

En **Windows** para iniciar MATLAB se da doble click en el ícono de MATLAB que se encuentra en el escritorio de Windows.

En **Mac OS x** se inicia MATLAB dando doble click en el icono de MATLAB en el ícono que se encuentra en el escritorio.

En **Linux**, se inicia MATLAB escribiendo MATLAB en el programa de comandos del sistema operativo.

Se puede personalizar el inicio MATLAB: por ejemplo, se puede cambiar el directorio en donde se inicia MATLAB o que se ejecuten automáticamente los archivos de inicio de MATLAB creando un *script* que se llame startup.m.

3.1.5 ARCHIVOS DE MATLAB

MATLAB es un fuerte lenguaje de programación así como también un ambiente de programación interactivo. Los archivos que contienen el código en este lenguaje se llaman archivo-M, los cuales se pueden crear usando un editor de texto, luego usarlos como se haría con cualquier función o comando de este programa.

Existen 2 clases de archivos-M:

- Scripts, que no aceptan argumentos de entrada ni devuelven argumentos de salida.
- Funciones, que aceptan argumentos de entrada y devuelven argumentos de salida.

3.1.5.1 Scripts

Cuando se llama a un script, MATLAB simplemente ejecuta los comandos encontrados en un archivo. Los scripts trabajan con la información existente en el área de trabajo, o pueden crear

nueva información con la cual trabajar. Sin embargo los scripts no devuelven un argumento de salida, sino que las variables que crean permanecen en el área de trabajo, para ser utilizadas en los cálculos posteriores.

3.1.5.2 Funciones

Las funciones son archivos-M que pueden aceptar argumentos de entrada y devolver argumentos de salida. Los nombres de los archivos-M y los de la función deben ser los mismos para que pueda existir una compatibilidad a la hora de correrlos.

Las funciones trabajan en variables dentro de su propia área de trabajo, independientemente del área de programación de comandos de MATLAB.

Un buen ejemplo se lo puede tomar con la función *rank* (rango). El archivo-M *rank.m* se encuentra disponible en el directorio *toolbox/matlab/matfun* o también podemos acceder a este digitando *type Rank* en la línea de comandos de MATLAB a lo cual tendremos como respuesta lo que se observa en la figura 3.1.

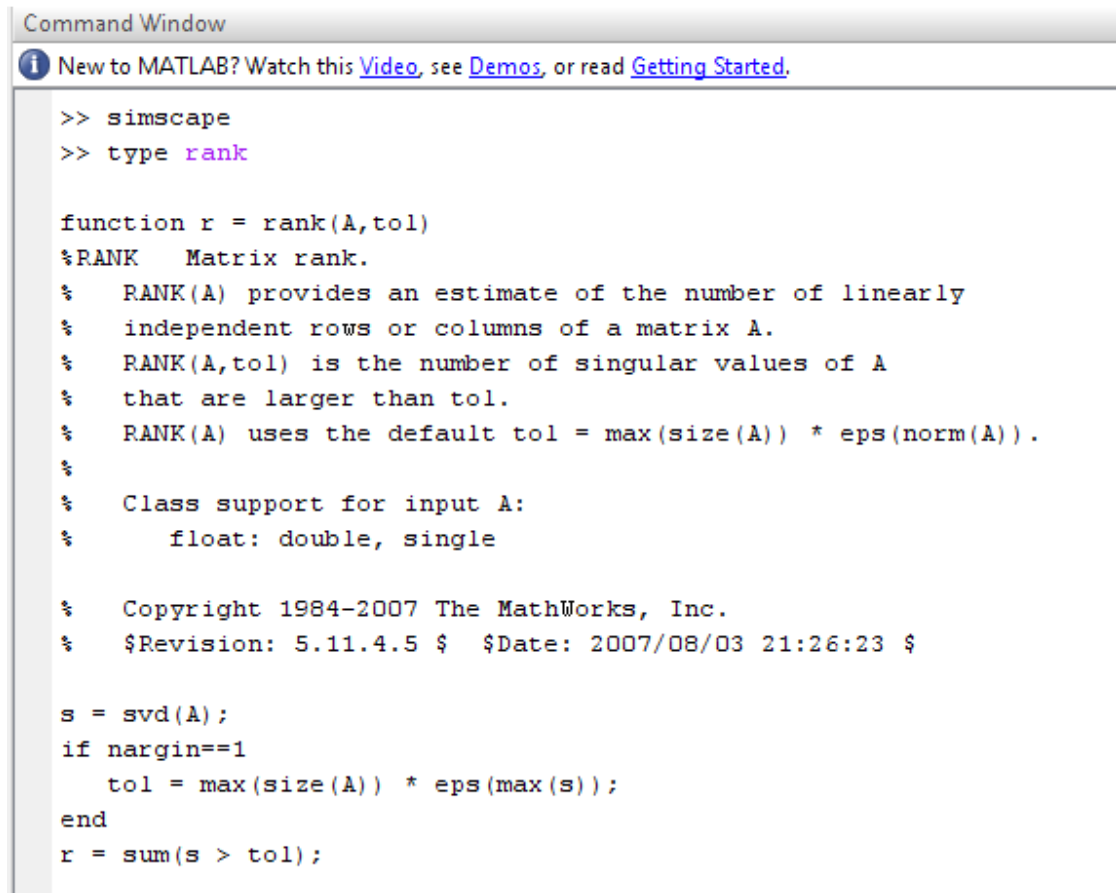
La primera línea de una función archivo-M comienza con el nombre que se le dará a la función así como otros argumentos necesarios para la función. En este caso en particular, hay 2 argumentos de entrada y uno de salida.

Las siguientes líneas de código, son las que proveen los textos de ayuda, es decir una breve explicación del contenido y desempeño de la función.

La primera línea del texto de ayuda es la línea H1, la cual MATLAB utiliza cuando se usa el comando *lookfor* o un requerimiento de ayuda a un directorio.

Luego del texto de ayuda, el resto de líneas del archivo son el código ejecutable de MATLAB que define la función. Las variables introducidas en el cuerpo de la función, así como las

variables de la primera línea (r, A, tol), pertenecen solo a esta función, por lo que no guardan relación con otras funciones que se lleguen a utilizar en el área de trabajo de MATLAB.



```
Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.

>> simscape
>> type rank

function r = rank(A,tol)
%RANK Matrix rank.
% RANK(A) provides an estimate of the number of linearly
% independent rows or columns of a matrix A.
% RANK(A,tol) is the number of singular values of A
% that are larger than tol.
% RANK(A) uses the default tol = max(size(A)) * eps(norm(A)).
%
% Class support for input A:
% float: double, single
%
% Copyright 1984-2007 The MathWorks, Inc.
% $Revision: 5.11.4.5 $ $Date: 2007/08/03 21:26:23 $

s = svd(A);
if nargin==1
    tol = max(size(A)) * eps(max(s));
end
r = sum(s > tol);
```

Figura 3.1: Respuesta de comando *type Rank*

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

Este ejemplo muestra un aspecto de las funciones de MATLAB que no se encuentra regularmente en otros lenguajes de programación y esto es que presentan un número variable de argumentos.

El rango de la función puede ser usado en muchas formas diferentes, como se muestra a continuación:

- rank(A)

- $r = \text{rank}(A)$
- $r = \text{rank}(A, 1.e-6)$

Muchos de los archivos-M trabajan de esta manera. Si no se ingresa ningún argumento de salida, el resultado se lo almacenara en una variable. Si el segundo argumento de entrada no es ingresado, la función otorga un valor por defecto. Para este propósito, en el cuerpo de la función, dos cantidades llamadas *nargin* y *nargout* están disponibles para indicar el número de argumentos de entrada y salida involucradas en cada uso particular de la función. La función rango usa *nargin* pero no usa *nargout*.

3.1.5.3 Tipos de funciones

MATLAB ofrece muchos tipos de funciones para ser usadas en la programación entre las que se encuentran:

Funciones anónimas: Una función anónima es una función de MATLAB que no requiere de un archivo-M y cual consiste en una expresión simple del programa y de cualquier número de argumentos de entrada y salida. Se pueden definir funciones anónimas desde la línea de comandos de MATLAB o junto al archivo-M de una función o *script*. Esto da ejemplos rápidos de cómo crear una función simple sin necesidad de crear un archivo-M cada vez que necesitemos de una función.

Sub-funciones: Anteriormente en MATLAB era obligatorio crear un archivo-M diferente para cada función que se iba a utilizar, el nombre de la función tenía que ser el mismo que el del nombre del fichero; esto cambió en la versión 5.0 cuando se introdujeron las sub-funciones, las cuales son funciones ya definidas por defecto en un archivo-M, que tienen nombres diferentes al del fichero de la función principal. Las sub-funciones solo pueden ser llamadas para ser usadas por las demás funciones que estén incluidas en el archivo M al que pertenecen y no por otras funciones externas.

Funciones Privadas: Las funciones privadas son aquellas que pueden ser llamadas a operar desde cualquier función externa, pero aun así presenta una restricción con ciertas funciones. Este tipo de funciones puede ser usado si se quiere tener acceso limitado a una función, o cuando se escoge no exponer la implementación de una de ellas. Estas funciones se definen en sub directorios llamados *private* y solo pueden ser llamadas por las funciones que se encuentran en el directorio principal llamado *private*. Cuando MATLAB realiza la búsqueda de nombres de directorios primero analiza las sub- funciones para luego proceder con las funciones primarias y finalmente con las funciones generales

Funciones anidadas: Se pueden definir funciones con el cuerpo de programación de cualquier otra función de archivo-M en MATLAB. A esto se le llama que hay una función anidada en una función exterior, la cual puede contener cualquier o todos los componentes de una función de archivo-M. Las funciones anidadas son usadas para tener mayor y mejor control en la visibilidad de las funciones al igual que la visibilidad que estas funciones tienen sobre las variables en el espacio de trabajo.

3.1.5.4 Creación de interfaces graficas para usuarios

A continuación se detallarán los aspectos más importantes para la creación de interfaces gráficas para usuarios:

GUIDE (*Graphical User Interface Development Environment*): es un ambiente de desarrollo para interfaz gráfica del usuario (GUI), provee una amplia gama de herramientas para la creación de las mismas, las cuales simplifican de gran manera el diseño y construcción de GUIs.

Se pueden usar las herramientas GUIDE para:

Diseñar una GUI: Usando el editor de diseño de GUIDE, se pueden diseñar GUIs fácilmente seleccionando y arrastrando componentes de las mismas, como son paneles, botones, campos de texto, barras de desplazamiento, menús y mas objetos, hacia el área de trabajo. GUIDE almacena el área extendida de GUI en un archivo FIG.

Programación de la GUI: La GUIDE automáticamente genera un archivo-M que controla como la GUI trabaja. El archivo-M inicia la GUI y contiene el área de trabajo para los componentes que han sido llamados más comúnmente a usarse.

Diseño de GUI: Para ingresar a GUIDE, se procede a escribir *guide* en la línea de comando de MATLAB. Esto nos llevara a un cuadro de dialogo de inicio rápido de GUIDE, como se muestra en la figura 3.2 a continuación:

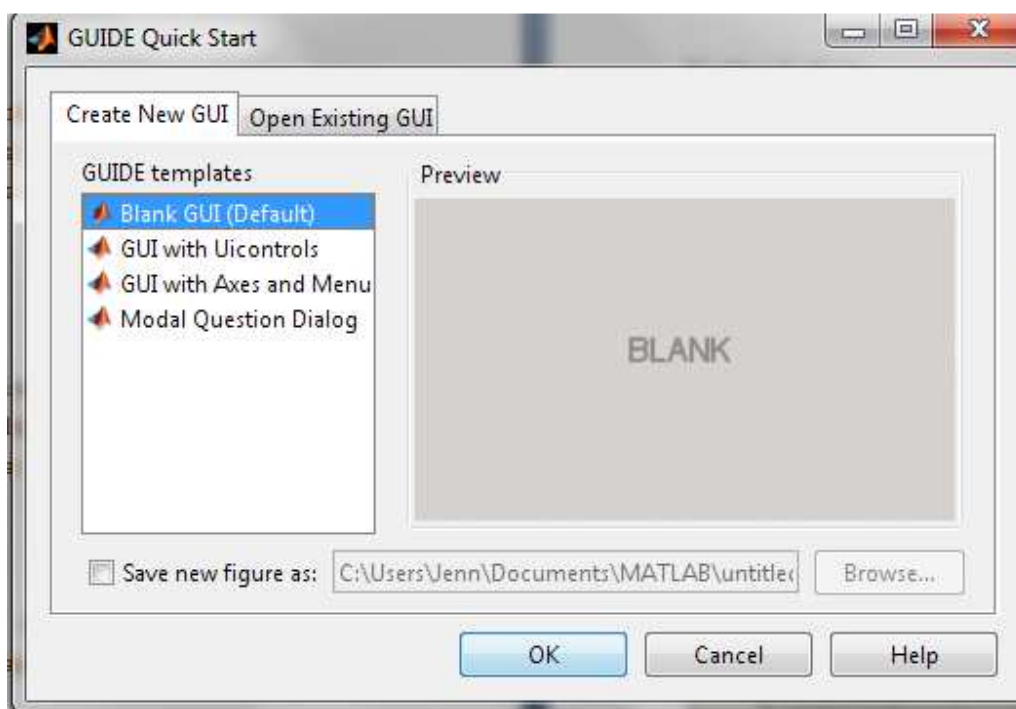


Figura3.2: Ventana de Inicio rápido de GUIDE

Función: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

Desde el inicio rápido de GUIDE se puede:

- Crear una nueva GUI desde unas de las plantillas de GUIDE, las cuales son GUIs pre-construidas que se pueden modificar según el propósito que se les quiera dar.
- Abrir una GUI existente.

El editor de diseño: Al elegir la opción de GUIDE en blanco, aparecerá una nueva ventana la cual contiene el editor de diseño, este consiste en un panel de control con todas las herramientas de GUIDE. La figura 3.3 a continuación muestra una plantilla en blanco de GUI.

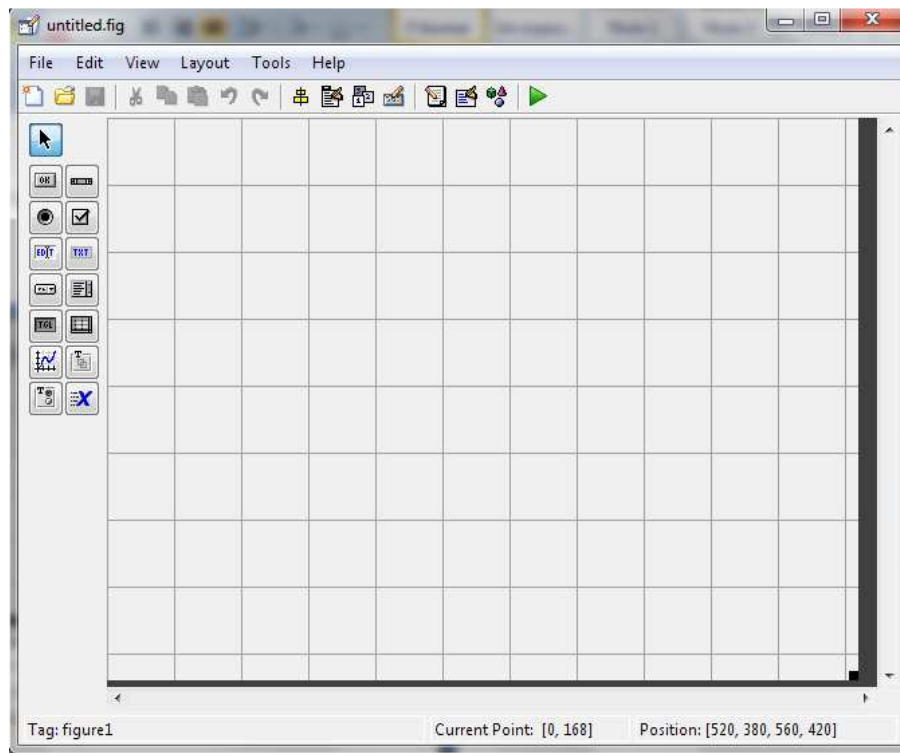


Figura 3.3: Ventana GUIDE en blanco

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

Se pueden diseñar las GUI arrastrando los componentes, como paneles, botones, menús o ejes, desde la paleta de componentes, en el lado izquierdo del editor de diseño en el área de diseño. Como se muestra en la figura 3.4, si se arrastra un botón, este aparecerá en el área de diseño, para el cambio de configuración de los elementos solo es necesario hacer doble click sobre los mismos y se abrirá una nueva ventana en la cual se puede proceder a este proceso.

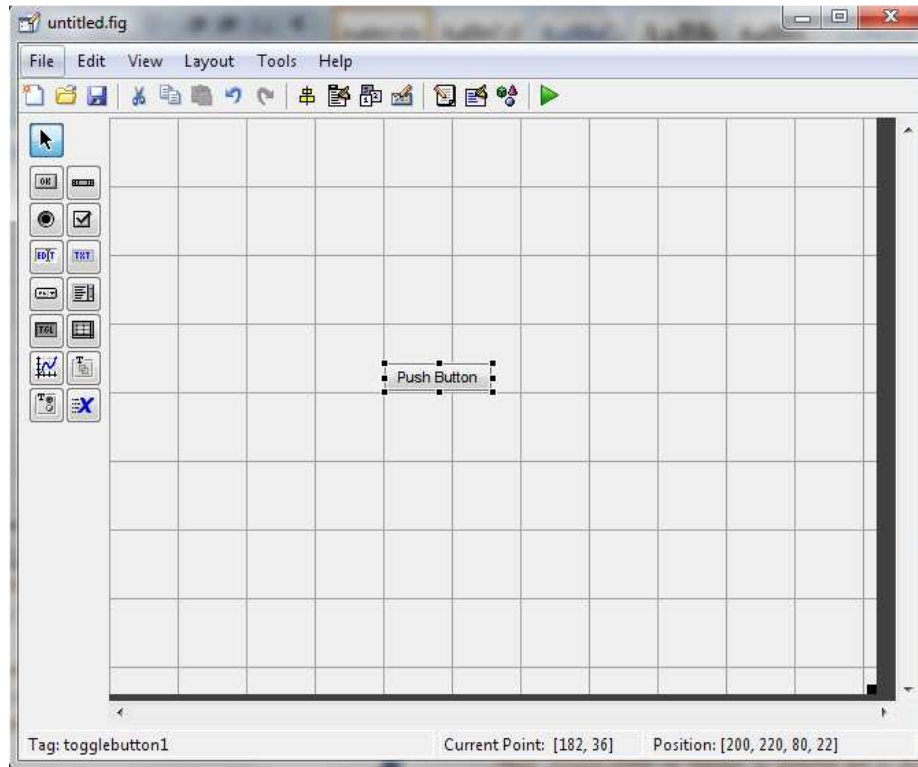


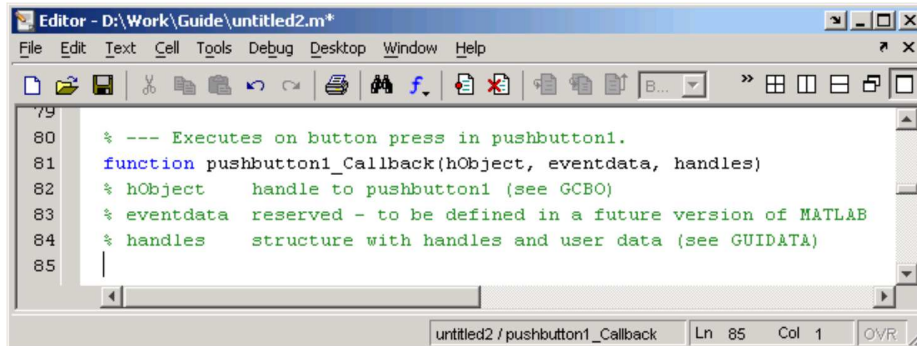
Figura 3.4: Construcción de un modelo GUIDE

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

Se puede usar el editor de diseño para crear menús y configurar propiedades básicas para los componentes de la GUI

Programando una GUI: Después de diseñar una GUI y configurar las propiedades de los componentes, el siguiente paso es programar la misma. Se programan las GUI con el código de una o más llamadas de respuesta para cada uno de estos componentes. Las llamadas de respuesta son funciones que se ejecutan en respuesta a alguna acción del usuario. Una acción frecuente es dándole doble click a un botón se cargan automáticamente las llamadas de respuesta que se encuentran en un archivo-M que genera GUIDE automáticamente. GUIDE agrega plantillas para las llamadas de respuesta más usadas para este archivo-M, pero se pueden agregar otros. Se puede usar el editor de archivos-M para cambiar este archivo.

La figura 3.5 muestra la llamada de respuesta para un botón.



```
Editor - D:\Work\Guide\untitled2.m*
File Edit Text Cell Tools Debug Desktop Window Help
untitled2 / pushbutton1_Callback Ln 85 Col 1 OVR

79
80 % --- Executes on button press in pushbutton1.
81 function pushbutton1_Callback(hObject, eventdata, handles)
82 % hObject    handle to pushbutton1 (see GCBO)
83 % eventdata  reserved - to be defined in a future version of MATLAB
84 % handles    structure with handles and user data (see GUIDATA)
85
```

Figura 3.5: Ventana GUIDE en blanco

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

3.2 SIMULINK

SIMULINK es un paquete de *software* diseñado para el modelado, simulación y análisis de sistemas dinámicos. Este programa soporta sistemas lineales y no lineales, modela en tiempo continuo así como también en tiempo de prueba o en un híbrido que mezcla los dos. Estos sistemas pueden tener múltiples tasas de procesamiento, por ejemplo puede poseer diferentes partes las cuales pueden ser muestreadas o actualizadas en diferentes tasas.

Además permite que el usuario pueda construir sus propios modelos desde cero, así como también como la inserción de modelos existentes a sus proyectos.

Las simulaciones que presenta son interactivas, por lo que el usuario puede realizar cambios en los parámetros durante la simulación y observar las variaciones que se producen.

Una de las ventajas de esta aplicación es que el usuario posee acceso inmediato a todas las herramientas de análisis de MATLAB, por lo que permite el manejo y visualización de resultados.

La meta de SIMULINK es ser práctico y confiable para el usuario, debido a que es un programa usado alrededor del mundo por ingenieros y estudiantes para resolver problemas de todo ámbito, es muy importante saber el uso de todas sus herramientas.

3.2.1 SIMULINK, UNA HERRAMIENTA DE DISEÑO DE MODELOS.

SIMULINK permite que el usuario experimente con modelos más realísticos y no solo con modelos lineales como otros programas, estos modelos abarcan la resistencia del aire, los factores de fricción, la velocidad, y otros fenómenos que se dan en la naturaleza. En sí, este *software* se convierte en un laboratorio portable al poder modelar y analizar sistemas que se perfilan como imposibles o poco prácticos de simular físicamente, estos incluyen el comportamiento de los elementos de un vehículo, la dinámica de movimiento de un avión y hasta los efectos de cambios en sistemas monetarios.

Para modelado, SIMULINK provee una interface de usuario grafico (GUI) para construir modelos como diagramas de bloque, usando operaciones muy sencillas para el usuario. Con GUI, el usuario puede llegar a dibujar los modelos como si lo estuviera haciendo con papel y lápiz.

SIMULINK incluye librerías de bloques de conectores, fuentes y componentes lineales y no lineales. El usuario también puede personalizar y crear sus propios modelos de bloques. Este acercamiento con el usuario le permite conocer como un modelo se organiza internamente y como sus partes interactúan. Luego de definir un modelo, se puede proceder a simularlo, usando una gama de métodos de integración ya sean pertenecientes a los menú de SIMULINK o ingresando comandos en las ventanas de MATLAB. Los menús son particularmente convenientes para un trabajo interactivo, mientras que las líneas de comando son más útiles a la hora de ejecutar bandejas de simulaciones. Al usar la opción de escalas o bloques de muestra, el usuario podrá observar los resultados de la simulación mientras esta se está procesando. Además, permite el cambio de parámetros lo cual se reflejará inmediatamente en la pantalla de muestra. Los resultados de la simulación pueden ser exportados al área de trabajo de MATLAB para un procesamiento posterior así como su visualización.

Debido a que SIMULINK y MATLAB están integrados, el usuario podrá simular, analizar y revisar sus modelos creados en cualquiera de las dos aplicaciones.

3.2.2 INICIO EN SIMULINK

Para poder acceder a SIMULINK, se debe ingresar a MATLAB, y seguir cualquiera de los dos pasos indicados a continuación:

- Dar click en el icono de Simulink en la barra de herramientas de MATLAB
- Escribir el comando Simulink en la línea de comandos de MATLAB



La figura 3.6 muestra la forma de ingreso a SIMULINK.

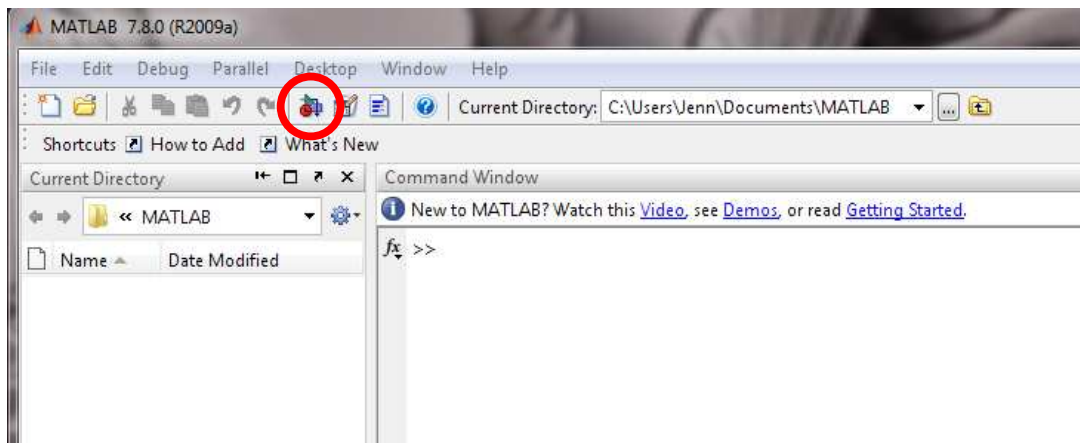


Figura 3.6: Forma de ingreso a Simulink

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

Luego de esto, aparecerá el buscador de librerías de Simulink, en el cual se podrá encontrar las librerías de bloques que se presentan organizadas en una estructura de árbol. El usuario podrá crear modelos copiando los bloques desde este buscador hacia la ventana del modelo.

En la figura 3.7 se puede observar la Ventana del buscador de librerías de SIMULINK.

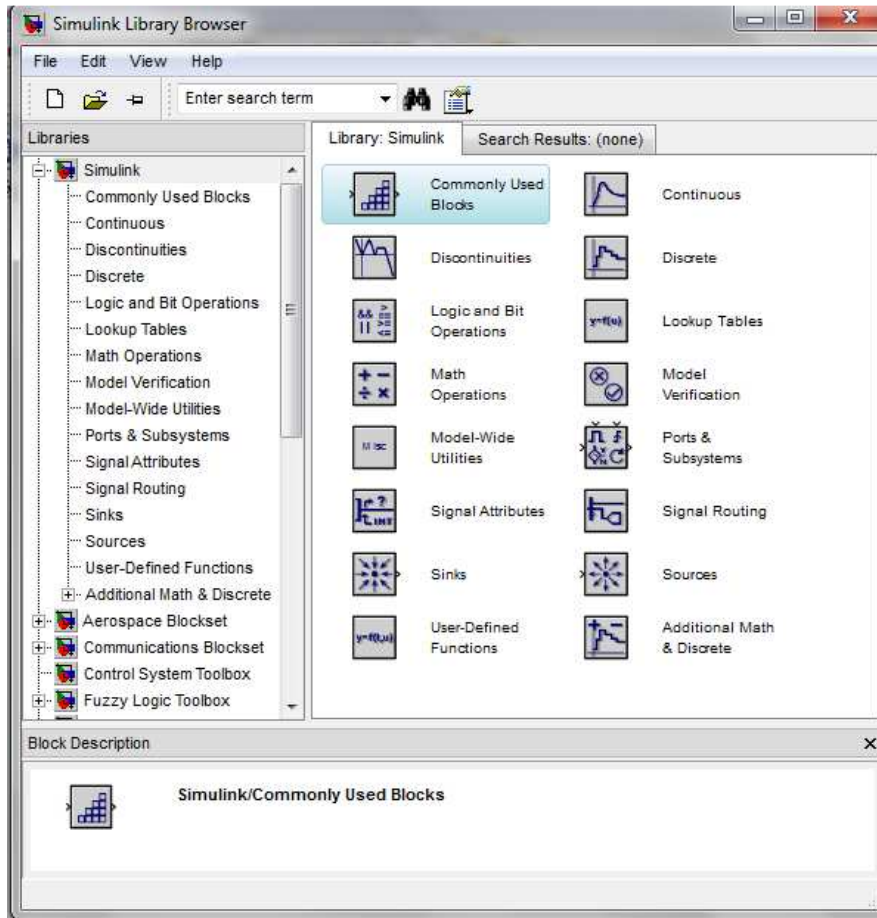


Figura3.7: Ventana del buscador de librerías de Simulink

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

Las librerías que más se usarán en el desarrollo de las prácticas para la asignatura Electrónica I son las siguientes:

Librería Sources: Esta librería, cuyos elementos se muestran en la figura 3.8, contiene una serie de fuentes de datos o señales los cuales son de gran utilidad a la hora de generar sistemas dinámicos, entre las fuentes que se pueden encontrar se tienen de entrada constante, de onda sinusoidal, paso, repetición de secuencia, rampa, etc.

Otro elemento característico de esta librería es *Ground* el cual es utilizado mayormente para conectar puertos que no han sido utilizados en el proyecto y de esta manera evitar alertas.

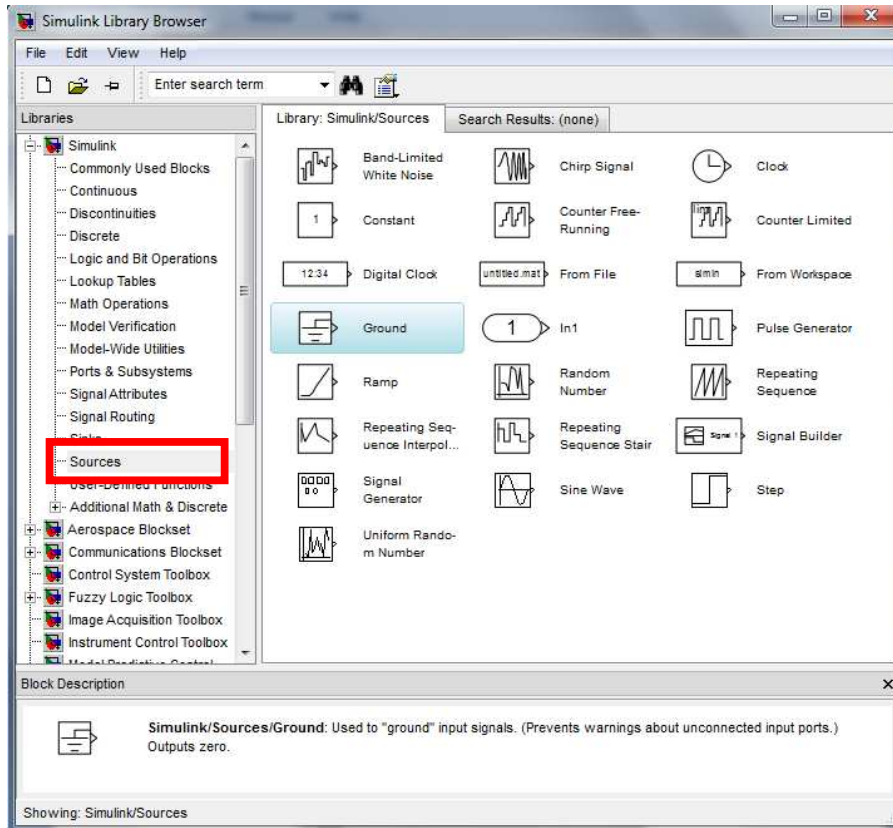


Figura3.8: Elementos Librería *Sources*

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

Librería *Sink*: Esta librería, cuya ventana se muestra en la figura 3.9, presenta una serie de bloques los cuales serán usados para mostrar las señales que circulan por el circuito construido en cualquier instancia del mismo.

Los bloques de esta librería que se utilizarán para este trabajo son los siguientes:

- ***Display*:** muestra valores de voltaje, corriente, potencia, etc. en las salidas de los elementos de manera numérica.
- ***Scope*:** muestra las señales que circulan por el circuito en cualquier instancia de manera gráfica.

Las gráficas y valores que se generan en estos bloques pueden ser almacenados en un archivo distinto del modelo en construcción para posterior uso del usuario.

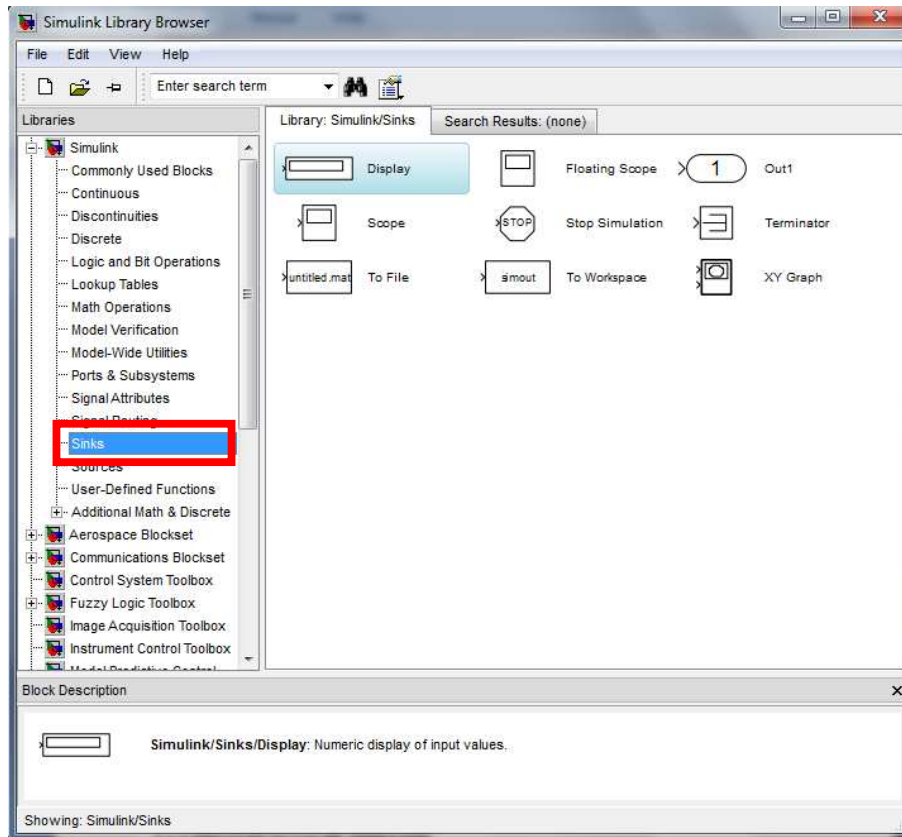


Figura3.9: Librería Sinks

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

Librería *Signal Routing*: Esta librería presenta una serie de buses de información los cuales pueden ser utilizados para la integración de diferentes señales en una sola así como el efecto contrario, tener una mejor imagen de presentación del diseño en lo que a cableado respecta.

Los bloques de esta librería que se utilizarán para este trabajo son los siguientes:

- Mux/Demux: divide/une señales en dos o más para la conexión con otros bloques.
- Goto/From: envía/recibe señales de bloques con la misma designación, evitando el cableado repetitivo hacia elementos en común.

La figura 3.10 muestra la librería *Signal Routing*.

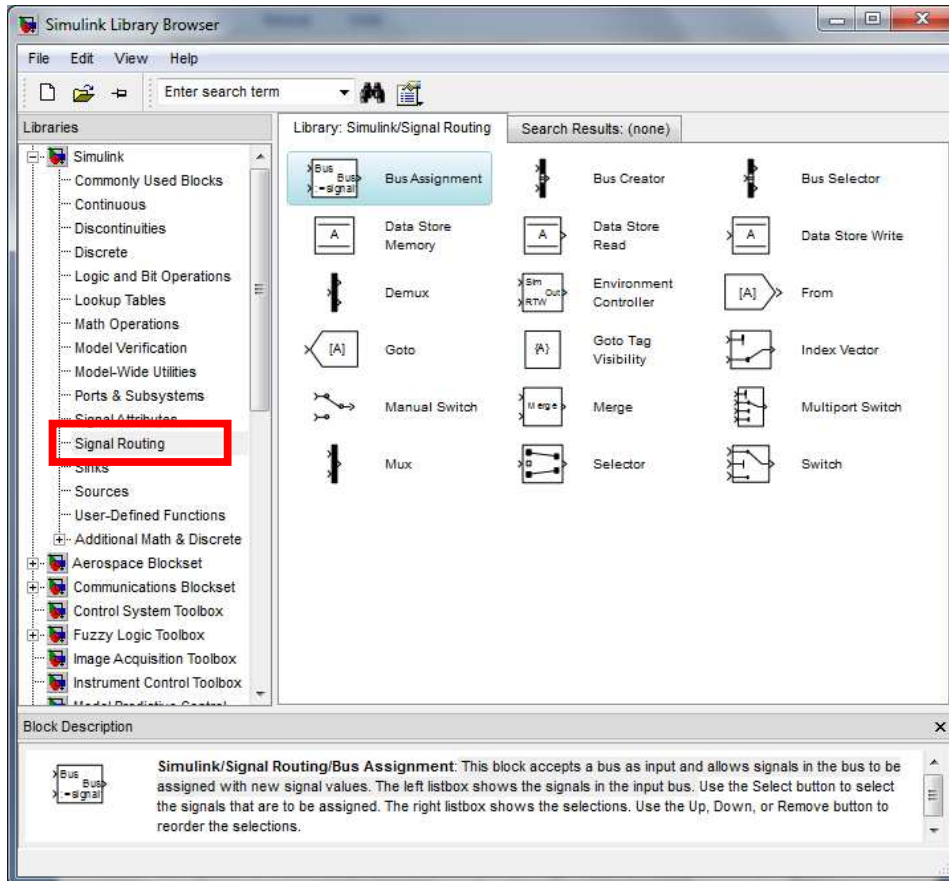


Figura3.10: Librería *Signal Routing*

Fuente: Print Screen de Matlab Version 7.8.0.347 (R2009a)

Para realizar las prácticas de Electrónica I son necesarios una serie de elementos, como son diodos, transistores fuentes eléctricas, etc., los cuales se pueden encontrar en la lista de librerías en la aplicación SimPowerSystems la cual se explicará más ampliamente a continuación.

SimPowerSystems: es un conjunto de librerías en las cuales se unifica los bloques representativos de elementos necesarios en el desarrollo de simulaciones para circuitos electrónicos, eléctricos y de potencia, incluyendo transformadores, líneas y máquinas, de una manera sencilla y rápida. Debido a su formato de trabajo basado en *clicks* y arrastre de elementos se presenta como una solución dinámica para el propósito de este trabajo, se necesitará

únicamente determinar los bloques necesarios y asignarles los parámetros con los cuales se desea trabajar, realizar las conexiones necesarias y por último visualizar los resultados para un análisis más profundo de la materia. La ventana correspondiente a esta librería se presenta en la figura 3.11.

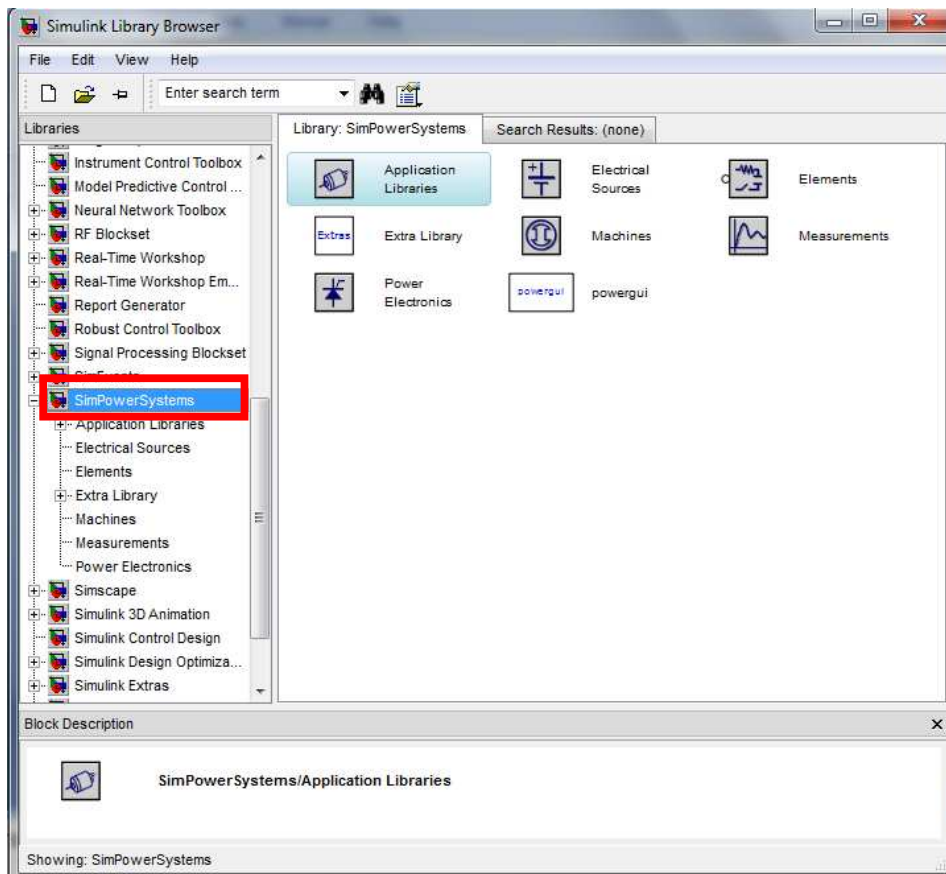


Figura 3.11: SimPowerSystems

Fuente: Print Screen de Matlab Version 7.8.0.347 (R2009a)

Las principales librerías de SimPowerSystems son las siguientes:

- **Fuentes eléctricas:** Teniendo en cuenta que las fuentes que se usarán en las prácticas son: *AC Current Source* (Fuente de corriente alterna), *AC Voltage Source* (Fuente de voltaje alterna) y *DC Voltage Source* (Fuente de voltaje continua). La ventana respectiva se puede observar en la figura 3.12.

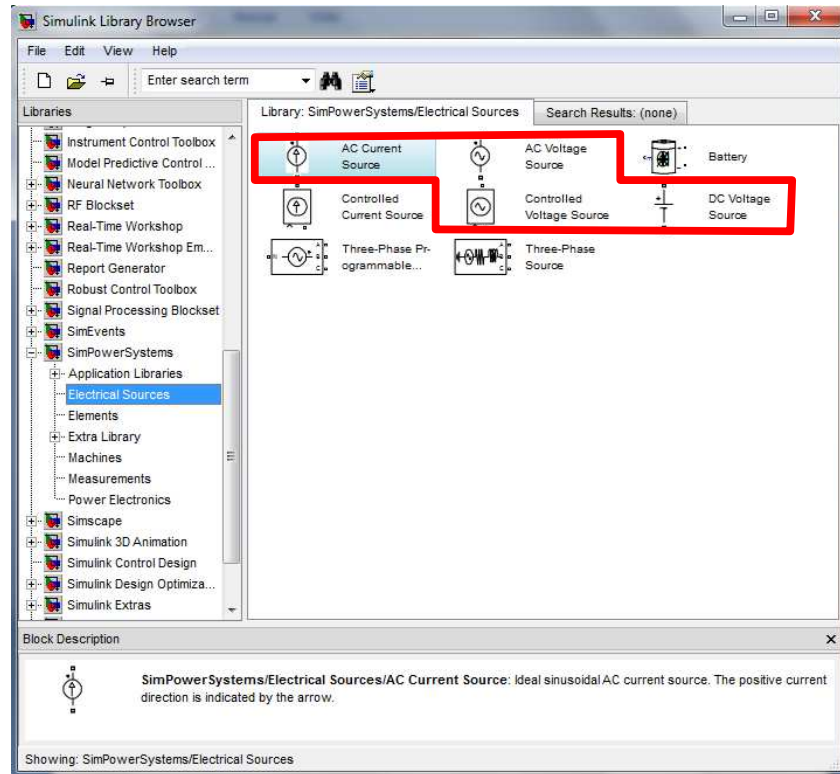


Figura3.12: Librería Fuentes Eléctricas

Fuente: Print Screen de Matlab Version 7.8.0.347 (R2009a)

- **Elementos:** Teniendo en cuenta que los elementos que se usarán en las prácticas son: *Parallel RLC Branch* o *Series RLC Branch*, en ambos al ingresar en la configuración del bloque se puede proceder a escoger con cuál de los tres elementos se desea trabajar o que combinación entre los tres.

En esta librería también se encontrará *Ground* (Tierra), elemento necesario para el aterrizaje del circuito. Esta librería se muestra en la figura 3.13.

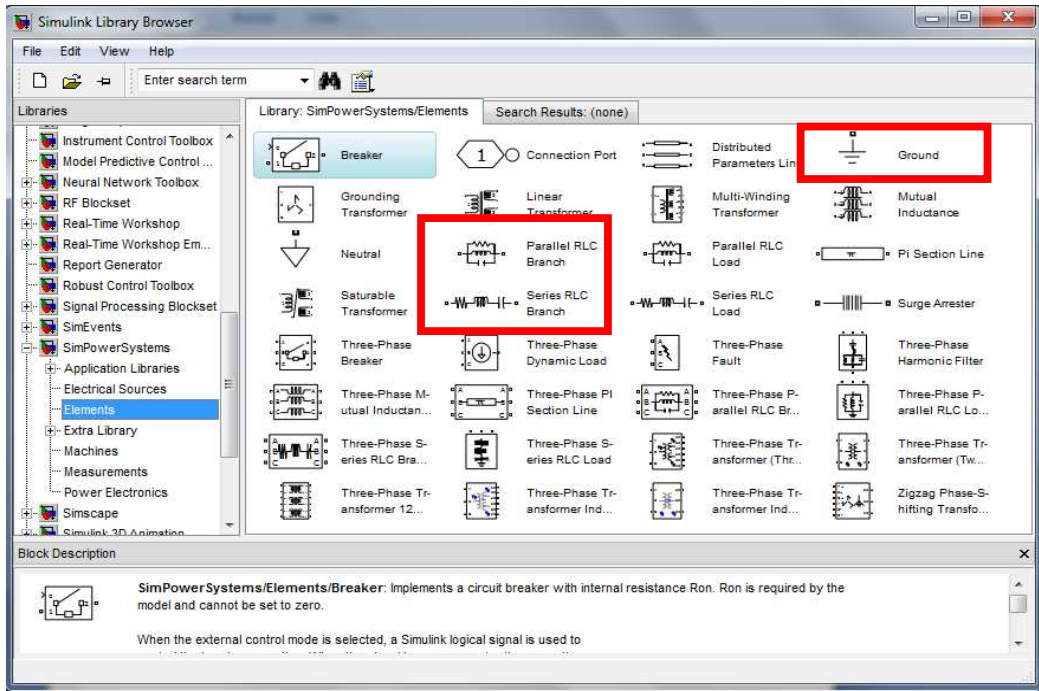


Figura3.13: Librería Elementos

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

- **Mediciones:** Teniendo en cuenta que los elementos de medición que se usarán en nuestras prácticas son: *Current Measurement* (Amperímetro), *Multimeter* (Multímetro) y *Voltage Measurement* (Voltímetro), los cuales mostrarán los parámetros de voltaje y corriente en cualquier punto del circuito, si la salida de estos bloques se conecta a un *Display*, se podrá observar el valor de los mismos, y si se lo conecta a un *Scope* se observará las ondas de estos parámetros.

En la figura 3.14 se muestra la ventana correspondiente a la Librería Mediciones.

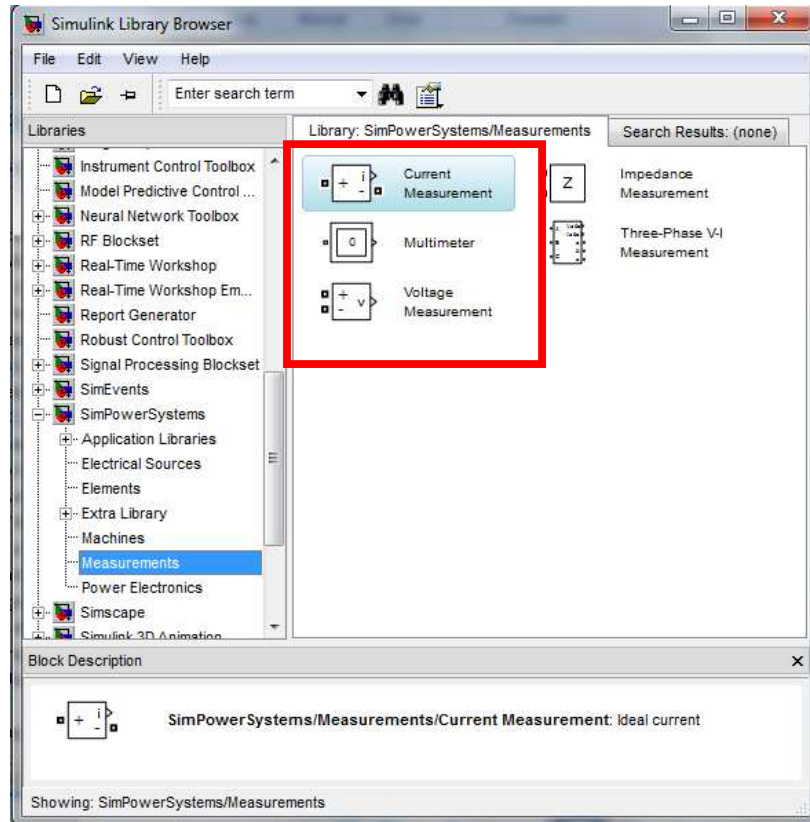


Figura3.14: Librería Mediciones

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

- Electrónica de Potencia: Considerando que los bloques de esta librería que se usarán en las prácticas son: *Diode*, *Mosfet* y *Universal Bridge* (Puente Universal) en el cual al ir a sus propiedades se podrá escoger si se desea que trabaje como Diodo o como un híbrido Diodo/MOSFET.

La ventana correspondiente a esta librería se presenta en la figura 3.15.

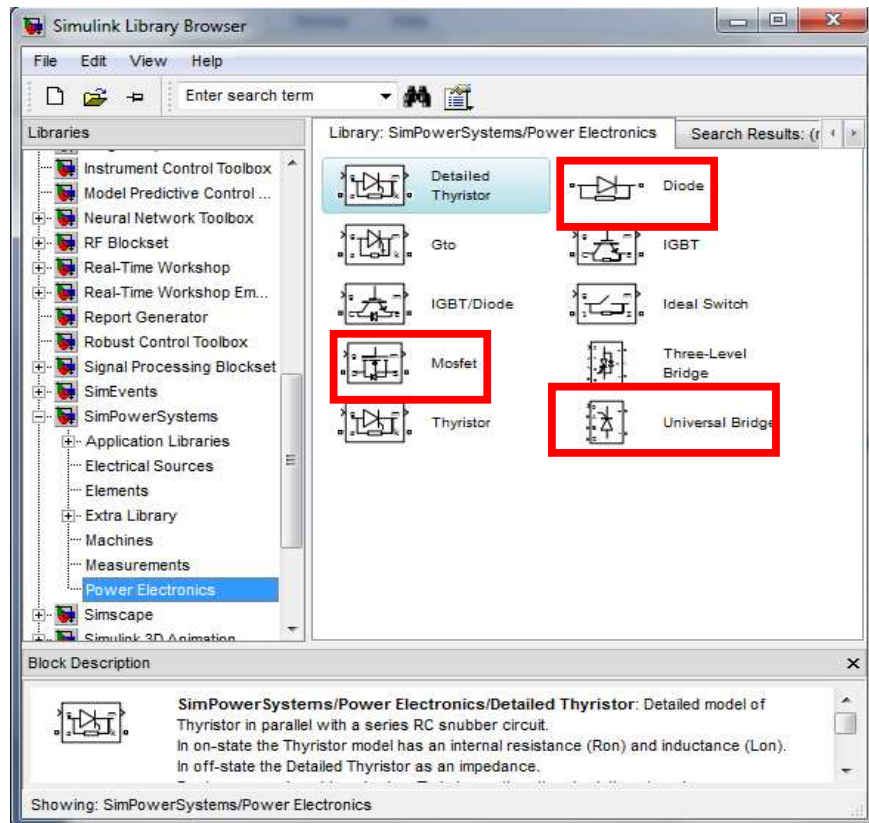


Figura3.15: Librería Electrónica de Potencia

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

- **Bloque PowerGui:** Este bloque debe estar presente en todo modelo generado bajo los bloques de SimPowerSystems, caso contrario se obtendrá como respuesta un mensaje de aviso que no puede ejecutarse la simulación.

Este bloque es necesario debido a que es el que almacena un equivalente del circuito en términos entendibles para Simulink así como las ecuaciones de espacio que están involucradas en el modelo.

Este bloque puede estar colocado en cualquier lugar del área de trabajo, solo puede existir uno en el modelo y no debe estar conectado con ningún otro componente del mismo.

3.2.3 PREFERENCIAS DE SIMULINK

La opción de Preferencias en MATLAB permite especificar la configuración de algunas opciones de Simulink. Para poder observar la ventana de Preferencias, seleccione esta opción en el menú de Archivo de SIMULINK.

En la figura 3.16 se presenta la ventana de Preferencias Generales.

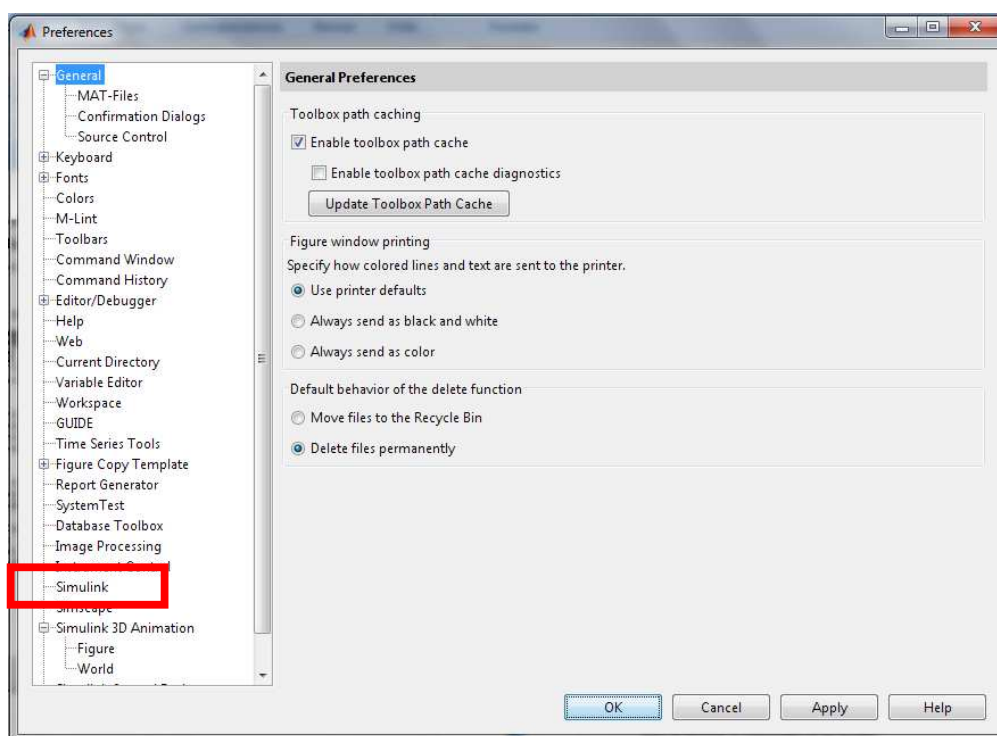


Figura 3.16: Ventana de Preferencias Generales en Matlab

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

3.2.4 TIPOS DE PREFERENCIAS SIMULINK

Para observar las preferencias específicas de Simulink, se debe escoger esta opción en la clasificación que se puede observar del lado izquierdo de la ventana correspondiente o estando ya en Simulink se puede ingresar mediante la ruta Archivos/Preferencias.

Luego de escoger esta opción se podrá observar un cambio del área derecha, la cual indicará las diferentes configuraciones de preferencias de SIMULINK.

En la figura 3.17 se presenta la ventana de Preferencias de SIMULINK.

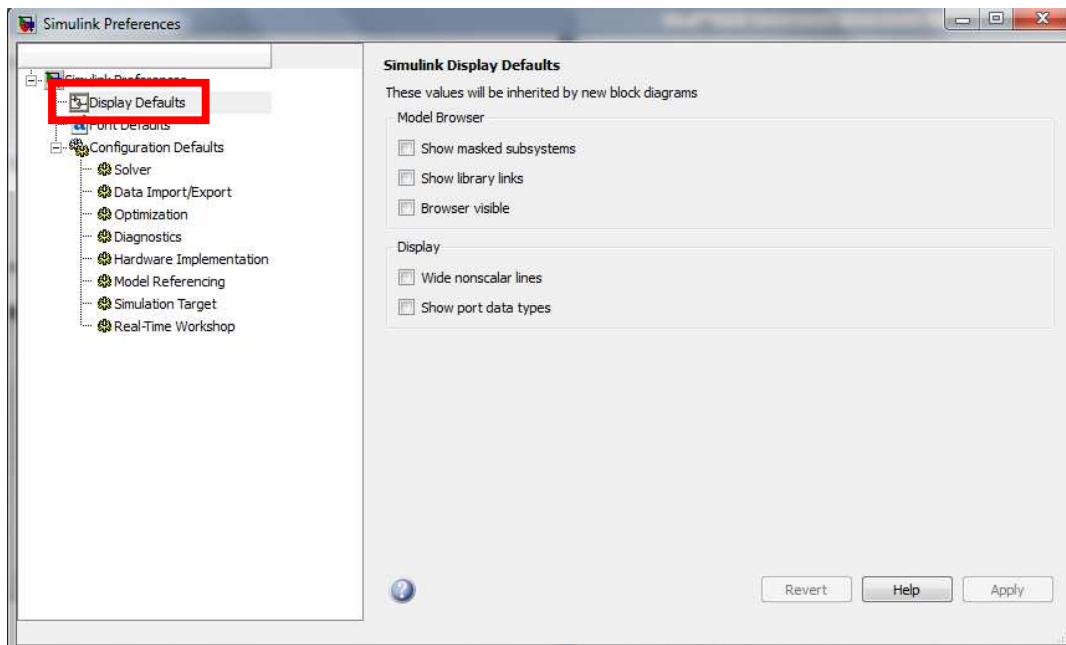


Figura3.17: Ventana de Preferencias de SIMULINK

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

En esta ventana se puede encontrar las siguientes Preferencias SIMULINK:

- *Model Browser*: en este grupo se podrá escoger si se quiere que se muestre el buscador al momento de abrir un modelo, así como elegir que el buscador muestre bloques importados de subsistemas y el contenido de subsistemas “*masked*” (subsistemas desarrollados que se presentan ya terminados en un solo ícono para fácil acceso).
- *Display*: en este grupo se puede escoger si se desea usar líneas gruesas en conexiones no-escalares entre bloques así como si se quiere mostrar los tipos de puertos en los diagramas de bloque.

3.2.4.1 Fuente

Esta ventana permite escoger el tipo de fuente que se desea usar para las etiquetas de bloques y líneas así como para las anotaciones de modelo.

La figura 3.18 muestra esta ventana.

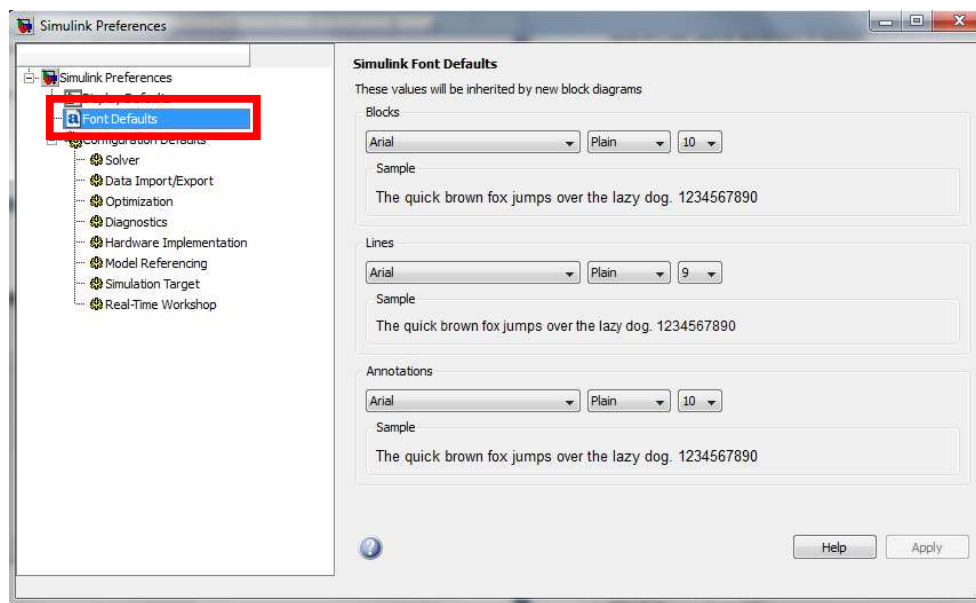


Figura3.18: Ventana de Fuentes.

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

3.2.4.2 Sistema de modelos

En SIMULINK un modelo de diagrama de bloques es una representación gráfica de un modelo matemático de un sistema dinámico, el cual viene definido por un conjunto de ecuaciones algebraicas, diferenciales, etc.

Estos modelos consisten en un conjunto de bloques y líneas, las cuales representan el tráfico de las señales, estos elementos al ser dispuestos de manera lógica y ordenada generan un diagrama de bloques el cual representará el sistema a desarrollar.

Siendo las señales la variación de estados a lo largo del tiempo, las cuales son definidas por los puntos en el tiempo donde el diagrama de bloques empieza y termina, SIMULINK permite especificar un amplio rango de atributos a estas señales, incluyendo un nombre específico, el tipo de datos, el tipo de número que usará, ya sean reales o complejos, la dimensión, etc.

Por otra parte los bloques son la representación de la relación entre señales y variables de estado definidas por ecuaciones. Estas ecuaciones reflejan cómo interactúan la entrada y salida de las señales así como el estado de las variables.

Simulink permite al usuario crear librerías de bloques personalizados para modelos específicos. Estos pueden ser creados gráficamente o por medio de programación.

Para el modo gráfico, básicamente se debe dibujar el diagrama de bloques que representará el comportamiento del bloque a crear, se le da propiedades de subsistema así como parámetros en los cuales trabajará.

Para realizarlo en modo de programación, se crea el archivo M el cual contiene las funciones del bloque, luego de este paso el archivo pasa a llamarse Archivo-S, para luego asociarlo con el bloque al que pertenecerá. En este caso también se le puede asignar parámetros extras al bloque creado.

La solución de un diagrama de bloques es obtenida mediante la evaluación de la relación de los elementos frente al factor tiempo, es decir se deben a los parámetros de inicio y finalización que el usuario disponga.

3.2.4.3 Creación de un modelo simple

En el ejemplo que se presentará a continuación se podrá observar el proceso de creación de un modelo usando los diferentes comandos y acciones con los cuales el usuario podrá generar sus propios modelos.

Las instrucciones para crear este modelo son puntuales. El modelo integra una señal sinusoidal y muestra el resultado, junto con la señal principal. El diagrama de bloques al cual se llegará con este procedimiento es el que se muestra en la figura 3.19:

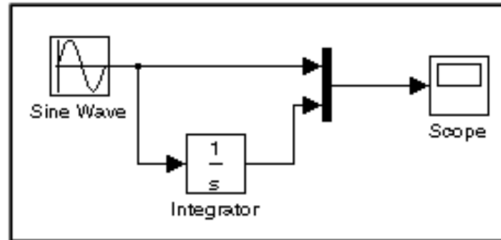


Figura 3.19: Diagrama de bloques con una señal sinusoidal de entrada

Fuente: <http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf>

Para crear el modelo, primero se debe escribir SIMULINK en la ventana de comandos de MATLAB. Si se está trabajando con Microsoft Windows, el buscador de librería SIMULINK se verá como se muestra en la figura 3.20.

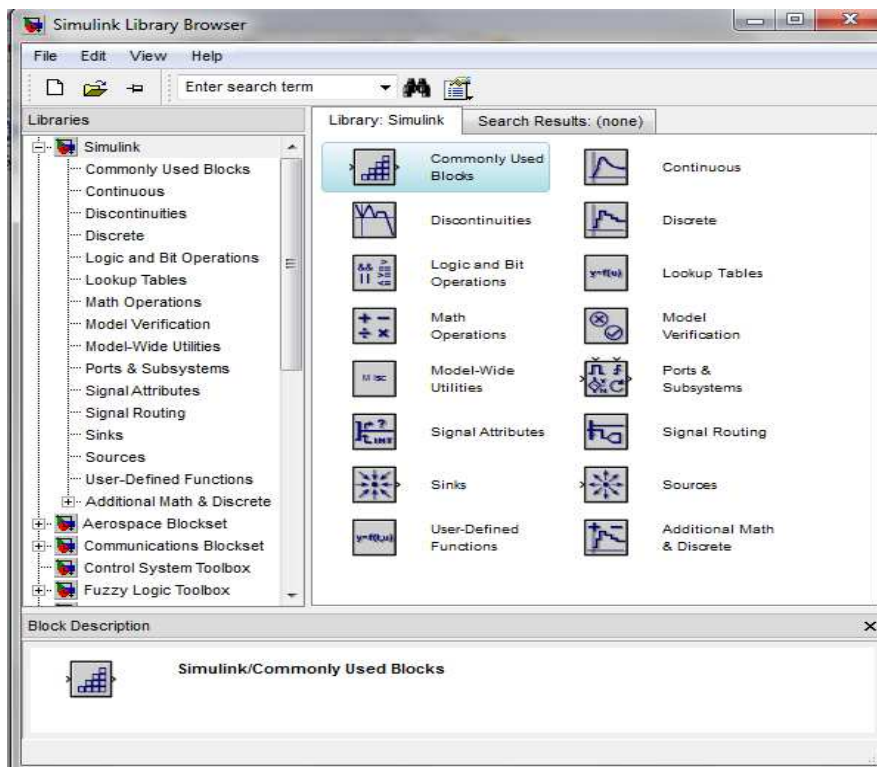


Figura 3.20: Buscador de librerías SIMULINK en Windows

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

Para crear un nuevo modelo, se selecciona *Model* en el submenú *Nuevo* de la ventana de menú *Archivo* de la librería SIMULINK. Si está trabajando en Windows, puede seleccionar directamente el botón *Nuevo Modelo* en la barra de herramientas de la librería, como se señala en la figura 3.21.

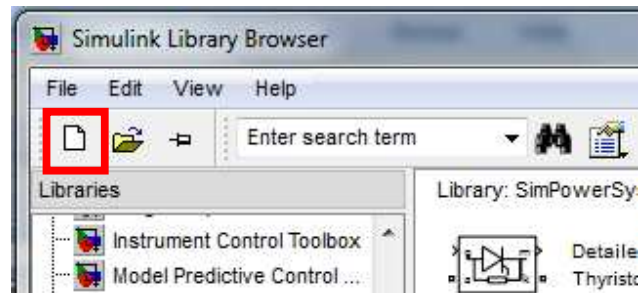


Figura 3.21: Selección de opción *Nuevo Modelo*
Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

A continuación SIMULINK abre una nueva ventana de modelo, la cual es mostrada en la figura 3.22.

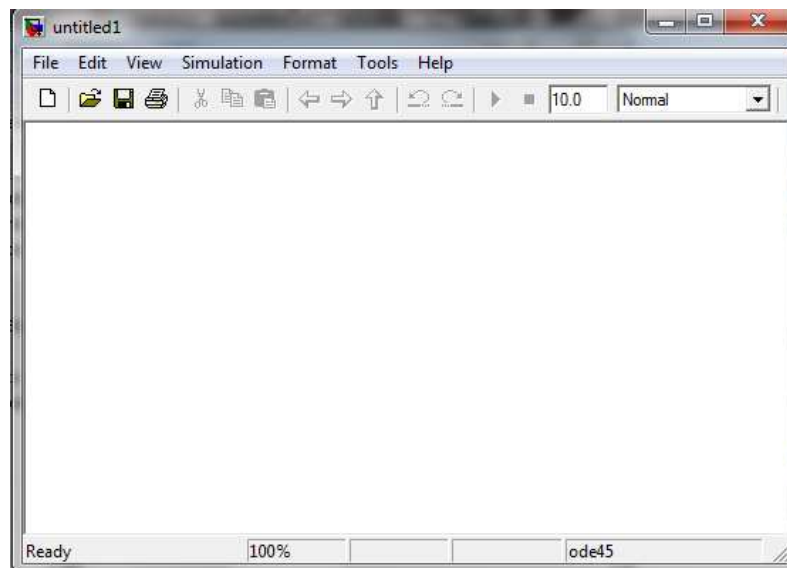


Figura 3.22: Nueva ventana de modelo
Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

Para crear un modelo en SIMULINK es necesario tener en el área de trabajo los bloques que se acoplen a la necesidad del proyecto. Así, para este modelo, se necesitará copiar los bloques en el modelo de las siguientes librerías de bloques de SIMULINK:

- Librería de fuentes (El bloque de señal sinusoidal)
- Librería *Sinks* (**bloque Scope**)
- Librería *Continuous* (**bloque Integrator**)
- Librería de *Signal Routing* (el bloque *Mux*)

Para copiar el bloque de Señal Sinusoidal del Buscador de Librerías, primero se debe expandir las opciones para poder observarlas y escoger la librería de Fuentes.

Para realizar este paso se puede hacer *click* en el nodo Fuentes para mostrar los bloques en la librería de Fuentes. Finalmente, se hace *click* en el nodo de señal sinusoidal para seleccionar su bloque.

A continuación se mostrará en la figura 3.23 la ventana final después del proceso realizado.

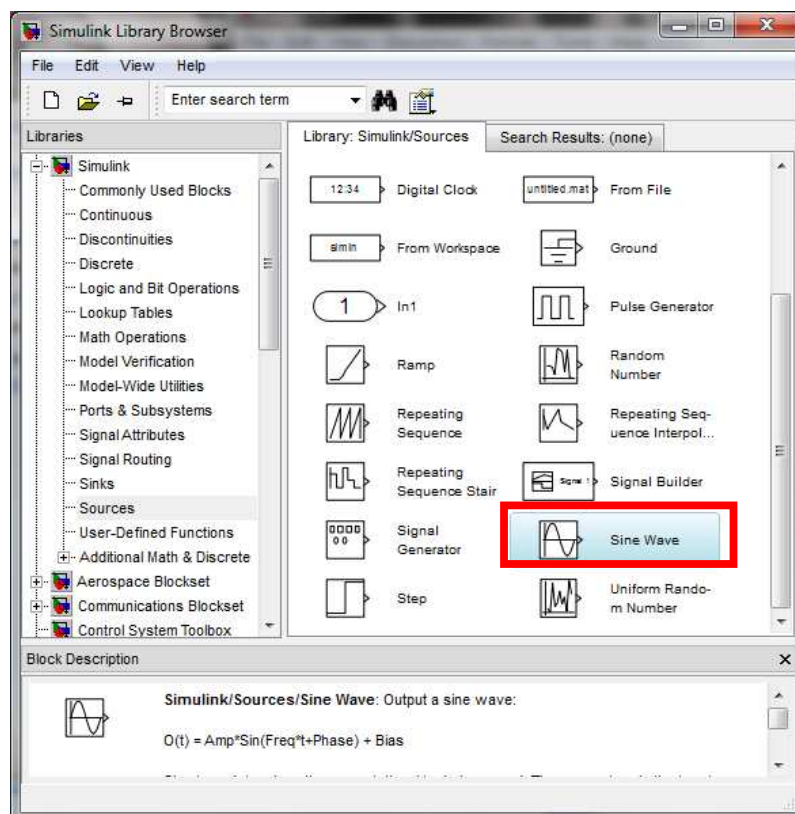


Figura 3.23: Ventana de bloques

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

Desde esta ventana se puede arrastrar el bloque necesitado y dejarlo en la ventana de trabajo del modelo. SIMULINK crea una copia del bloque al ser dejado en la ventana.

La figura 3.10 muestra la ventana con el bloque de la señal sinusoidal, el cual es llevado a una nueva ventana como se puede observar en la figura 3.24.

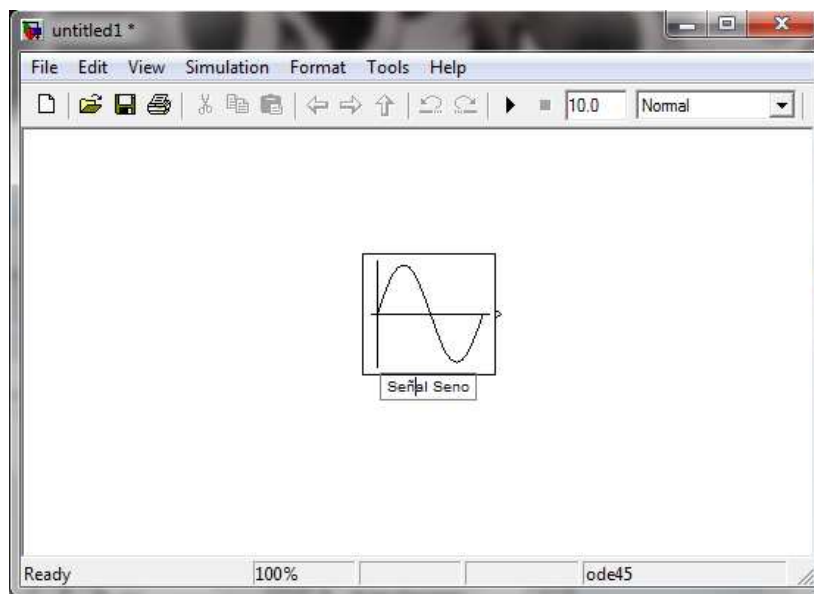


Figura 3.24: Bloque de la Señal Seno en la ventana de trabajo

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

A continuación se copia el resto de bloques desde sus librerías respectivas de manera similar hacia la ventana de modelo. El usuario podrá mover los bloques en la ventana de trabajo con solo desplazarlos dentro de la misma al lugar deseado o por medio de las teclas de desplazamiento en su teclado.

Luego de haber transportado los bloques necesarios, el modelo en la ventana de trabajo debe lucir como se muestra en la figura 3.25.

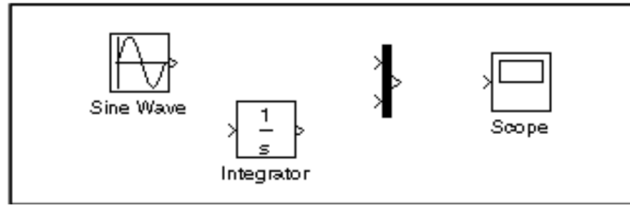


Figura 3.25: Bloques del modelo

Fuente: <http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf>

Al examinar los iconos de los bloques, se puede observar una pequeña flecha a la derecha de la Señal Seno y dos en el lado izquierdo del bloque Mux. El símbolo > cuando está apuntando hacia afuera del bloque indica que es un puerto de salida, y si el símbolo apunta hacia el bloque, significa que es un puerto de entrada. Una señal viaja desde un puerto de salida hacia un puerto de entrada de otro bloque a lo largo de una línea de conexión. Cuando los bloques están correctamente conectados, el símbolo del puerto desaparece. Los significados de los símbolos correspondientes a los puertos de entrada o salida se muestran en la figura 3.26.

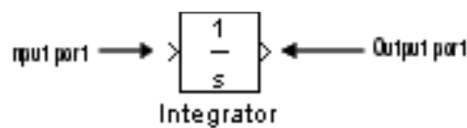


Figura 3.26: Entrada/salida de bloque

Fuente: <http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf>

El paso siguiente corresponde a la conexión de los bloques. Se empieza conectando el bloque de Señal Seno al puerto de entrada del bloque Mux. A continuación se posiciona el puntero sobre el puerto de salida en el lado derecho del bloque de Señal Seno. Puede observarse como la forma del cursor cambia. La conexión de bloques se muestra en la figura 3.27.

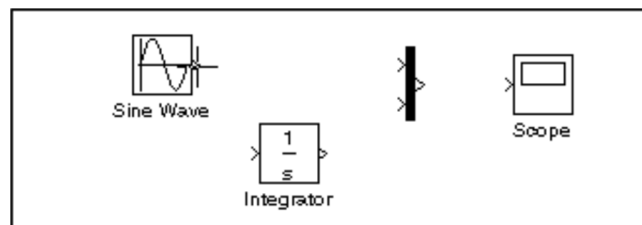


Figura 3.27: Inicio de la conexión de bloques

Fuente: <http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf>

Al mantener presionado el botón del mouse y moviendo el cursor hasta el puerto de entrada del bloque Mux, se nota que la línea es punteada mientras el mouse se encuentra presionado y que la forma del cursor cambia a una cruz de doble línea mientras se acerca al bloque Mux, como puede observarse en la figura 3.28.

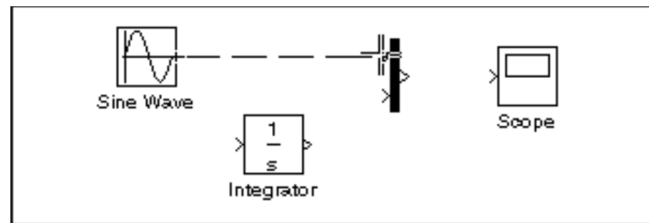


Figura 3.28: Cambios del cursor en la conexión de bloques

Fuente: <http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf>

Luego, se suelta el botón del mouse, en este momento los bloques pasan a estar conectados. También puede realizar la conexión soltando el botón del mouse mientras el cursor se encuentra dentro del icono. Si lo hace de esta manera, la línea se conectará al puerto de entrada más cercano a la posición del cursor. La conexión terminada puede verse en la figura 3.29.

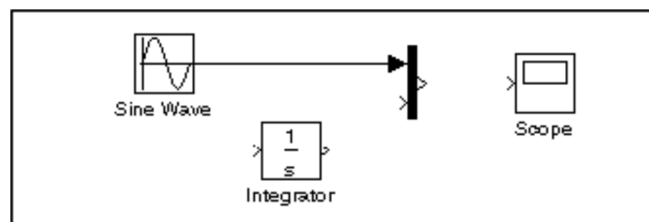


Figura 3.29: Conexión de bloques terminada

Fuente: <http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf>

Puede ahora notarse que en el modelo principal, la mayoría de las líneas conectan puertos de salida de bloques a puertos de entrada de otros bloques. Aunque una línea conecta a otra línea al puerto de entrada de otro bloque. Esta línea, llamada Línea de Rama, conecta la salida de la Señal Seno al bloque *Integrator*, y lleva la misma señal que pasa del bloque Señal Seno hacia el bloque Mux.

Dibujar este tipo de líneas es un poco diferente a las líneas ya realizadas. Para realizar una conexión en una línea existente, se realizan los siguientes pasos:

1. Primero se posiciona el cursor en la línea entre los bloques de Señal Seno y Mux, como se muestra en la figura 3.30.

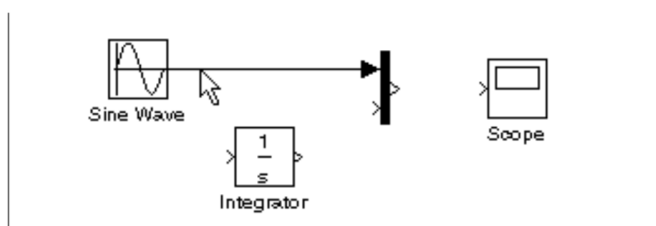


Figura 3.30: Inicio de la conexión de la Línea de Rama

Fuente: <http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf>

2. Manteniendo presionada la tecla Ctrl (o haciendo click en el botón derecho del mouse), se presiona el botón del mouse, para luego arrastrar el puntero hacia el Puerto de Entrada del bloque *Integrator* o dentro del mismo bloque, el resultado de esta operación puede observarse en la figura 3.31.

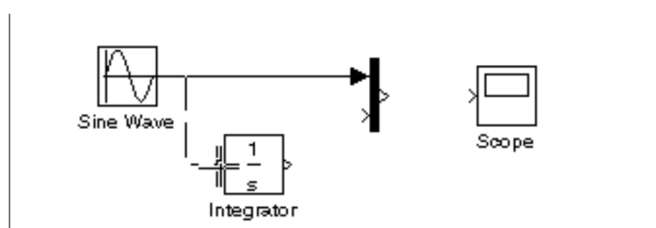


Figura 3.31: Arrastre del cursor para la conexión de la Línea de Rama.

Fuente: <http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf>

3. Al Soltar el botón del mouse SIMULINK dibuja una línea entre el punto de inicio de la conexión y el puerto de entrada del bloque *Integrator*, como se indica en la figura 3.32.

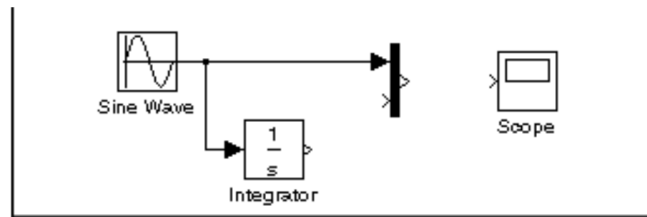


Figura 3.32: Conexión de Línea de Rama

Fuente: <http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf>

Al terminar las conexiones de bloque, el modelo debe lucir de la manera como se muestra en la figura 3.33.

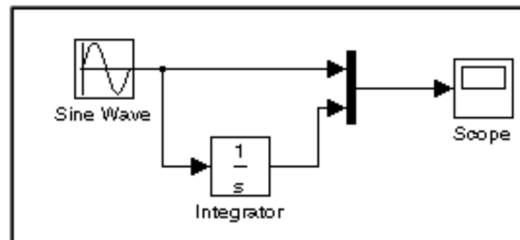


Figura 3.33: Modelo de bloques

Fuente: <http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf>

En el siguiente paso, se abre el bloque *Scope* para ver la salida de la simulación. Manteniendo la ventana de *Scope* abierta, se configura SIMULINK (Ctrl + E) para que la simulación dure 10 segundos. Primero, se completa los parámetros de simulación, escogiendo Parámetros de Simulación del menú de Simulación.

En la ventana de diálogo que aparece puede verse que el tiempo de finalización debe estar en 10.0. La configuración de parámetros se muestra en la figura 3.34.

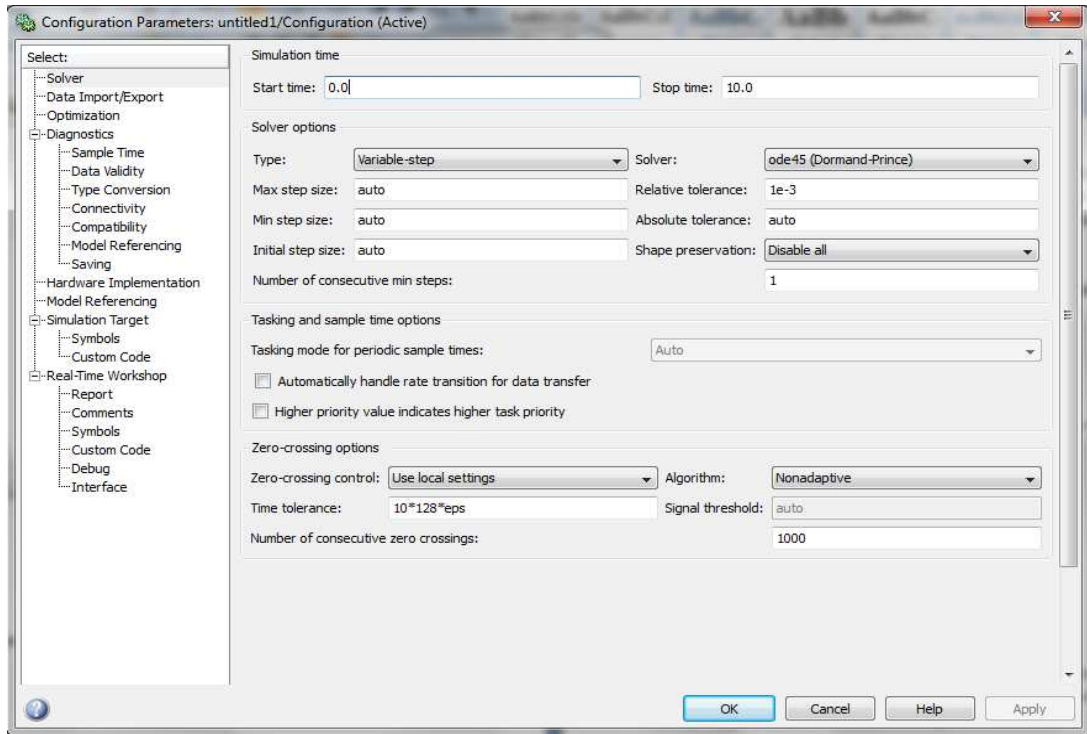


Figura 3.34: Configuración de parámetros

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

A continuación se cierra la ventana de Parámetros de Simulación, haciendo *click* en el botón Ok. Luego se escoge *Start* en el menú de Simulación y se podrá observar las trazas de las entradas del bloque *Scope*, como puede observarse en la figura 3.35.

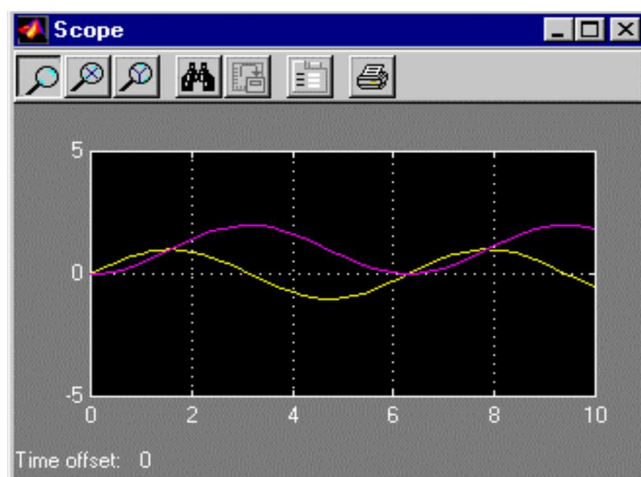


Figura 3.35: Visualización de Señales

Fuente: <http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf>

La simulación se detiene cuando llega al tiempo especificado en la ventana de Parámetros de Simulación o cuando se escoge *Stop* del menú de Simulación.

Para guardar este modelo, se escoge *Guardar* del menú *Archivo* y se ingresa el nombre del fichero y su ubicación.

3.2.4.4 Recomendaciones

Algunas recomendaciones importantes que es necesario considerar para la observación de las señales resultantes del diagrama para la simulación del circuito se detallan a continuación:

- Para observar diferentes señales, el usuario deberá agregar un bloque *Scope* por cada señal entrante.
- Si solo desea usar un bloque *Scope*, puede enviar las señales al área de trabajo de MATLAB, donde las puede mostrar normalmente. Esto lo puede realizar usando el bloque '*Sinks – To Workspace*'.
- Si desea observar algunas señales en un solo *Scope*, debe coectar las mismas verticalmente en el tramo previo a la entrada del puerto del *Scope*.

3.2.4.5 Ayuda de SIMULINK

Seleccione “*Ayuda SIMULINK*” del menú de *Ayuda* en el *Buscador de librerías*. En esta opción se puede encontrar tutoriales, *demos* (explicaciones paso a paso para realizar modelos previamente diseñados), información en bloques disponibles y mucho más. Esta ventana puede observarse en la figura 3.36.

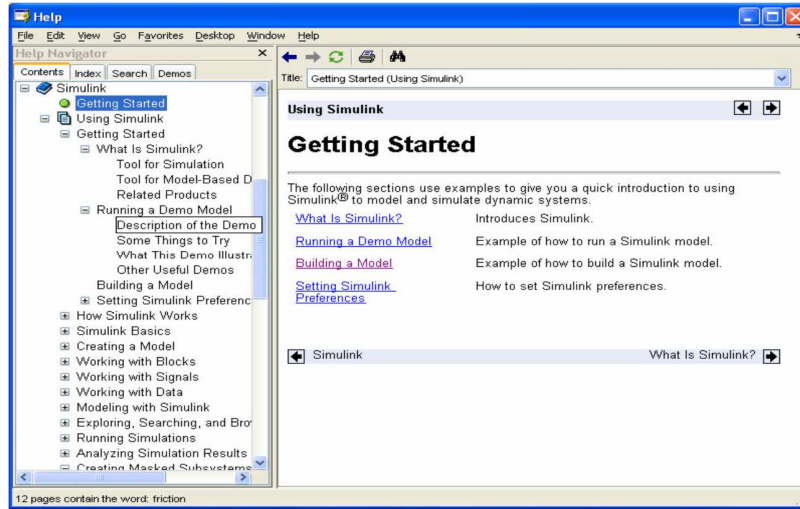


Figura 3.36: Ventana Ayuda SIMULINK

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

3.2.4.6 Cambio de parámetros

Para realizar cualquier cambio de parámetro, se debe hacer doble *click* en el bloque que se desee hacer el cambio lo cual mostrará otra ventana en la cual se podrá modificar los parámetros necesarios para el ejercicio. Esta ventana puede verse en la figura 3.37

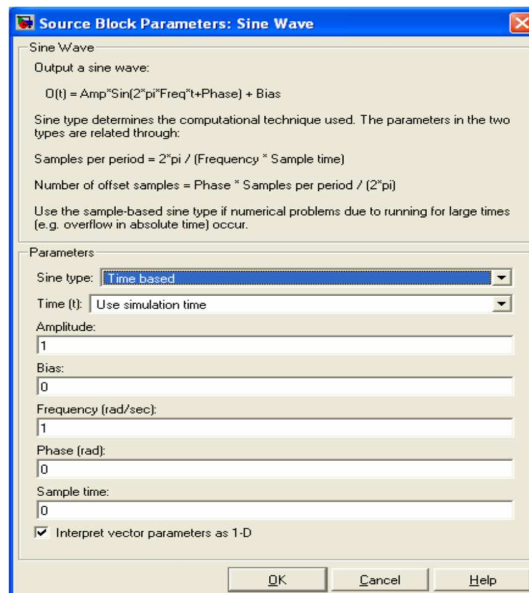


Figura 3.37: Ventana de Parámetros de Bloque

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

3.2.4.7 Comentarios

El usuario puede agregar comentarios en cualquier sitio del diagrama de bloques al realizar doble *click* e ingresando el texto deseado. También puede cambiar los comentarios por default de cada bloque, así mismo haciendo doble *click* y editando el texto. Un ejemplo se muestra en la figura 3.38.

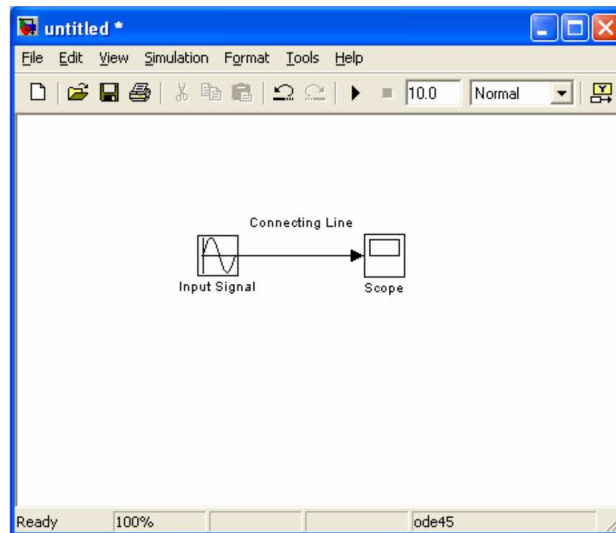


Figura 3.38: Cambio de comentarios en bloques.

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

3.2.4.8 Subsistemas

Un diagrama de bloques SIMULINK puede consistir de un conjunto de capas, siendo cada una de estas definida por un subsistema. Un subsistema es parte de un diagrama de bloques y de forma ideal no genera ningún impacto en el desempeño del mismo. Estos son de gran ayuda en los aspectos de la organización del diagrama.

Existen dos clases de subsistemas, los virtuales y los no-virtuales, siendo la diferencia más marcada entre los dos el hecho de que los subsistemas no-virtuales permiten el control de los contenidos del subsistema a la hora de evaluar los mismos.

SIMULINK permite que el usuario disponga un grupo de bloques juntos creando un subsistema, para lograr esto se debe seleccionar los bloques a unir para luego hacer *click* derecho, y escoger la opción *Crear Subсистema*. Esto hará que se cree un solo bloque. Si luego de esto se desea visualizar lo que forma parte del subsistema solo es necesario dar doble *click* sobre el mismo y podrá ver lo que se encuentra debajo de la “*máscara*”. A estos subsistemas se les llama “*Masked*”. La visualización de un subsistema es presentada en la figura 3.39.

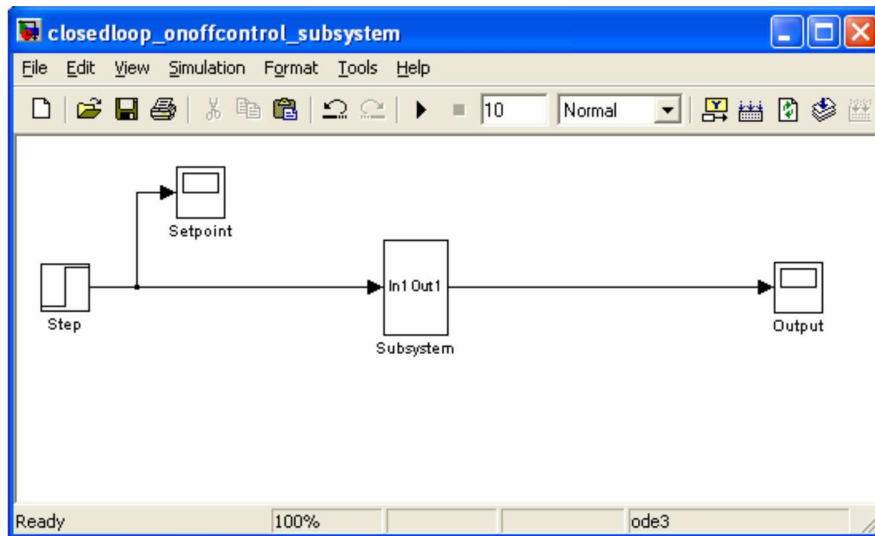


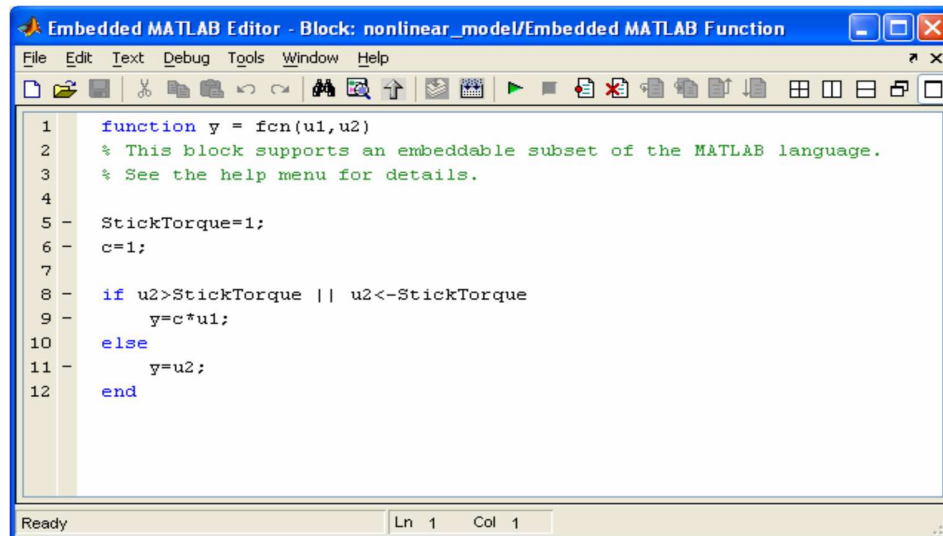
Figura 3.39: Visualización de un subsistema

Fuente: <http://www.mathworks.com/>

3.2.4.9 Funciones derivadas de MATLAB

Debido a que SIMULINK se deriva de MATLAB, es posible importar funciones realizadas en éste y asignársela a un bloque o subsistema determinado. Esto se lo puede realizar insertando los archivos M definidos por el usuario mediante el bloque “*Embedded MATLAB Function*” el cual se encuentra en la librería de *Funciones definidas por el Usuario*.

Un ejemplo de la programación en MATLAB se presenta en la figura 3.40.



```
1 function y = fcn(u1,u2)
2 % This block supports an embeddable subset of the MATLAB language.
3 % See the help menu for details.
4
5 - StickTorque=1;
6 - c=1;
7
8 - if u2>StickTorque || u2<=-StickTorque
9 -     y=c*u1;
10 else
11 -     y=u2;
12 end
```

Figura 3.40: Programación en MATLAB

Fuente: <http://www.mathworks.com/>

En la figura 3.41 se presenta un diagrama en SIMULINK con una función MATLAB asignada a un bloque.

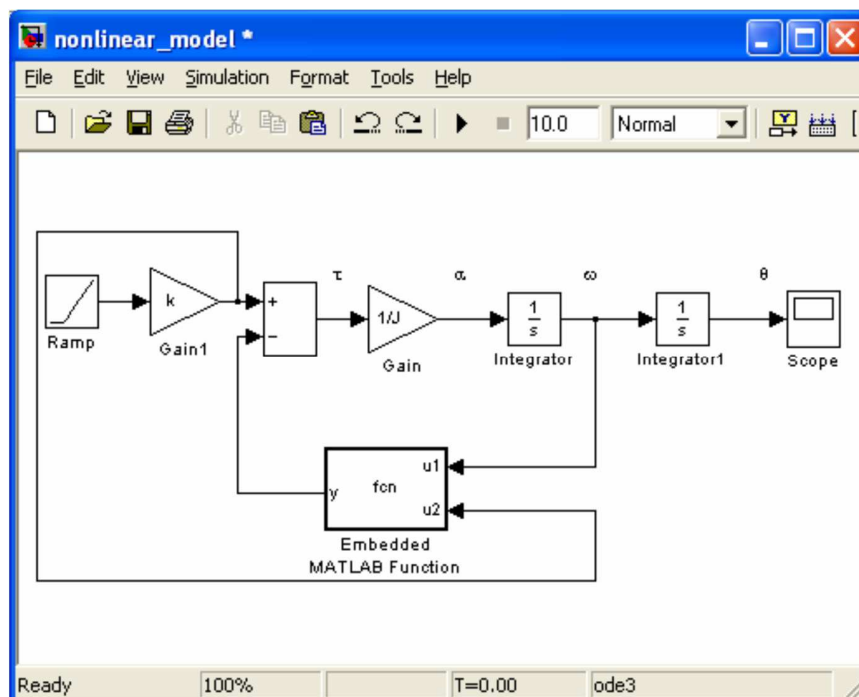


Figura 3.41: Diagrama en SIMULINK con una función MATLAB asignada a un bloque

Fuente: <http://www.mathworks.com/>

3.2.4.10 Métodos de bloque

Como se había indicado previamente, los bloques representan múltiples ecuaciones, las cuales dentro de SIMULINK son conocidas como *Métodos*. Los métodos son la base a la hora de analizar los pasos a seguir en el proceso de ejecución del diagrama de bloque. La evaluación de los métodos se da dentro del tiempo de simulación establecido, donde cada ciclo representa la evaluación del bloque en un determinado punto en el tiempo.

3.2.4.11 Tipos de métodos

SIMULINK asigna nombres a los tipos de funciones que se desarrollan en los métodos de bloques. Una de las más importantes es la Función de transferencia; se le da este nombre a la función matemática hipérbola equilátera. Para configurar esta función en la que convergen polinomios en el numerador y denominador, se debe realizar doble *click* en el bloque de función de transferencia y procediendo a establecer los parámetros necesarios. El diagrama de bloques correspondiente se muestra en la figura 3.42.

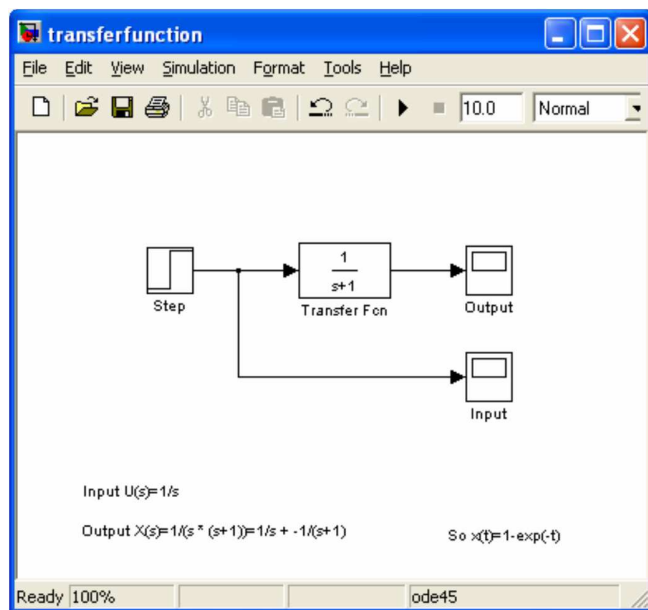


Figura 3.42: Diagrama del bloque *Transferencia*.

Fuente: <http://www.mathworks.com/>

En la figura 3.43 se muestra la ventana para el establecimiento de parámetros del Bloque *Transferencia*.

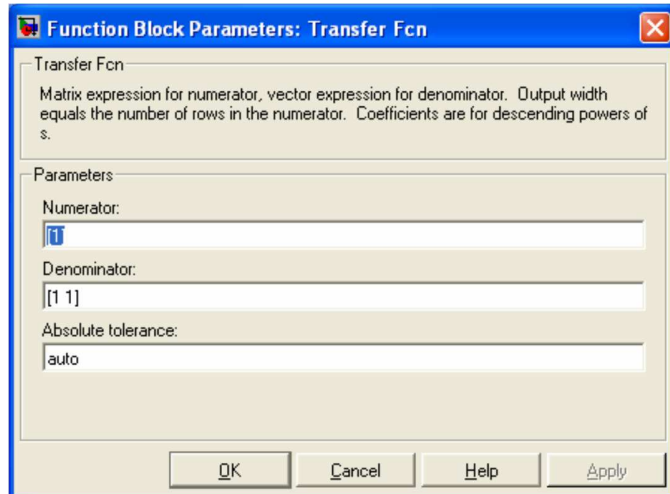


Figura 3.43: Ventana de parámetros

Fuente: <http://www.mathworks.com/>

La visualización de la entrada y salida se muestra en la figura 3.44.

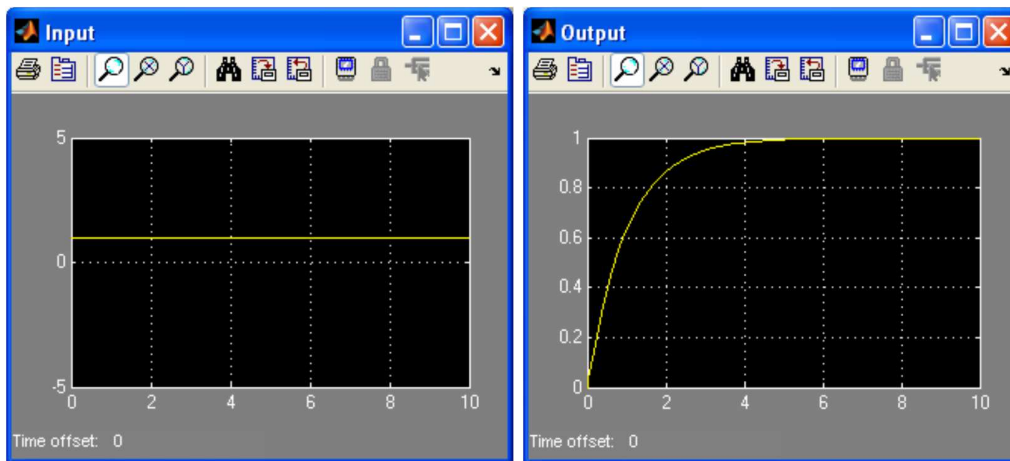


Figura 3.44: Simulación de función

Fuente: <http://www.mathworks.com/>

3.2.4.12 Integrales y Derivadas

Esta función se refiere al cambio de estado continuo a un estado en un determinado tiempo, dependiendo de la entrada del bloque y los valores de los estados en el periodo anterior.

Los bloques de estas funciones se los puede encontrar en la librería “*continuous*”.

Para el caso de las integrales, se puede establecer la condición inicial y el límite de la salida para que no supere o trabaje por debajo de valores determinados, en las propiedades de ese bloque. Su diagrama de bloques se presenta en la figura 3.45.

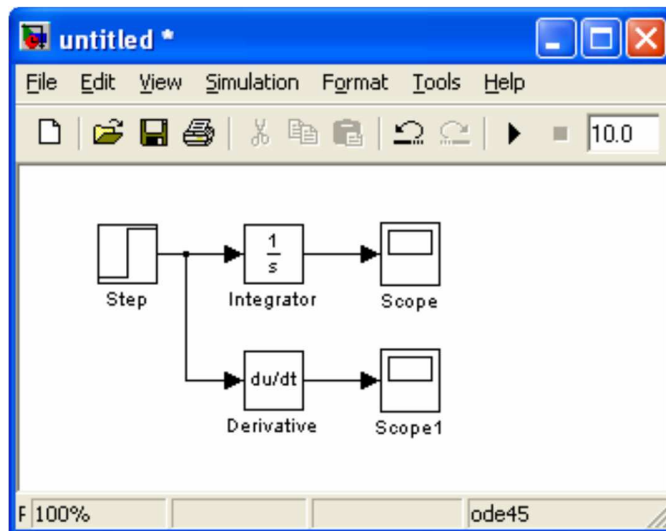


Figura 3.45: Diagrama de bloques de funciones Integrales y Derivadas

Fuente: <http://www.mathworks.com/>

La visualización de las señales de salida producidas por estas funciones se muestra en la figura 3.46.

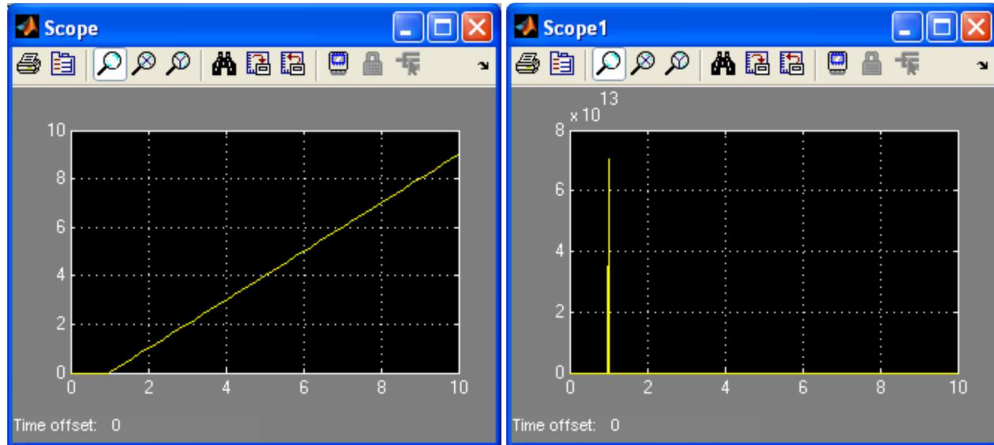


Figura 3.46: Señales de salida
Fuente: <http://www.mathworks.com/>

3.2.4.13 Usando Demos

SIMULINK posee una amplia variedad de archivos los cuales albergan la explicación dinámica de los pasos necesarios para desarrollar modelos construidos previamente por el fabricante, los mismos que ilustran conceptos muy útiles para el modelado de proyectos. Para el acceso a estos *demos*, el usuario debe ubicarse en la ventana de comandos de MATLAB, luego hacer *click* en el botón de “*Start*” en la esquina izquierda de la misma y acción seguida aparecerá el menú de esta opción, como se observa en la figura 3.47.

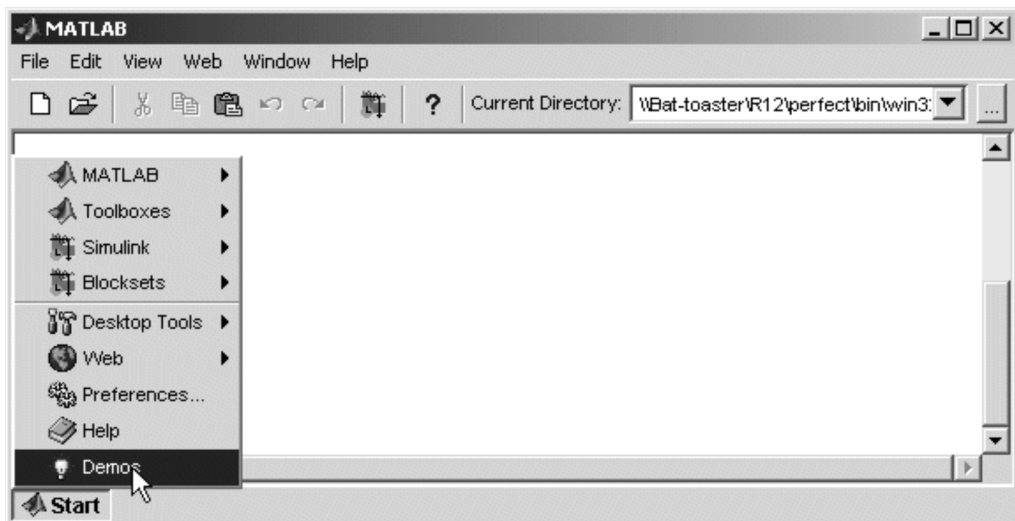


Figura 3.47: Ingreso a Demos
Fuente: <http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf>

Luego se debe escoger la opción *Demos* de este menú, con lo que el buscador de *Ayuda de MATLAB* aparecerá mostrando así el panel de *Demos* seleccionado previamente, el cual puede verse en la figura 3.48.

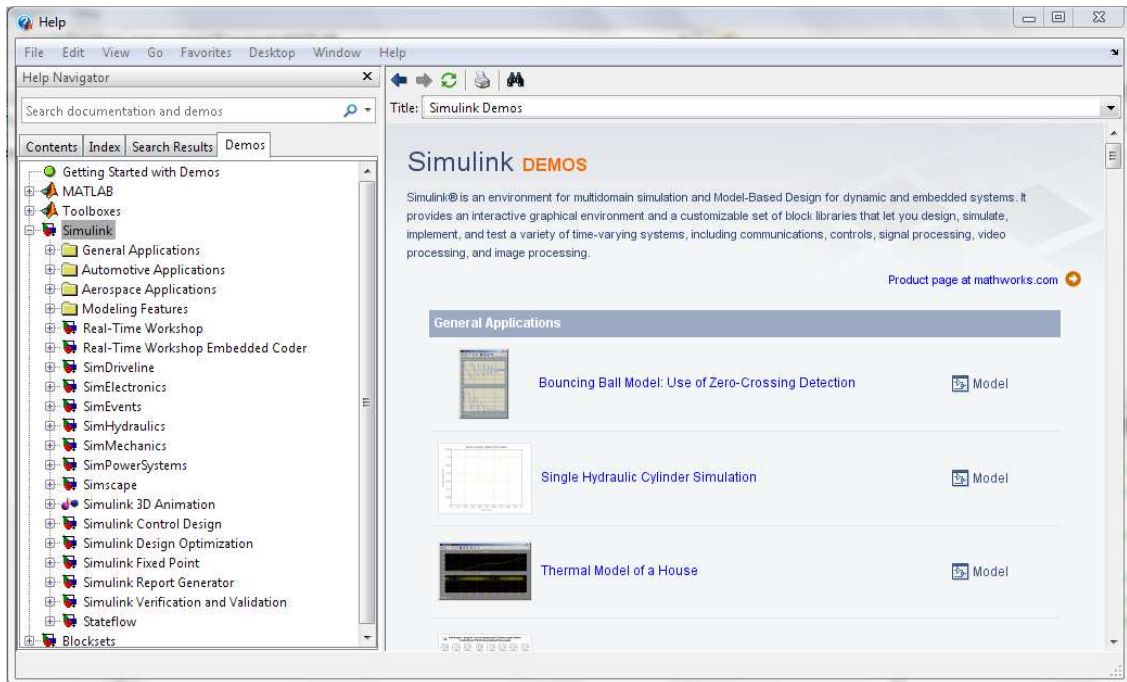


Figura 3.48: Panel de Demos

Fuente: Print Screen de Matlab Versión 7.8.0.347 (R2009a)

En esta ventana se debe hacer *click* en la opción *SIMULINK* del panel de *Demos*. Al realizarlo la opción se expandirá mostrando los grupos de *demos*. Usando el buscador para navegar en *demos* se pueden encontrar aquellos que puedan ser de utilidad para un proyecto determinado. Cada opción posee una breve explicación así como un *link* directo hacia el *demo*.

En el siguiente capítulo se aplicarán los conocimientos adquiridos en el manejo del programa de simulación *SIMULINK* de *MATLAB* para desarrollar prácticas cuya utilización se recomienda en las clases de la asignatura de *Electrónica I* como un complemento de las mismas, como resultado de este proceso se presentará un nuevo programa de la materia con la inclusión de estas prácticas.

CAPITULO 4: IMPLEMENTACIÓN DE PRÁCTICAS CON EL PROGRAMA DE SIMULACIÓN SIMULINK DE MATLAB

En este capítulo se aplicarán los conocimientos adquiridos acerca del programa de simulación SIMULINK de MATLAB para la implementación de prácticas cuya utilización se recomienda a los docentes de la asignatura Electrónica I. Para la consecución del objetivo principal de este trabajo de investigación, en la primera parte del capítulo se presenta el programa actual de la asignatura y posteriormente el nuevo programa con las prácticas recomendadas.

4.1 PROGRAMA ACTUAL DE LA ASIGNATURA ELECTRÓNICA I

El programa actual de la asignatura Electrónica I es el siguiente:

Diseño de las unidades de estudio (UE)

UE I: DIODOS SEMICONDUCTORES

Contenidos específicos: Introducción. El Diodo Ideal. Materiales semiconductores. Materiales extrínsecos: tipo p y tipo n. Diodo semiconductor. Niveles de resistencia. Circuitos equivalentes para diodos. Hojas de especificaciones de diodos. Diodos Zener. Análisis mediante la recta de carga. Aproximaciones de diodos. Configuraciones de diodos en serie con entradas dc. Configuraciones en paralelo y en serie-paralelo. Entradas senoidales: rectificación de media onda. Rectificación de onda completa. Recortadores. Cambiadores de nivel. Diodos Zener. Circuitos multiplicadores de voltaje.

UE II: TRANSISTORES BIPOLARES DE UNION

Contenidos específicos: Introducción. Construcción de transistores. Operación del transistor. Configuración de base común. Acción amplificadora del transistor. Configuración de emisor común. Configuración de colector común. Límites de operación. Hoja de especificaciones de transistores. Punto de operación. Circuito de polarización fija. Circuito de polarización

estabilizado en emisor. Polarización por divisor de voltaje. Polarización de dc por retroalimentación de voltaje. Diversas configuraciones de polarización. Operaciones de diseño. Redes de conmutación con transistores.

UE III: MODELAJE DE TRANSISTORES BIPOLARES

Contenidos específicos: Amplificación en el dominio de ac. Modelaje de transistores BJT. Los parámetros importantes. El modelo de transistor r_e . El modelo híbrido equivalente. Análisis a señal pequeña del transistor bipolar. Configuración de emisor común con polarización fija. Polarización mediante divisor de voltaje. Configuración de emisor común con polarización en emisor. Configuración emisor-seguidor. Configuración de base común. Configuración con retroalimentación en colector. Configuración con retroalimentación de dc en colector. Circuito equivalente híbrido aproximado. Modelo equivalente híbrido completo.

UE IV: Transistores de efecto de campo

Contenido Específico: Construcción y características de los JFET. Características de transferencia. Hojas de especificaciones. MOSFET de tipo decremental. MOSFET de tipo incremental. Polarización del FET. Configuración de polarización fija. Configuración de autopolarización. Polarización mediante divisor de voltaje. MOSFET de tipo decremental. MOSFET de tipo incremental. Modelo de señal pequeña del FET. Configuración de polarización fija del JFET. Configuración de autopolarización del JFET. Configuración de divisor de voltaje del JFET. Configuración fuente-seguidor para el JFET. Configuración de compuerta común para el JFET. MOSFET de tipo decremental. MOSFET de tipo incremental.

4.3 IMPLEMENTACION DE LAS PRÁCTICAS

Para la instalación y desarrollo de las prácticas los requerimientos mínimos son:

- Windows XP (Service Pack 1 or 2) o Windows 2000 (Service Pack 3 or 4)

- Procesador Intel Pentium III o posterior, procesador Celeron, o procesadores de la familia Intel Xeon; AMD Athlon / Duron / Opteron; o procesadores compatibles
- 256 MB RAM (se recomienda 512 MB RAM)
- Espacio mínimo en el disco de 500 MB
- 16-, 24-, o 32-bit OpenGL tarjeta aceleradora de gráficos
- Programa Matlab Versión 7.8.0.347 (R2009a).

A continuación se detallarán las prácticas que como resultado de este trabajo de investigación se recomienda se incluyan en el programa de la asignatura de Electrónica I.

4.2.1 PRÁCTICA # 1: CONFIGURACIONES DE DIODOS CON ENTRADA EN DC

Los diodos presentan como característica que dependiendo de la configuración en que se los disponga, van a permitir que la señal de entrada sea convertida en una señal directa a la salida del circuito.

En el análisis de la señal de entrada que se va a usar para esta práctica, se puede notar que en el ingreso se tendrá un voltaje directo, por lo que no habrá cambio entre la entrada y salida, siempre será el voltaje constante. Este tipo de características se pueden usar cuando se necesite la estabilización de un voltaje determinado en la entrada de un circuito.

Desarrollo de la práctica.

En esta práctica se procederá al análisis de dos circuitos los cuales presentan una entrada de voltaje directo, para lo cual se necesitará la conexión de los elementos citados en las librerías a continuación:

Librerías: se utilizarán las siguientes:

- Signal routing
 - MUX / DEMUX

- Sinks
 - Scope
- Electrical sources
 - DC Voltage source
 - Ground
 - RLC Branch
- Measurements
 - Current mesurement
 - Voltage mesurement
- Power electronic
 - Diode

La disposición de los elementos citados anteriormente debe ser la misma que se detalla en las figuras 4.1 y 4.2.

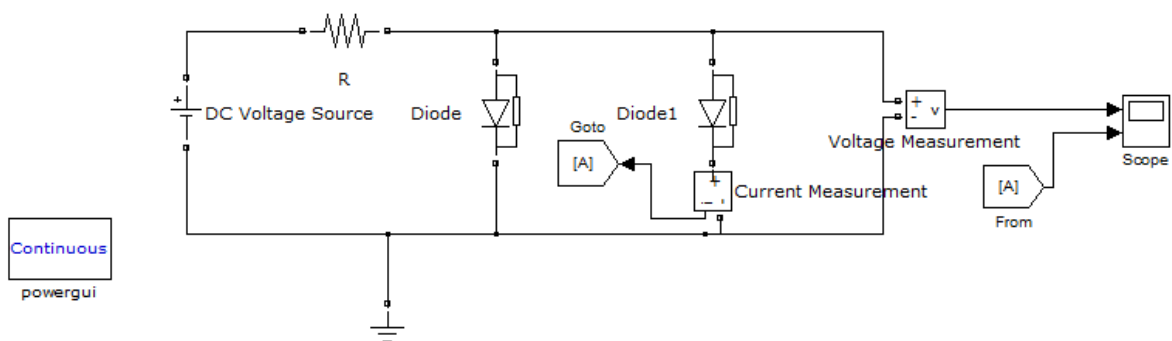


Figura 4.1: Diodos en paralelo

Fuente: Resultado de la práctica en Simulink en MATLAB

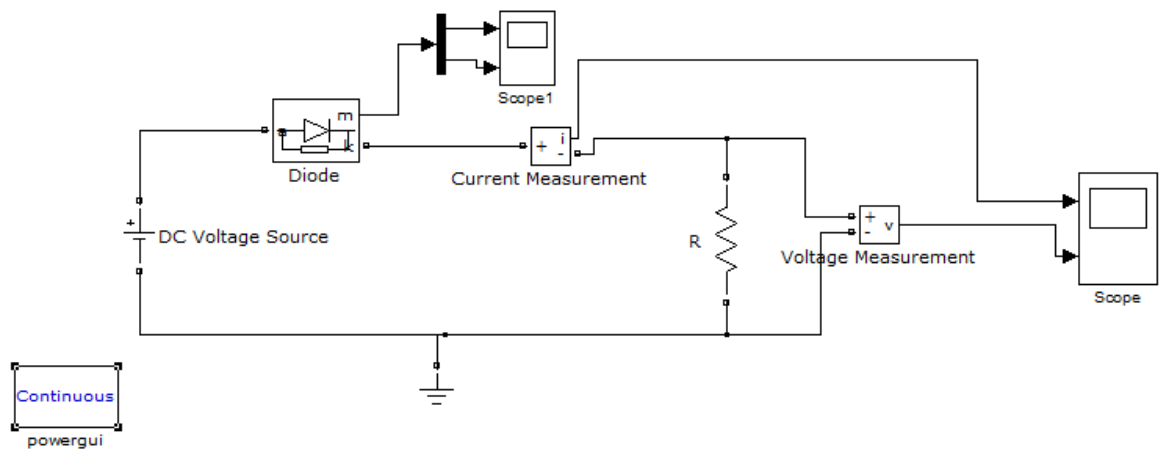


Figura 4.2: Rectificador de media onda con entrada DC
Fuente: Resultado de la práctica en Simulink en MATLAB

Parámetros

Diodos en paralelo

DC Voltage Source

Amplitud $V = 10$ voltios

Resistance

$R = 330$ ohms

Diode

Resistance = 50ohms

Inductance = 0 H

Forward voltage = 0.7 voltios

Initial current = 0 Amperios

Diode 1

Resistance = 50ohms

Inductance = 0 H

Forward voltage = 0.7 voltios

Initial current = 0 Amperios

Rectificador de media onda con entrada DC

DC Voltage Source

Amplitud $V = 100$ voltios

Resistance

$R = 8$ ohms

Diode

Resistance = 0.01 ohms

Inductance = 0 H

Forward voltage = 0.7 voltios

Initial current = 0 Amperios

Señal de salida

Diodos en paralelo

En la figura 4.3 se presenta la visualización de las señales de voltaje y corriente

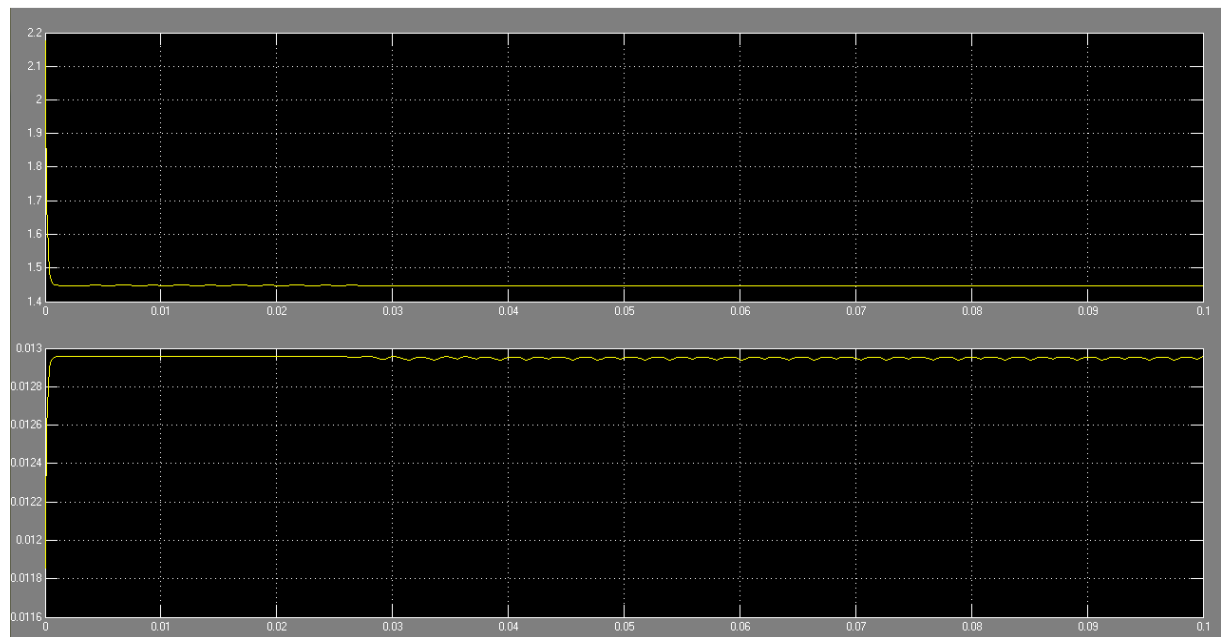


Figura 4.3: Visualización de señales de voltaje y corriente

Fuente: Resultado de la práctica en Simulink en MATLAB

En la gráfica se observa el comportamiento tanto del voltaje como de la corriente a la salida del circuito, el voltaje presenta decremento en su valor debido tanto a la caída de voltaje en la resistencia como la caída necesaria en los dos diodos para el funcionamiento, mientras que la corriente presenta un incremento llegando al punto en que se estabiliza en el momento en que los diodos permiten su paso.

Rectificador de media onda con entrada DC

La visualización de las señales de voltaje y corriente en el diodo se muestra en las figuras 4.4 y 4.5.

Scope

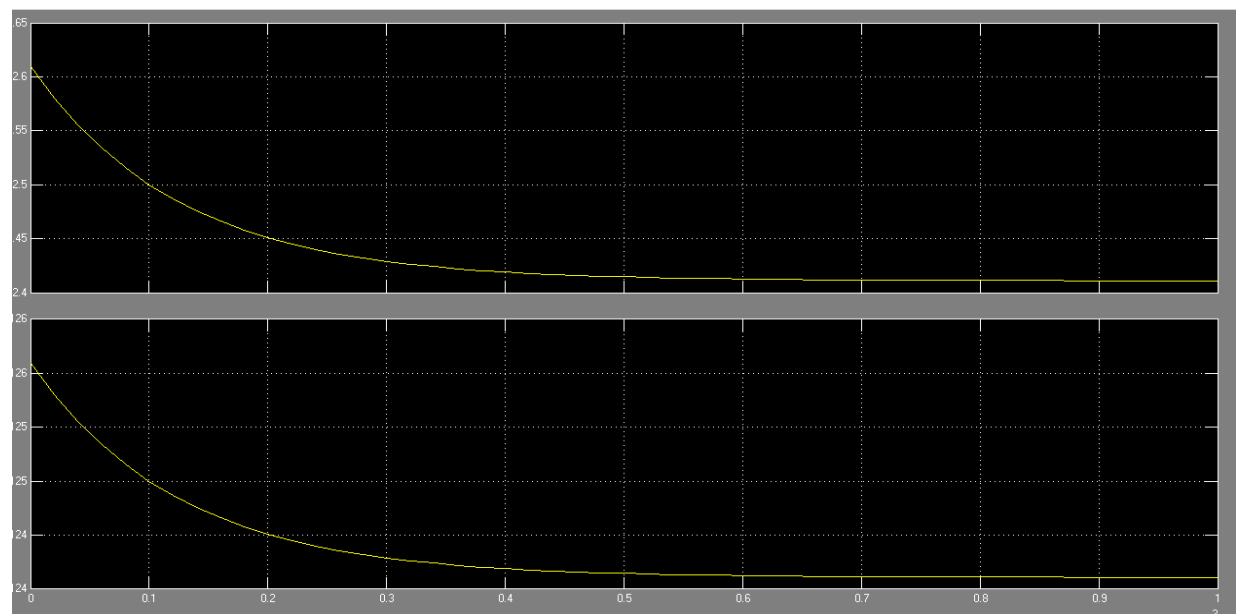


Figura 4.4: Visualización de señales de voltaje y corriente en el diodo

Fuente: Resultado de la práctica en Simulink en MATLAB

Scope 1

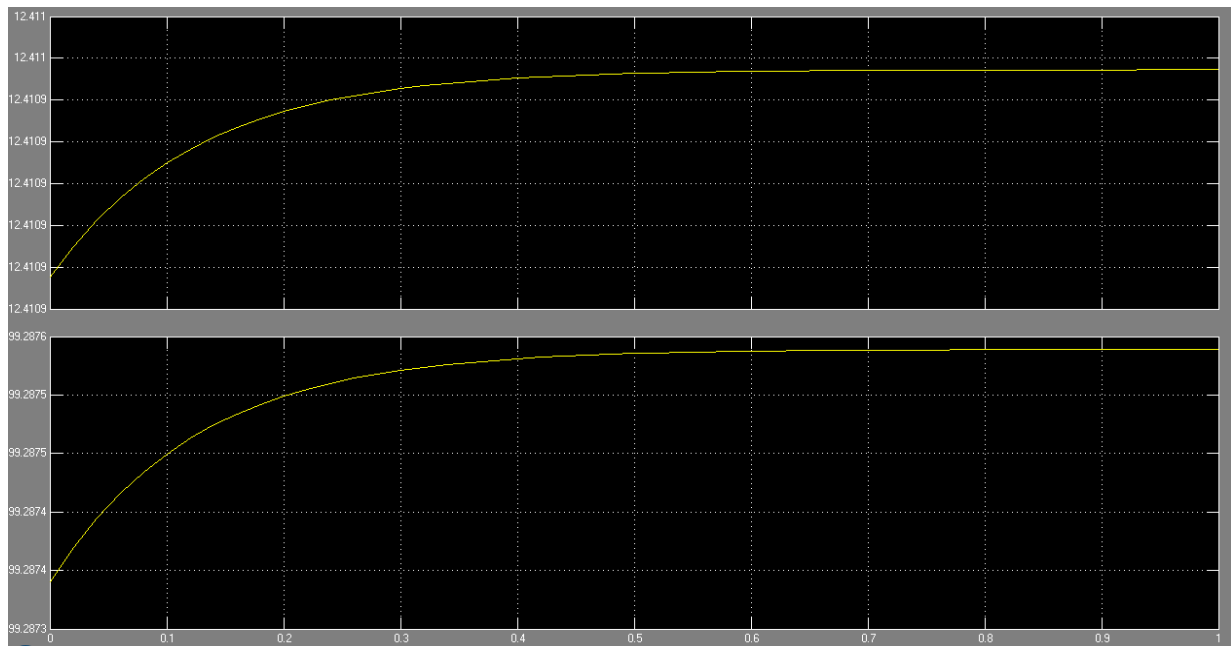


Figura 4.5: Visualización de señales de voltaje y corriente a la salida del circuito.

Fuente: Resultado de la práctica en Simulink en MATLAB

4.2.2 PRÁCTICA #2: RECTIFICADORES DE MEDIA ONDA POLARIZACIÓN DIRECTA

El rectificador de media onda es un circuito electrónico que se usa para la eliminación del área ya sea positiva o negativa de una onda determinada siendo esta de corriente alterna, este tipo de circuitos permite que la corriente alterna de ingreso luego del paso por el mismo llegue a ser corriente directa a la salida.

Como ya se había tratado los diodos permiten el paso de la corriente en una sola dirección por lo que dependiendo de la polarización que se le aplique a estos se comportaran como circuito abierto o permitiendo el paso de la corriente.

En el caso de esta práctica el diodo va a permitir el paso de la corriente, y teniendo en cuenta que el diodo necesita un voltaje característico para su polarización (germanio:0,3 y silicio:0,7) el voltaje de entrada, a la salida se verá reducido en este valor.

- $V_o = V_i - V_D \rightarrow V_o = V_i - 0,7$ (Silicio)
- $V_o = V_i - V_D \rightarrow V_o = V_i - 0,3$ (Germanio)

La característica de esta configuración es que suprime la parte negativa de la señal sinusoidal de entrada, así como la conversión de estado alterno a directo de la corriente de entrada. El circuito se presenta en la figura 4.6 y las formas de onda de entrada y salida en la figura 4.7.

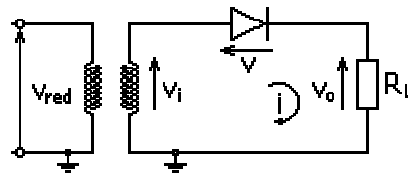


Figura4.6: Configuración de un rectificador de media onda

Fuente: http://es.wikipedia.org/wiki/Archivo:Tensi%C3%B3n_rectificada_media_onda.png

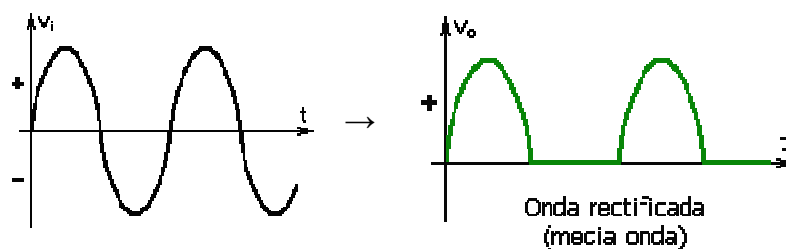


Figura 4.7: Tensión de entrada / rectificada

Fuente: http://es.wikipedia.org/wiki/Archivo:Tensi%C3%B3n_rectificada_media_onda.png

Desarrollo de la práctica.

En esta práctica se procederá al análisis de un circuito de el cual permite eliminar el semi-ciclo positivo o negativo dependiendo de la polarización del diodo, para lo que se usan las siguientes librerías:

Librerías:

- Signal routing
 - MUX / DEMUX
 - FROM / GOTO
- Sinks
 - Scope
- Electrical sources
 - AC Voltage source
 - Ground
 - RLC Branch
- Measurements
 - Current measurement
 - Voltage measurement
- Power electronic
 - Diode

La disposición de los elementos citados anteriormente debe ser la misma que se detalla en la figura 4.8 a continuación:

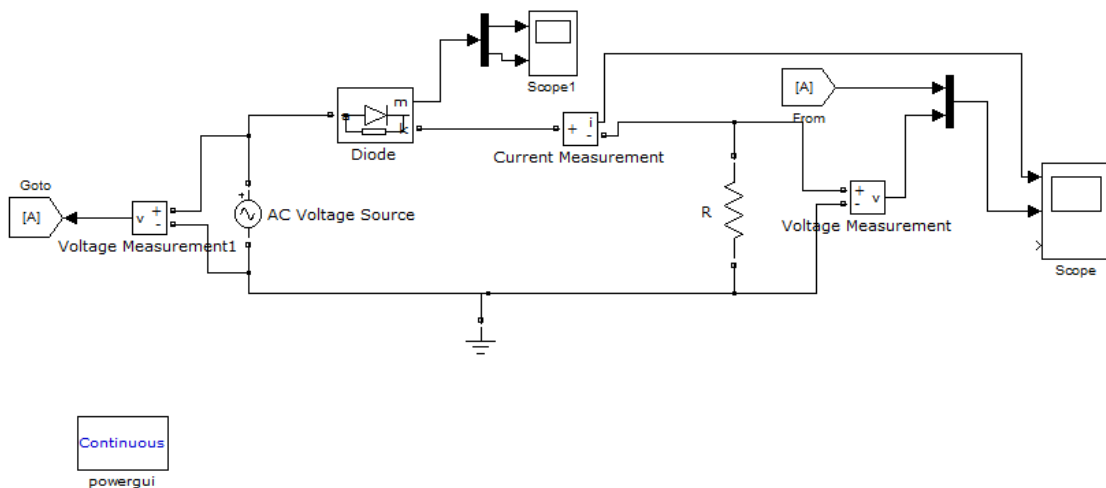


Figura 4.8: Resultado de la práctica Rectificadores de media onda polarización directa

Fuente: Resultado de la práctica en Simulink en MATLAB

Parámetros

AC Voltage Source

Peak Amplitud $V = 100$ voltios

Phase = 0 deg

Frecuency = 60 Hz

Sample time : 0

Resistance

$R = 8$ ohms

Diode

Resistance = 0.001 ohm

Inductance = 0 H

Forward voltage = 0.7 voltios

Initial current = 0 Amperios

Señal de salida

Las señales de voltaje y corriente a la salida del circuito se muestran en las figuras 4.9 y 4.10.

Scope

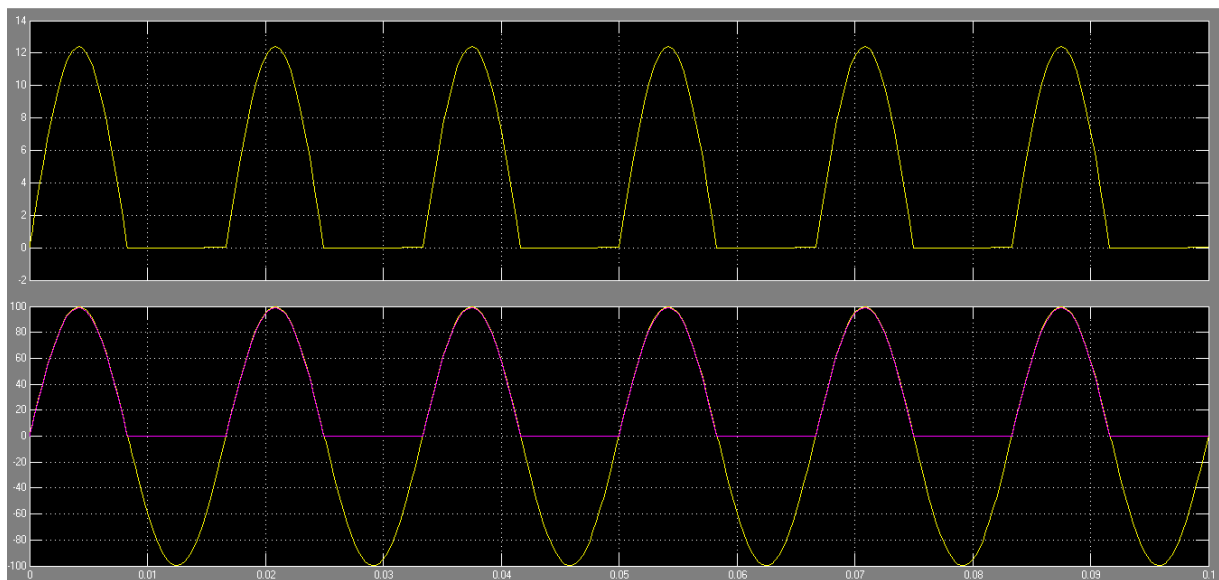


Figura 4.9: Visualización de señales de voltaje y corriente a la salida del circuito.

Fuente: Resultado de la práctica en Simulink en MATLAB

Scope 1

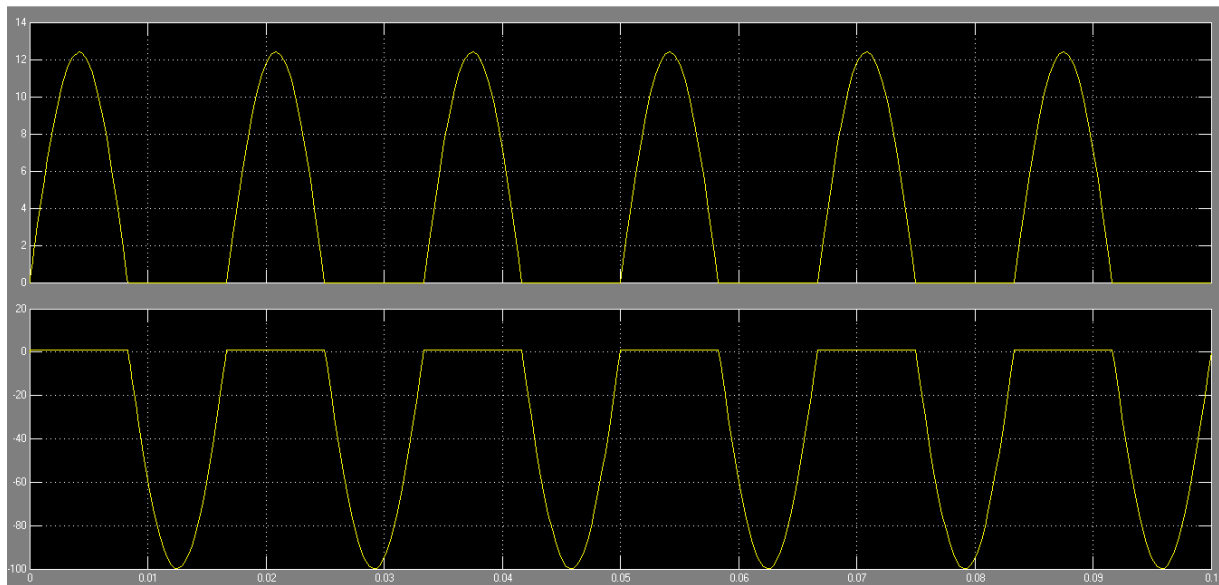


Figura 4.10: Visualización de señales de voltaje y corriente en el diodo.

Fuente: Resultado de la práctica en Simulink en MATLAB

En la gráfica del primer Scope se puede observar como se recorta el semi-ciclo negativo tanto de la corriente como del voltaje, estando resaltada con color morado la parte que se mostrará como resultante del funcionamiento del circuito. Esto se da como resultado de que el diodo solo permite el paso de la corriente cuando se encuentra polarizado directamente y esto se da en los semi-ciclos positivos del voltaje.

4.2.3 PRÁCTICA # 3: RECTIFICADORES DE MEDIA ONDA POLARIZACIÓN INVERSA

En el numeral anterior ya se indicó la función de un rectificador de media onda. En el caso de esta práctica, el diodo va a estar dispuesto de tal manera que no va a permitir el paso de la corriente, comportándose como circuito abierto, por esta razón a la hora de observar las gráficas del voltaje de salida se podrá notar que solo existirá registro de los semiciclos negativos, suprimiendo los que representan tránsito de corriente por el circuito. En la figura 4.11 se observa el circuito y en la 4.12 las ondas de entrada y salida.

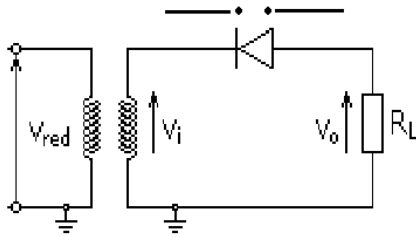


Figura4.11: Configuración de un rectificador de media onda

Fuente: http://es.wikipedia.org/wiki/Archivo:Tensi%C3%B3n_rectificada_media_onda.png

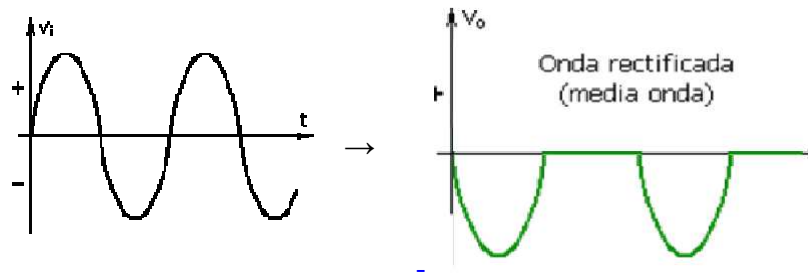


Figura 4.12: Tensión de entrada / rectificada

Fuente: http://es.wikipedia.org/wiki/Archivo:Tensi%C3%B3n_rectificada_media_onda.png

Desarrollo de la práctica.

En esta práctica se procederá al análisis de un circuito de el cual permite eliminar el semi-ciclo positivo o negativo dependiendo de la polarización del diodo, para lo que se usan las siguientes librerías:

Librerías:

- Signal routing
 - MUX / DEMUX
 - FROM / GOTO
- Sinks
 - Scope
- Electrical sources
 - AC Voltage source

- Ground
- RLC Branch
- Measurements
 - Current mesurement
 - Voltage mesurement
- Power electronic
 - Diode

La disposición de los elementos citados anteriormente debe ser la misma que se detalla en la figura 4.13 a continuación:

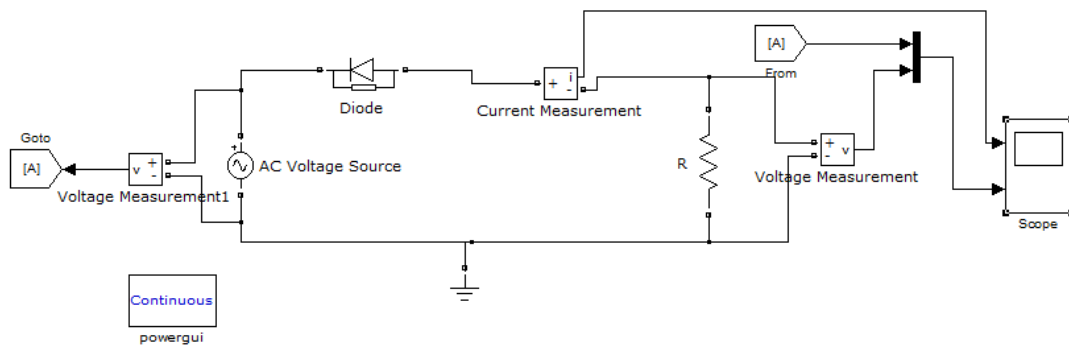


Figura 4.13: Resultado de la práctica Rectificadores de media onda polarización inversa

Fuente: Resultado de la práctica en Simulink en MATLAB

Parámetros

AC Voltage Source

Peak Amplitud V= 100 voltios

Phase = 0 deg

Frecuency = 60 Hz

Sample time: 0

Resistance

R= 8 ohms

Diode

Resistance= 0.001 ohm

Inductance = 0 H

Forward voltage = 0.7 voltios

Initial current = 0 Amperios

Señal de salida

Las señales de salida se muestran en la figura 4.14

Scope

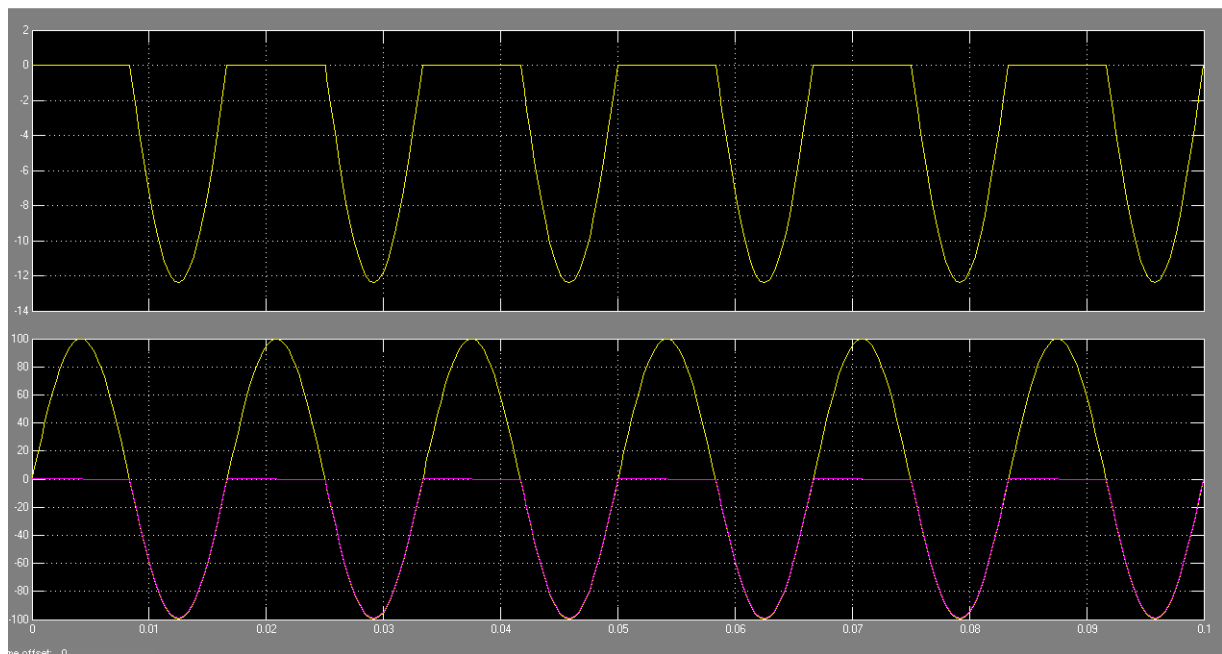


Figura 4.14: Visualización de señales de voltaje y corriente a la salida del circuito.

Fuente: Resultado de la práctica en Simulink en MATLAB

En las imágenes previas podemos visualizar la señal resultante del circuito sugerido, la misma que nos presenta la señal entrante versus la señal de salida pudiendo darnos cuenta como la misma es recortada por la función del diodo, es decir se eliminan los semi-ciclos positivos de la onda, debido a la polarización inversa que se le proporciona. Quedando así solo las partes de la onda con un valor negativo.

4.2.4 PRÁCTICA #4: RECTIFICADOR DE ONDA COMPLETA

El rectificador de onda completa es un circuito el cual convierte una onda alterna en una onda directa, rescatando los lapsos en que se presenta el semiciclo negativo de la señal de entrada; es decir, se dispone a los diodos de tal manera que siempre se vea reflejado un voltaje positivo en la salida.

Estos rectificadores presentan dos tipos de configuraciones:

- Puente de diodos
- Rectificador de derivación central

Para esta práctica se analizará el rectificador por puente de diodos, como el mostrado en la figura 4.15

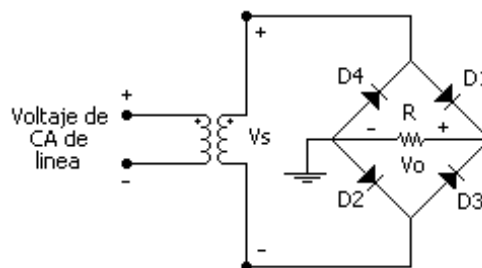


Figura 4.15: Puente de Diodos

Fuente: http://es.wikipedia.org/wiki/Archivo:Tensi%C3%B3n_rectificada_media_onda.png

Esta configuración da paso a dos casos:

- En el semiciclo positivo: D1 y D2 conducirán reflejando un voltaje en R
- En el semiciclo negativo: D3 y D4 conducirán reflejando un voltaje en R.

Con esto se puede comprobar que en la salida del circuito se tendrá un voltaje unipolar y directo.

Desarrollo de la práctica.

El desarrollo de esta práctica involucra el análisis de la disposición de cuatro diodos de tal manera que al ingresar una señal sinusoidal como es el caso del voltaje alterno, se puedan usar todos los semi-ciclos de la señal invirtiendo la parte negativa hacia el área positiva. En esta disposición se encuentran dispositivos detallados en las siguientes librerías.

Librerías:

- Signal routing
 - FROM / GOTO
- Sinks
 - Scope
- Electrical sources
 - AC Voltage source
 - Ground
 - RLC Branch
- Measurements
 - Current measurement
 - Voltage measurement
- Power electronic
 - Diode

En la figura 4.16 se muestra el circuito rectificador de onda completa.

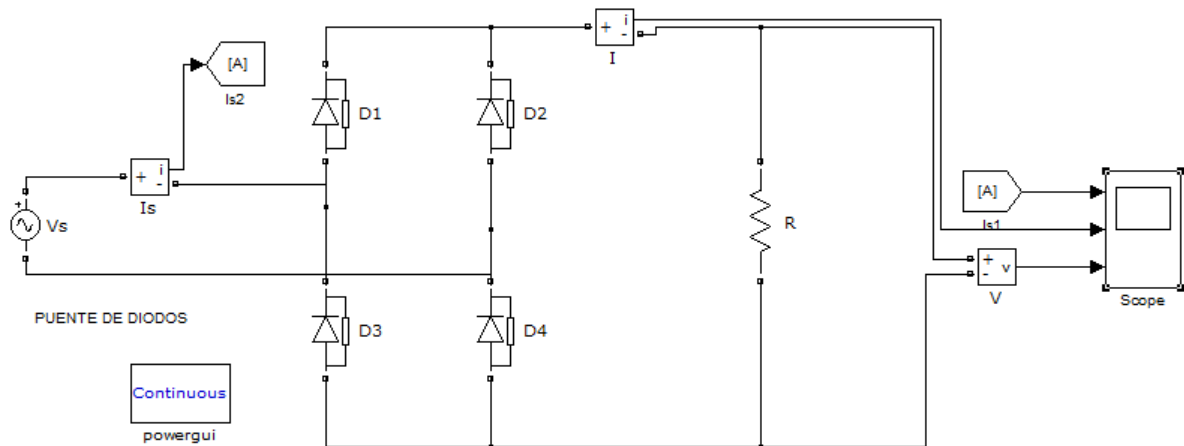


Figura 4.16: Resultado de la práctica Rectificador de onda completa

Fuente: Resultado de la práctica en Simulink en MATLAB

Parámetros

AC Voltage Source

Peak Amplitud $V = 100$ voltios

Phase = 0 deg

Frecuency = 60 Hz

Sample time : 0

Resistance

$R = 25$ ohms

Diode 1

Resistance = 0.001 ohm

Inductance = 0 H

Forward voltage = 0.7 voltios

Initial current = 0 Amperios

Diode 2

Resistance = 0.001 ohm

Inductance = 0 H

Forward voltage = 0.7 voltios

Initial current = 0 Amperios

Diode 3

Resistance= 0.001 ohm

Inductance = 0 H

Forward voltage = 0.7 voltios

Initial current = 0 Amperios

Diode 4

Resistance= 0.001 ohm

Inductance = 0 H

Forward voltage = 0.7 voltios

Initial current = 0 Amperios

Señal de salida

La visualización de las señales de voltaje y corriente a la salida del circuito se muestran en la figura 4.17.

Scope

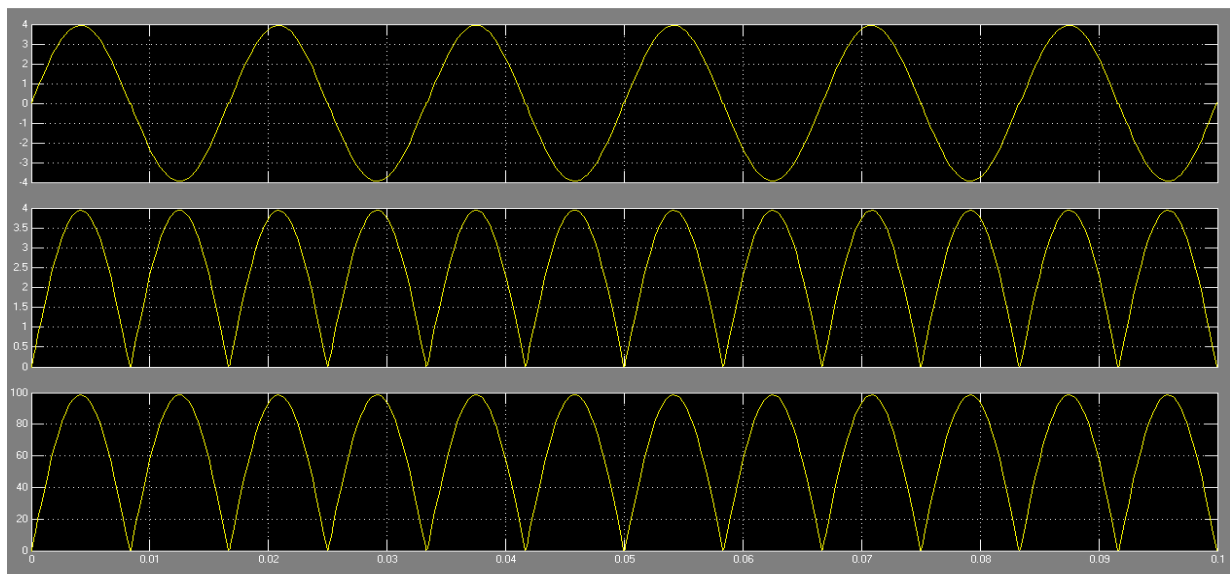


Figura 4.17: Visualización de señales de voltaje y corriente a la salida del circuito.

Fuente: Resultado de la práctica en Simulink en MATLAB

En la gráfica se puede observar, específicamente en primer lugar la entrada de la señal correspondiente al voltaje alterno, luego como al pasar por el circuito ésta se rectifica llevando los semi-ciclos negativos a ser parte de los semi-ciclos positivos.

4.2.5 PRACTICA #5 CIRCUITOS CAMBIADORES DE NIVEL

Los circuitos cambiadores de nivel se caracterizan por llevar a la señal a un voltaje determinado, para lograr esto se valen del uso de un capacitor en la composición del circuito.

El circuito respectivo se muestra en la figura 4.18.

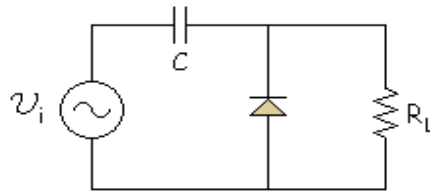


Figura 4.18: Circuito cambiador de nivel

Fuente: http://www.sc.ehu.es/sbweb/electronica/elec_basica/tema4/Paginas/Pagina19.htm

Este capacitor funciona de la siguiente manera:

- En el semiciclo negativo: el diodo se comportará como cortocircuito permitiendo el paso de la corriente, con lo el capacitor se procederá a cargar
- En el semiciclo positivo: el diodo trabajará como circuito abierto, por lo que procederá a descargarse el capacitor, dándole de esta manera el voltaje con que se carga a la salida, por lo que la señal de entrada se verá incrementada en el valor que provenga del capacitor dando como resultado la señal de salida.

Desarrollo de la práctica

Al desarrollar esta práctica se puede observar el comportamiento de la señal de voltaje alterno, el cual debido a la acción del circuito tiene un desplazamiento que es generado por la carga del capacitor, el movimiento de la gráfica dependerá de la forma en la que se polarice el diodo.

Para la implementación de esta práctica se usarán los siguientes dispositivos

Librerías:

- Sinks
 - Scope
- Electrical sources
 - AC Voltage source
 - Ground
 - RLC Branch
- Measurements
 - Voltage measurement
- Power electronic
 - Diode
 - Capacitor

La figura 4.19 muestra el circuito cambiador de nivel.

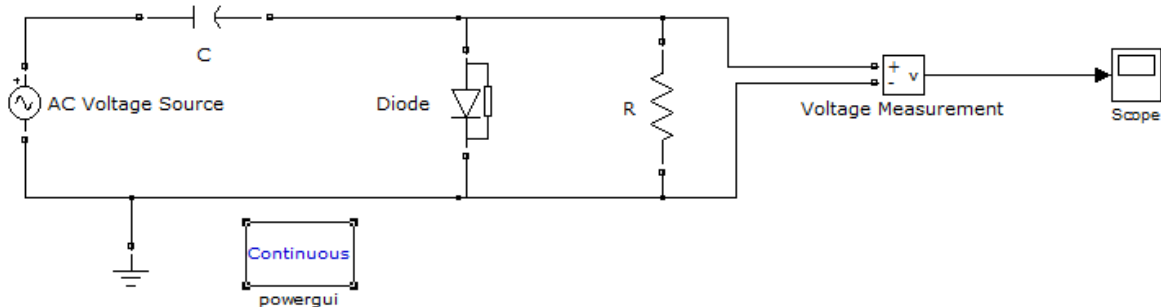


Figura 4.19: Resultado de la práctica Circuitos cambiadores de nivel

Fuente: Resultado de la práctica en Simulink en MATLAB

Parámetros

AC Voltage Source

Peak Amplitud V= 100 voltios

Phase = 0 deg

Frecuency = 60 Hz

Sample time : 0

Resistance

R= 8 ohms

Diode 1

Resistance= 0.001 ohm

Inductance = 0 H

Forward voltage = 0.7 voltios

Initial current = 0 Amperios

Capacitor

Capacitance = 1e-6 F

Señal de salida

La visualización de las señales de voltaje y corriente a la salida del circuito se muestran en la figura 4.20

Scope

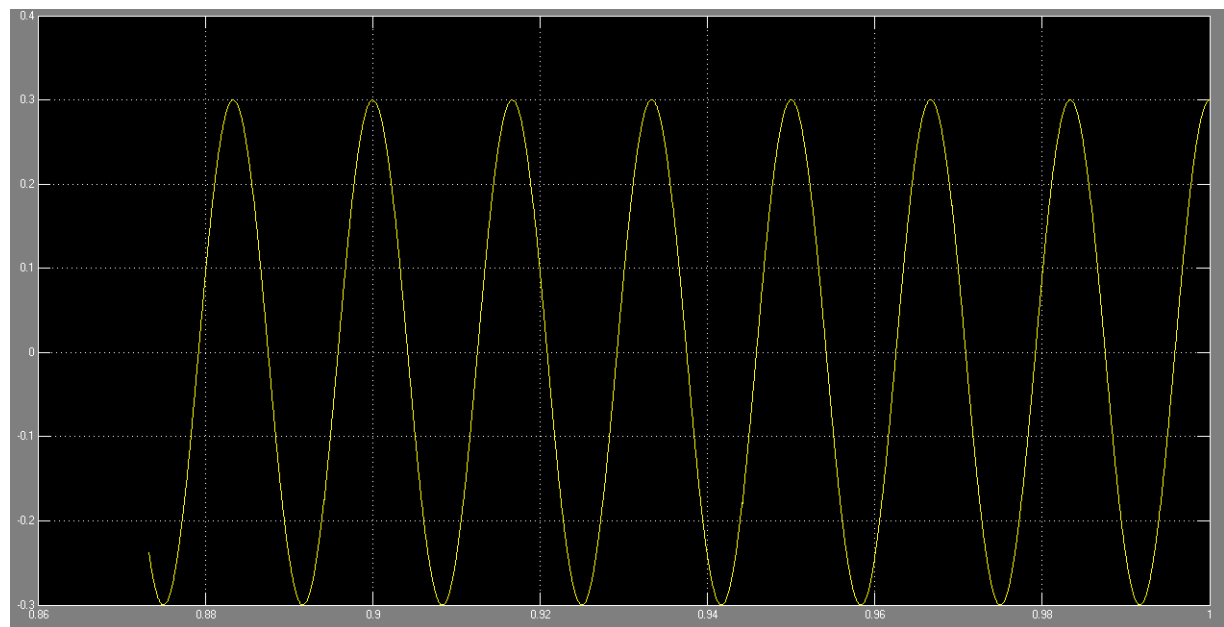


Figura 4.20: Visualización de señales de voltaje y corriente a la salida del circuito.

Fuente: Resultado de la práctica en Simulink en MATLAB

La gráfica previa muestra el desplazamiento experimentado por la señal debido a que en la configuración del circuito se encuentra un capacitor el cual en los semi-ciclos se carga y en los positivos brinda el voltaje de carga para ser sumado al voltaje de entrada lo cual genera el desplazamiento comentado previamente.

4.2.6 PRÁCTICA #6: MOSFET DECREMENTAL

El transistor de efecto de campo MOS (MOSFET), se refiere a un semiconductor Metal-Oxido, en el cual se basa el funcionamiento de este tipo de transistores, los cuales presentan una compuerta metálica la que se presenta aislada del canal del MOS por una capa de óxido.

Este dispositivo presenta dos tipos en su clasificación, siendo estos decremental e incremental.

Para esta práctica se analizará el tipo decremental, este elemento se caracteriza por presentar tres terminales, estos son drenaje, fuente y compuerta. Entre los cuales se puede encontrar tránsito de corrientes características de este dispositivo así como los voltajes necesarios para el funcionamiento del mismo, como se podrá visualizar en la simulación del circuito.

Desarrollo de la práctica

Al realizar esta práctica se podrá apreciar el funcionamiento de cada una de las partes que integran al MOSFET decremental así como los diferentes valores que se reflejan tanto en voltaje como en corriente en los componentes característicos del mismo.

A continuación se mencionan los elementos necesarios y las librerías en los que se encuentran.

Librerías:

- Sinks
 - Scope
 - Display
- Electrical sources

- DC Voltage source
- Ground
- RLC Branch
- Measurements
 - Current measurement
 - Voltage measurement
- Power electronic
 - MOSFET

Modelo: el modelo del circuito a simularse se presenta en la figura 4.21.

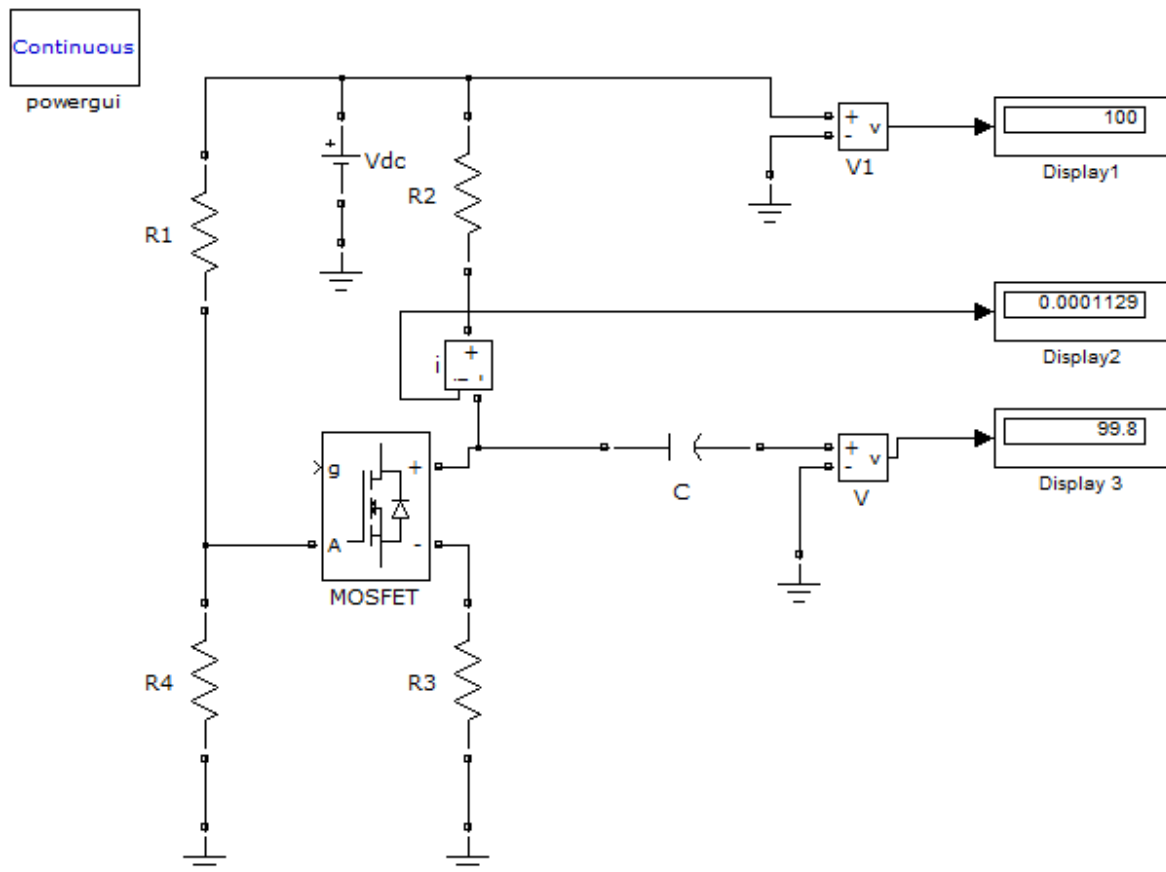


Figura 4.21: Resultado de la práctica MOSFET Decremental
Fuente: Resultado de la práctica en Simulink en MATLAB

Parámetros

MOSFET

Snubber resistance $R_s = 1e5$ ohms

Snubber capacitance $C_s = 0.1 \mu\text{F}$

$R_{on} = 1e-3$ ohms

Capacitor

Capacitance = $1e-6$ F

DC Voltage Source

Amplitud $V = 100$ voltios

Resistance 1

$R = 110000000$ ohms

Resistance 2

$R = 1800$ ohms

Resistance 3

$R = 750000$ ohms

Resistance 4

$R = 10e6$ ohms

Parámetros de salida

Display : 99.8 voltios

Display 1: 0.0001129 amperios

Display 2: 100 voltios

En estos *displays* se pueden observar los siguientes resultados: en el primero el voltaje de sumidero, en el segundo la corriente de sumidero y en el tercero el voltaje de la fuente. Estos parámetros pueden variar dependiendo del valor que se le aplique a las resistencias así como también al voltaje de entrada.

4.3 PROGRAMA DE LA ASIGNATURA ELECTRONICA I CON LAS PRACTICAS RECOMENDADAS

Diseño de las unidades de estudio (UE)

UE I: DIODOS SEMICONDUCTORES

Contenidos específicos: Introducción. El Diodo Ideal. Materiales semiconductores. Materiales extrínsecos: tipo p y tipo n. Diodo semiconductor. Niveles de resistencia. Circuitos equivalentes para diodos. Hojas de especificaciones de diodos. Diodos Zener. Análisis mediante la recta de carga. Aproximaciones de diodos. Configuraciones de diodos en serie con entradas dc. Configuraciones en paralelo y en serie-paralelo. Desarrollo de la práctica # 1 cuyo tema es Configuraciones de diodos con entrada en dc. Entradas senoidales: rectificación de media onda. Desarrollo de la práctica #2 con tema Rectificadores de media onda polarización directa. Desarrollo práctica # 3 la cual tiene como tema Rectificadores de media onda polarización inversa. Rectificación de onda completa. Desarrollo de la práctica #4 cuyo tema es Rectificador de onda completa. Recortadores. Cambiadores de nivel. Desarrollo de la práctica #5 con el tema Circuitos cambiadores de nivel. Diodos Zener. Circuitos multiplicadores de voltaje.

UE II: TRANSISTORES BIPOLARES DE UNION

Contenidos específicos: Introducción. Construcción de transistores. Operación del transistor. Configuración de base común. Acción amplificadora del transistor. Configuración de emisor común. Configuración de colector común. Límites de operación. Hoja de especificaciones de transistores. Punto de operación. Circuito de polarización fija. Circuito de polarización estabilizado en emisor. Polarización por divisor de voltaje. Polarización de dc por retroalimentación de voltaje. Diversas configuraciones de polarización. Operaciones de diseño. Redes de conmutación con transistores.

UE III: MODELAJE DE TRANSISTORES BIPOLARES

Contenidos específicos: Amplificación en el dominio de ac. Modelaje de transistores BJT. Los parámetros importantes. El modelo de transistor r_e . El modelo híbrido equivalente. Análisis a señal pequeña del transistor bipolar. Configuración de emisor común con polarización fija. Polarización mediante divisor de voltaje. Configuración de emisor común con polarización en emisor. Configuración emisor-seguidor. Configuración de base común. Configuración con retroalimentación en colector. Configuración con retroalimentación de dc en colector. Circuito equivalente híbrido aproximado. Modelo equivalente híbrido completo.

UE IV: Transistores de efecto de campo

Contenido Específico: Construcción y características de los JFET. Características de transferencia. Hojas de especificaciones. MOSFET de tipo decremental. MOSFET de tipo incremental. Polarización del FET. Configuración de polarización fija. Configuración de autopolarización. Polarización mediante divisor de voltaje. MOSFET de tipo decremental. MOSFET de tipo incremental. Modelo de señal pequeña del FET. Configuración de polarización fija del JFET. Configuración de autopolarización del JFET. Configuración de divisor de voltaje del JFET. Configuración fuente-seguidor para el JFET. Configuración de compuerta común para el JFET. MOSFET de tipo decremental. MOSFET de tipo incremental. Desarrollo de la práctica #6 cuyo tema es MOSFET decremental

CONCLUSIONES

Las conclusiones derivadas de la realización del proyecto de investigación realizado son las siguientes:

- Luego de analizar el programa de la asignatura Electrónica I, se estableció como necesidad el refuerzo del área teórica de la misma, escogiendo a MATLAB como la aplicación base para el desarrollo de prácticas sobre temas específicos de la materia.
- Se analizó y estudió las propiedades que brinda MATLAB obteniendo un conocimiento básico del manejo de sus herramientas por medio del manual Learning Matlab publicado por Mathworks, así también se realizó una revisión de los temas puntuales que se dictan en la asignatura Electrónica I.
- Habiendo aprendido sobre el manejo de MATLAB, se procedió a escoger la aplicación Simulink, la cual presenta un ambiente amigable para el usuario así como también cuenta con los elementos necesarios para el desarrollo del propósito establecido para esta investigación.
- Los temas que se tomaron en cuenta para su desarrollo práctico, fueron escogidos basándose en que contienen aspectos que podrían ser entendidos en su totalidad por los estudiantes mediante una observación en tiempo real de su funcionamiento y parámetros bajo los cuales opera.
- Habiendo desarrollado las prácticas consideradas necesarias, se procede a ingresarlas en el programa actual de la asignatura Electrónica I, quedando cubiertos algunos temas de la asignatura que son de mucha importancia en el aprendizaje de la Electrónica en general.
- El producto final de esta investigación nos muestra el cumplimiento de los objetivos estipulados para su desarrollo, así como la inserción de material de tipo práctico al programa de la asignatura Electrónica I dictada en Cuarto ciclo de la Carrera de Ingeniería en Telecomunicaciones lo cual llevara al estudiante a un aprendizaje integral y completo de la misma, así como a desarrollar habilidades de manejo del programa MATLAB el cual es muy usado en diferentes áreas de estudio de esta carrera.

RECOMENDACIONES

Las recomendaciones derivadas de la ejecución de este proyecto de investigación son las siguientes:

- Tomando en cuenta el mundo competitivo y de desarrollo tecnológico continuo al cual se enfrentan los estudiantes y profesionales, es necesario que cada una de las materias que conforman la malla curricular de la Carrera de Ingeniería en Telecomunicaciones tengan un alto porcentaje de desarrollo práctico de los temas a analizar, ya que esta es la única forma de afianzar correctamente los conocimientos adquiridos.
- Analizar otros programas o aplicaciones que brinden desarrollo práctico de temas técnicos, y establecer nuevos programas de estudio para que los estudiantes a la hora de experimentar el mundo laboral tengan una noción más real de los conocimientos adquiridos y puedan sentirse seguros en su desempeño.

GLOSARIO

- **BJT.-** (*Silicon controlled rectifier*), rectificador controlado de silicio
- **AC.-** (*Alternating Current*), corriente alterna
- **DC.-** (*Direct Current*), corriente directa
- **JFET.-** (*Junction Field-Effect Transistor*), transistor de efecto de campo de unión
- **MOSFET.-** (*Metal Oxide Semiconductor Field Effect Transistor*), transistor de efecto de campo basado en la estructura MOS
- **VDHL** (*VHSIC Hardware Description Language*), lenguaje de descripción de hardware
- **PC.-** *Personal Computer*, Computadora Personal u Ordenador Personal.
- **USB:** Bus universal en serie
- **FORTRAN:** lenguaje de programación alto nivel
- **DSP:** (*design signal procesing*), procesamiento de diseño de señal
- **API.-** Interface de aplicación de programa MATLAB
- **IDE.-** (*integrated development environment*), entorno de desarrollo integrado
- **UE.-** Unidades de Estudio
- **AMD.-** (*Advanced Micro Devices*), micro equipos avanzados
- **UCSG.-** Universidad Católica de Santiago de Guayaquil

BIBLIOGRAFIA

Aranzabal, A. (2001). *El Cambiador de Nivel de Continua*. Recuperado el 20 de Junio de 2011, de Electrónica Básica:

http://www.sc.ehu.es/sbweb/electronica/elec_basica/tema4/Paginas/Pagina19.htm

Archivo: Tensión Rectificada Media Onda. png. (2011). Recuperado el 16 de Junio de 2011, de Wikipedia.org:

http://es.wikipedia.org/wiki/Archivo:Tensi%C3%B3n_rectificada_media_onda.png

Camarena, Y., & Flores, M. (2009). *Sistema Mínimo de Electrocirugía para Corte y Coagulación a 1 MH*. Recuperado el 16 de Abril de 2011, de Universidad Autónoma Metropolitana: <http://148.206.53.231/UAMI14500.pdf>

Cid, R. d. (2007). *Manual del Laboratorio de Ingeniería Eléctrica 2*. Recuperado el 15 de Abril de 2011, de USAC Facultad de Ingeniería:

http://electrotecniausac.yolasite.com/resources/Manual_IE-2_V-3.0.pdf

CODETEL. (2009). *Componentes Electrónicos*. Recuperado el 15 de Abril de 2011, de www.solociencia.com: <http://www.solociencia.com/electronica/electronica-componentes.html>

Computer Vision with MATLAB. (s.f.). Recuperado el 20 de Junio de 2011, de www.mathworks.com:

Diodo. (2011). Recuperado el 15 de Abril de 2011, de Wikipedia.org: <http://es.wikipedia.org/wiki/Diodo>

Giner, S. (2008). *Curso de MATLAB*. Recuperado el 10 de Abril de 2011, de <http://sedici.unlp.edu.ar/ARG-UNLP-EBook-0000000013/5999.pdf>

Interfaces Físicas. (s.f.). Recuperado el 3 de Mayo de 2011, de

<http://www.dtic.upf.edu/~jlozano/interfaces/interfaces8.html>

Introducción al Matlab. (s.f.). Recuperado el 16 de Abril de 2011, de Regulación Automática: <http://usuarios.multimania.es/automatica/temas/tema1/intmatlab/intmat.htm>

Matlab. (s.f.). Recuperado el 20 de Abril de 2011, de [Ilustrados.com](http://www.ilustrados.com): <http://www.ilustrados.com/publicaciones/EpyVZFFppEfNrxMWLL.php>

Recortadores en Paralelo. (2011). Recuperado el 21 de Junio de 2011, de [Galeon.com](http://gafuhe.galeon.com): <http://gafuhe.galeon.com/tarea4.htm>

Rectificador de Media Onda. (2011). Recuperado el 19 de Junio de 2011, de Electrónica Unicrom: http://www.unicrom.com/Tut_rectificador_media_onda.asp

Simulink. Simulation and Model-Based Design. (2004). Recuperado el 16 de Junio de 2011, de The MathWorks: <http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf>

TP de Introducción a MATLAB/SIMULINK. (s.f.). Recuperado el 10 de Abril de 2011, de UNR-Ingeniería Departamento de Electrónica:

http://www.fceia.unr.edu.ar/dsf/files/TP_Introd.PDF

Transistor. (2011). Recuperado el 15 de Abril de 2011, de Wikipedia.org: <http://es.wikipedia.org/wiki/Transistor>

Transistor Bipolar NPN Original. (s.f.). Recuperado el 20 de Junio de 2011, de Electrónica González:

http://www.electronicagonzalez.com/3401_Transistor+Bipolar+NPN+Original.php