



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA ELECTRÓNICA EN TELECOMUNICACIONES Y REDES

**"DESARROLLO DE PRÁCTICAS DE MATLAB PARA LABORATORIO DE ANÁLISIS DE
SEÑALES"**

TESIS

Previa la obtención del título de

INGENIERO EN ELECTRÓNICA Y COMPUTACIÓN

Presentado por:

HERNANDO EDISON SANTILLÁN ANDRADE

RIOBAMBA-ECUADOR

2010

Dedicatoria

Dedico este trabajo de tesis a Dios, a mis padres Alberto y Germania, a ellos que hasta estos días me ayudan a sobrellevar los avatares de la vida, con su tenacidad y fortaleza, a Laurita mi señora, sustento espiritual y material de este pequeño paso hacia adelante en mi carrera profesional, le agradezco su bondad y paciencia durante este tiempo.

A todos mis seres queridos entre ellos la memoria de mis abuelitas Hortensia y Luz María recordadas por siempre, la memoria de mis tíos queridos Jorge y Gonzalito, de Jorgito que en esta vez le extrañaré su libro de recompensa cada que he alcanzado una nueva meta a nivel académico, su recuerdo y ejemplo está siempre vigente entre todos los sobrinos que le respetamos tanto, la memoria de mi suegro el Señor Estuardo Ruiz que me brindó su apoyo y confianza en los momentos más duros de mi carrera, que de manera directa influyó en la continuación de mis estudios para conseguir esta nueva meta académica, por supuesto el recuerdo de la memoria de mi hermano Gustavo, pilar de nuestros logros familiares del pasado, hoy y siempre.

Por otro lado a todos quienes hemos formado parte de la Escuela de Electrónica y Tecnología en Computación entre profesores y alumnos, desde sus inicios hasta hoy y siempre, un proyecto muy ambicioso que ha rendido sus frutos en beneficio de la comunidad local y nacional, formando ciudadanos de bien para el progreso de nuestro país.

Agradecimiento

De la manera más sentida, la colaboración de los ingenieros Paul Romero y Wilson Baldeón, docentes de la facultad, que a través de su consejo y precisos detalles me han permitido terminar este trabajo de graduación.

Además de la gran ayuda de las señoras secretarías de la Escuela de Ingeniería Electrónica y del Vicedecanato por su ágil y proactiva ayuda en todos los trámites para la consecución de este trabajo.

Y de manera especial a Iván Ruiz, técnico en electrónica de las Fuerzas Armadas (retirado) su estímulo y conocimiento han sembrado las bases en lo que se refiere a mis destrezas prácticas en este campo de las Telecomunicaciones electrónicas, a él un sincero reconocimiento.

Derechos de Autoría.

“Yo, HERNANDO EDISON SANTILLÁN ANDRADE soy responsable de las ideas, doctrinas y resultados expuestos en esta tesis; y, el patrimonio intelectual de la Tesis de Grado pertenece a la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO”.

FIRMAS

ING. IVÁN MENES _____

DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ING. JOSÉ GUERRA _____

DIRECTOR ESC. ING. ELECTRÓNICA EN TELECOMUNICACIONES Y REDES

ING. PAUL ROMERO _____

DIRECTOR DE TESIS

ING. WILSON BALDEÓN _____

MIEMBRO DE TESIS

TEC. CARLOS RODRÍGUEZ _____

DIRECTOR CENTRO DE DOCUMENTACIÓN

NOTA DE LA TESIS _____

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

INDICE DE ABREVIATURAS

CAPÍTULO I

GENERALIDADES

1.1 ANTECEDENTES -14-

1.2 JUSTIFICACION DEL PROYECTO DE TESIS -16-

1.3 OBJETIVOS -16-

1.4 HIPOTESIS -17-

CAPÍTULO II

MARCO TEÓRICO

2.1 INTRODUCCIÓN A MATLAB -18-

2.2 ¿COMO SE USA MATLAB EN LA INDUSTRIA? -19-

2.3 USO EN LA INGENIERÍA ELÉCTRICA Y ELECTRÓNICA -20-

2.4 USO EN DINÁMICA DE FLUIDOS -20-

2.5 RESOLUCIÓN DE PROBLEMAS EN INGENIERÍA Y CIENCIAS -20-

2.6 SIMULINK -22-

2.7 HERRAMIENTA PARA EL DISEÑO BASADO EN MODELOS -22-

2.8 HERRAMIENTA PARA SIMULACIÓN -23-

2.9 HERRAMIENTA PARA ANÁLISIS -23-

2.10 CÓMO SIMULINK INTERACTÚA CON EL ENTORNO DE MATLAB -23-

CAPÍTULO III

DESARROLLO DE LAS PRÁCTICAS	
3.1 PRÁCTICA DE SEÑALES Y ESPECTROS	-25-
3.2 PRÁCTICA ONDAS DETERMINÍSTICAS Y ALEATORIAS	-43-
3.3 PRÁCTICA CAPACIDAD DE CANAL Y SISTEMAS DE COMUNICACIONES IDEALES	-51-
3.4 PRÁCTICA POTENCIA	-55-
3.5 PRÁCTICA SERIE DE FOURIER	-58-
3.6 PRÁCTICA SERIES DE FOURIER USANDO ARCHIVOS-M	-76-
3.7 PRÁCTICA TEOREMA DE MUESTREO	-84-
3.8 FILTRO PASABANDA SIMULINK	-93-
3.9 PRÁCTICA RUIDO BLANCO GAUSSIANO ADITIVO	-103-
3.10 PRÁCTICA TRANSFORMADA DISCRETA DE FOURIER	-116-
CAPÍTULO IV	
ANÁLISIS DE RESULTADOS	
4.1 DESVIACIÓN ESTÁNDAR DE UNA MUESTRA	-122-
4.2 CÁLCULO DE LA DESVIACIÓN ESTÁNDAR S A PARTIR DE UNA POBLACIÓN CONOCIDA	-123-
4.3 DESVIACIÓN ESTÁNDAR DE UNA POBLACIÓN σ	-123-
4.4 CÁLCULO DEL ESTADÍSTICO DE PRUEBA, DISTRIBUCIÓN CHI-CUADRADA	-124-
4.5 PRUEBA DE HIPÓTESIS	-124-
CONCLUSIONES	
RECOMENDACIONES	
RESUMEN	
SUMMARY	
ANEXOS	
BIBLIOGRAFÍA	

ÍNDICE DE FIGURAS

Figura II.01. Arrancando MATLAB Revisión 2008A.....	-19-
Figura II.02. Inicio de MATLAB y Simulink.....	-22-
Figura III.03. Similitud a una Calculadora Científica.....	-26-
Figura III.04. Variable tipo texto, uso de CLC.....	-26-
Figura III.05. Escritura primer archivo-M.....	-27-
Figura III.06. Búsqueda de otra carpeta de trabajo para MATLAB.....	-27-
Figura III.07. Un vector y una matriz.....	-28-
Figura III.08. Creación vector tiempo t, para simulación de señales.....	-28-
Figura III.09. Función señal Seno sencilla.....	-29-
Figura III.10. Listado: SEGUNDO_ARCHIVO.M.....	-30-
Figura III.11. Señales Seno y Coseno.....	-30-
Figura III.12. Listado: TERCER_ARCHIVO.M.....	-31-
Figura III.13. Ayuda en MATLAB.....	-32-
Figura III.14. Salida TERCER_ARCHIVO.M.....	-32-
Figura III.15. Señales Básicas.....	-33-
Figura III.16. Salida bloque SCOPE a) Onda Seno b) Onda Coseno.....	-34-
Figura III.17. Configuración del bloque DE LA FUNCIÓN Seno(t).....	-34-
Figura III.18. Fase pi/2 de una onda Seno transformada en Coseno (t).....	-35-
Figura III.19. Inicio de Simulink.....	-36-
Figura III.20. Biblioteca Sources/Sine Wave (Fuentes/Onda Seno).....	-36-
Figura III.21. Biblioteca Commonly Used Blocks/Scope (Bloques Comúnmente usados/Osciloscopio)	-37-
Figura III.22. Ventana Nuevo modelo.....	-38-
Figura III.23. Implementación de nuestra simulación.....	-38-
Figura III.24. Valores por omisión onda seno (Sine Wave) de la biblioteca Fuentes (Sources).....	-39-

Figura III.25. Cambio de nombre al bloque Sine Wave por Onda Seno.....	-39-
Figura III.26. Cambio de nombre del bloque Scope a Osciloscopio.....	-40-
Figura III.27. Corte de pantalla a 2 salidas.....	-40-
Figura III.28. Conexión de señales.....	-41-
Figura III.29. Autoescala de la salida de la Simulación.....	-41-
Figura III.30. Abrir modelo ya creado.....	-42-
Figura III.31. Modulador/Demodulador Digital por Desplazamiento de Frecuencia FSK (Frequency Shift Keying).....	-44-
Figura III.32. Bloque Generador Binario Aleatorio de Bernoulli.....	-45-
Figura III.33. Bloque que Añade Ruido Blanco Gaussiano a la Simulación.....	-46-
Figura III.34. Bloque Scope (Osciloscopio).....	-46-
Figura III.35. Modelo Casi terminado.....	-47-
Figura III.36. Configuración Bloque Scope.....	-48-
Figura III.37. Guardar Modelo Simulink Creado.....	-48-
Figura III.38. Modelo Final listo para simularse.....	-49-
Figura III.39. Cambio interactivo del tiempo de ejecución de la simulación.....	-49-
Figura III.40. Salida del bloque Scope (Osciloscopio) de una señal determinística y de una señal aleatoria.	-50-
Figura III.41. Ejecución archivo-M cálculo capacidad del canal usando el teorema de Shannon.	-54-
Figura III.42. Convención de polaridad utilizada para el voltaje y la corriente.....	-56-
Figura III.43. Salida relación Potencia en función del tiempo.....	-57-
Figura III.44. Bloque Suma biblioteca Bloques comúnmente usados.....	-60-
Figura III.45. Búsqueda de bloque Constante.....	-61-
Figura III.46. Bloques Constante y Ganancia de la biblioteca Bloques Comúnmente usados de Simulink.	-62-
Figura III.47. Bloques para modelar la Serie de Fourier de una Onda Senoidal con Rectificación de Onda Completa.....	-62-
Figura III.48. Representación Coeficiente $\cos(2\omega t)$ en el bloque Sin Wave.....	-63-
Figura III.49. $\cos(4\omega t)$, siendo $\omega = 2\pi f$	-64-
Figura III.50. Cambio de valor constante	-65-
Figura III.51. Ganancia del término $\cos(2\omega t)$	-66-
Figura III.52. Otra forma de representar la ganancia $-\frac{4}{\pi(3)(5)}$ de $\cos(4\omega t)$ de la ecuación 4.1.	-66-

Figura III.53. Contraste de las dos formas de ganancias de los 2 últimos coeficientes de esta serie. a) ganancia de $\cos(6\omega t)$ b) ganancia de $\cos(8\omega t)$	-67-
Figura III.54. Añadiendo señales de entrada al bloque Suma (Sum)	-68-
Figura III.55. División de las pantallas del Osciloscopio simulado.....	-68-
Figura III.56. Tip Uso de la tecla CONTROL.....	-69-
Figura III.57. Modelo terminado corresponde a la ec 2.1 $v(t) = \frac{2V}{\pi} - \frac{4V \cos(2\omega t)}{\pi(1)(3)} - \frac{4V \cos(4\omega t)}{\pi(3)(5)} - \frac{4V \cos(6\omega t)}{\pi(5)(7)} - \frac{4V \cos(8\omega t)}{\pi(7)(9)}$	-69-
Figura III.58. Senoide con Rectificación de onda completa y sus componentes.....	-70-
Figura III.59. Circuito Rectificador de Onda completa.....	-71-
Figura III.60. a). Rectificador en puente.....	-73-
Figura III.61. b) Osciloscopio digital simulado muestra la onda rectificada de onda completa.	-73-
Figura III.62. Encapsulado KBU de 4 diodos en puente.....	-74-
Figura III.63. Salida del archivo-M PRACTICA4_2_1.M.....	-76-
Figura III.64. Salida del archivo-M PRACTICA4_2_1.M.....	-77-
Figura III.65. Salida del archivo-M PRACTICA4_2_3.M.....	-78-
Figura III.66. Salida del archivo-M PRACTICA4_2_4.M.....	-80-
Figura III.67. Salida del archivo-M PRACTICA4_2_5.M.....	-81-
Figura III.68. Salida del archivo-M PRACTICA4_2_6.M.....	-82-
Figura III.69. Ejecución del guión-M.....	-88-
Figura III.70. Salida del guión.....	-88-
Figura III.71. Nuevo cálculo, frecuencia de muestreo.....	-89-
Figura III.72. Nueva salida con fm=500.....	-89-
Figura III.73. Respuesta en tiempo y frecuencia.....	-92-
Figura III.74. Circuito RLC representado en el dominio s.....	-95-
Figura III.75. Bloque Función de Transferencia dominio s.....	-96-
Figura III.76. Entrada de los coeficientes de la función de transferencia planteada.....	-97-
Figura III.77. Bloque escalón del Modelo.....	-98-
Figura III.78. Modelo en construcción con 2 bloques: Entrada y Función de Transferencia.....	-98-
Figura III.79. Modelo Scope (Osciloscopio).....	-99-
Figura III.80. Modelo Terminado.....	-100-

Figura III.81. Respuesta en frecuencia de un filtro RLC pasabaja pasivo.....	-101-
Figura III.82. Bloque Generador Binario de Bernoulli.....	-106-
Figura III.83. Configuración Bloque Generador de Señales aleatorias.....	-107-
Figura III.84. Bloque M-FSK Modulador FM Digital.....	-107-
Figura III.85. Modelo actual Modulador/Demodulador FM Digital.....	-108-
Figura III.86. Bloque AWGN (Ruido Blanco Gaussiano Aditivo o TÉRMICO).....	-109-
Figura III.87. Bloques de la simulación actuales.....	-109-
Figura III.88. Bloque Scope (Osciloscopio).....	-110-
Figura III.89. Dividir Scope en 2 canales.....	-110-
Figura III.90. Osciloscopio listo para recibir las señales.....	-111-
Figura III.91. Modelo actual con todos los bloques, listos a ser unidos sus canales de flujo.	-111-
Figura III.92. Modelo Simulink Ruido Gaussiano o Térmico terminado.....	-112-
Figura III.93. Análisis de las señales generadas por el modelo.....	-112-
Figura III.94. Bloque Generador Pulsos Enteros Aleatorios.....	-113-
Figura III.95. Simulación QAM usando canal AWGN (Ruido Blanco Aditivo Gaussiano) o Ruido Térmico.	-114-
Figura III.96. Señal Digitalizada de entrada.....	-114-
Figura III.97. Gráfica de Fase y Amplitud Q-AM.....	-115-
Figura III.98. Resultado de una señal truncada escalón en el dominio del tiempo y frecuencia.	-119-
Figura III.99. Ingreso de coeficientes de la función de transferencia recomendada.....	-120-
Figura III.100. Modelo de Transformada Z a investigar.....	-120-
Figura III.101. Salida de la práctica recomendada.....	-121-
Figura IV.102. Ubicación del Estadístico de Prueba en la Distribución Chi-cuadrada para la prueba de hipótesis H1.....	-126-

ÍNDICE DE TABLAS

Tabla IV.I. Notas y equivalencias usadas en la muestra estadística.....	-123-
Tabla IV.II. Valores calculados de la desviación estándar, poblacional, y población.....	-124-
Tabla IV.III. Datos de la tabla A-4 (Anexa) de la Distribución Chi-Cuadrada.....	-126-

INDICE DE ABREVIATURAS

EDA	Electronic D esign A utomation (Automatización del Diseño Electrónico).
MATLAB	M atrix L aboratory (Laboratorio de Matrices).
SIMULINK	Modela, simula, y analiza sistemas dinámicos, software que es parte de MATLAB.
B	Ancho de banda de banda base.
C	Capacidad del canal.
E	Energía.
<i>f</i>	Frecuencia.
<i>f_c</i>	Frecuencia de portadora.
<i>f_s</i>	Frecuencia de muestreo.
H	Entropía.
K	Número de bits en una palabra binaria que representa un mensaje digital.
<i>m(t)</i>	Forma de onda de mensaje (modulación).
<i>r(t)</i>	Señal recibida con ruido.
<i>s(t)</i>	Señal.
<i>ñ(t)</i>	Señal distorsionada.
S/N	Relación potencia señal a potencia de ruido.
<i>t</i>	Tiempo.
T	Un intervalo de tiempo.
<i>T_b</i>	Período de bit.
<i>v(t)</i>	Una forma de onda de voltaje.
<i>w(t)</i>	Una forma de onda.
λ	Longitud de onda.
<i>J_n(.)</i>	Función de Bessel de primer tipo, n-ésimo orden.

CAPÍTULO I

GENERALIDADES

1.1 ANTECEDENTES

El desarrollo de la carrera de Ingeniería Electrónica de la Facultad de Informática y Electrónica de la Escuela Superior Politécnica del Chimborazo, ha definido su actual pensum de estudios al crear dos nuevas carreras profesionales la una en el área de las Telecomunicaciones y Redes informáticas y la carrera de Control y Redes Industriales, dentro de las mismas se ha contemplado una asignatura base en el conocimiento de las mencionadas especialidades, se trata de la materia de “ANÁLISIS DE SEÑALES”, cátedra común para el fortalecimiento académico del futuro ingeniero electrónico.

Necesitándose para la demostración de la teoría impartida en esta materia de herramientas tanto en software como hardware; tales como MATLAB, un lenguaje técnico moderno.

MATLAB, originalmente escrito en lenguaje de programación Fortran por los inicios de los ochentas por el científico Cleve Moler, uno de los fundadores de MATHWORKS y jefe científico de esta empresa. Pionero en este tipo de programas, el mismo que está basado en matrices, al momento tiene incorporado alrededor de 80.000 funciones, fuera de las Cajas de Herramientas

(ToolBoxes), posee un lenguaje de programación propio en base a guiones, llamados archivos M y diversas interfaces con muchas tarjetas electrónicas de propósito específico. Como es claro se ha venido desarrollando mejoras a esta suite que es denominada el lenguaje de la computación técnica, avalada por cientos de universidades alrededor del mundo, que además tiene un soporte personalizado al momento de querer conseguir más ayuda, basa su conocimiento en los llamados Webinar, tan solo al inscribir su email al sitio web www.mathworks.com. MATLAB, tiene además SIMULINK, que es el software orientado a objetos de simulación, posee además de una interface gráfica de desarrollo de aplicaciones orientada a objetos y clases, llamada GUIDE (Graphical User Interface Tools- Herramientas para Interfaz Gráfica de Usuario), que nos ayuda de manera intuitiva a generar nuestras aplicaciones en forma visual, la programación integrada está basada en la lógica de la programación de los archivos M. Sobre el tema de las ToolBoxes, se debe saber que en algún momento, alguien que, partiendo de este trabajo de investigación concerniente al desarrollo de estas prácticas de laboratorio de las asignaturas de Análisis de Señales, pueda extender sus estudios pensando alcanzar metas tales como el desarrollo de sus propios guiones o ya más avanzado, la creación de Cajas de Herramientas personalizadas, como ya lo vienen haciendo muchos autores de prestigiosos textos en muchas áreas de la ciencia, tal es el caso de una Toolbox para Robótica, desarrollada por Barrientos y otros. Se pretende entonces con este estudio dar las bases necesarias para que el estudiante no agote esfuerzos en la consecución de nuevos proyectos, debido a las bases teórico-prácticas que se pretende instaurar en ellos a través de MATLAB. Como hemos dicho, muchas universidades de renombre del mundo, por citar unas pocas Urbana Champaign de Illinois, California Polytechnic State University de San Luis Obispo en California, a través de breves documentos o artículos, comparten sus proyectos de investigación vía Internet usando el sitio web de MATHWORKS como anfitrión (Academic Digest). Es por tanto uno de los principales motivos que nos alientan a desarrollar esta tesis con la visión de que el formato de prácticas a realizarse sean revisadas, probadas, y analizadas en forma estadística y poder comprobar nuestra hipótesis planteada, es decir que MATLAB es una herramienta efectiva en el cálculo, modelado y simulación de proyectos de ingeniería electrónica en el campo del análisis de señales. La versión de MATLAB que usaremos para nuestra investigación es la 7.6.0.324 (R2008a), por estar ya instalada en los laboratorios de la Escuela de Ingeniería Electrónica. Es válido decir que estas versiones siempre están siendo actualizadas y que todos los experimentos realizados planteados en este trabajo de investigación son soportados por algunas versiones anteriores y posteriores en su gran mayoría, esta ha sido la característica de trabajo de

parte de la empresa que ha desarrollado MATLAB, para el disfrute de la comunidad científica moderna por el lapso de tres décadas.

1.2 JUSTIFICACION DEL PROYECTO DE TESIS.

La falta de documentación y bibliografía técnica dentro de la institución (ESPOCH) sobre la aplicación de MATLAB en la Ingeniería Electrónica en sus áreas de especialidad tanto de Telecomunicaciones o Control, y puntualmente en lo que se refiere a la materia de ANÁLISIS DE SEÑALES. Por ser además una asignatura dentro del pensum de estudios de la carrera de Ingeniería Electrónica en su formación básica de quinto semestre a través de la materia de Herramientas EDA. Por ser una sofisticada herramienta con la cual se puede compartir estudios de investigación científica entre educandos y profesionales del ramo de manera rápida y efectiva. Tomando en cuenta que en sus inicios MATLAB ha sido una herramienta que ha sido ya aplicada en estudios de Ingeniería Eléctrica, por estar MATLAB basado en matrices. MATLAB fue escrito originalmente por Cleve Moler en los inicios de los años 80s con tan solo 80 funciones y que actualmente contiene alrededor de 80.000 funciones, soporte de archivos-M y Cajas de Herramientas (ToolBoxes) para muchas áreas del conocimiento, por tanto es mi deseo al presentar este tema de tesis, que esta potente herramienta MATLAB, sea explotada en nuestro perfil académico. La presente tesis está pensada para resolver todas las interrogantes y necesidades planteadas en los epígrafes anteriores y demostrar en forma fehaciente mediante métodos científicos y estadísticos en lo que se refiere a la prueba de hipótesis que la misma es viable y puede ser aplicada de inmediato en las aulas, con la respectiva tutoría y venia del profesor titular de las materias de Análisis de Señales y Herramientas EDA.

1.3 OBJETIVOS

OBJETIVO GENERAL. La presente tesis de carácter investigativa pretende estudiar, investigar, desarrollar y evaluar la herramienta científica MATLAB como una herramienta EDA aplicada y obtener un material de apoyo para la materia de “ANALISIS DE SEÑALES”, vigente en la malla curricular académica común de las carreras de Ingeniería Electrónica de la ESPOCH, tanto de Telecomunicaciones y Control.

OBJETIVOS ESPECIFICOS:

Aprovechar MATLAB para diseñar y elaborar documentos técnicos.

Elaborar un medio de comprensión del fundamento teórico, a través de prácticas diseñadas y elaboradas para cada uno de los capítulos temáticos de la materia de Análisis de Señales.

Analizar en forma estadística su uso y beneficio de las prácticas de laboratorio, con miras a optimizar el mejor aprovechamiento de esta importante herramienta EDA.

Implementar estas prácticas como parte del laboratorio de la asignatura de ANÁLISIS DE SEÑALES, básica en la formación de un ingeniero electrónico en cualquiera de sus especialidades.

Evaluar el uso de las prácticas elaboradas y realizar un estudio estadístico de su aceptación y prueba de hipótesis.

Documentar y producir un CD que puede ser el modelo base para futuros trabajos de investigación en próximos trabajos de investigación, relacionados al área.

1.4 HIPOTESIS:

DETERMINAR Y DEMOSTRAR EL SOFTWARE DE MATLAB COMO HERRAMIENTA ÚTIL DE DISEÑO, MODELACIÓN Y CÁLCULO EN LA ASIGNATURA DE ANÁLISIS DE SEÑALES.

CAPÍTULO II

MARCO TEÓRICO

2.1 INTRODUCCIÓN A MATLAB

MATLAB es una de las muchas sofisticadas herramientas de computación disponibles en el comercio para resolver problemas de matemáticas, tales como Maple, Mathematica y MathCAD. Todas estas herramientas junto con MATLAB tienen sus fortalezas y debilidades. Cada una permitirá efectuar cálculos matemáticos básicos, pero difieren del modo como se manejan los cálculos simbólicos y procesos matemáticos más complicados, como la manipulación de matrices. Por tanto MATLAB (Laboratorio de Matrices) en sus inicios escrito por el científico Cleve Moler, y fundador de la empresa MATHWORKS, es superior en el tratamiento de matrices, mientras que Maple lo supera en cálculos simbólicos. En un nivel fundamental, se puede pensar que estos programas son sofisticadas calculadoras con base en un ordenador. Son capaces de realizar las mismas funciones que una calculadora científica, y muchas más.

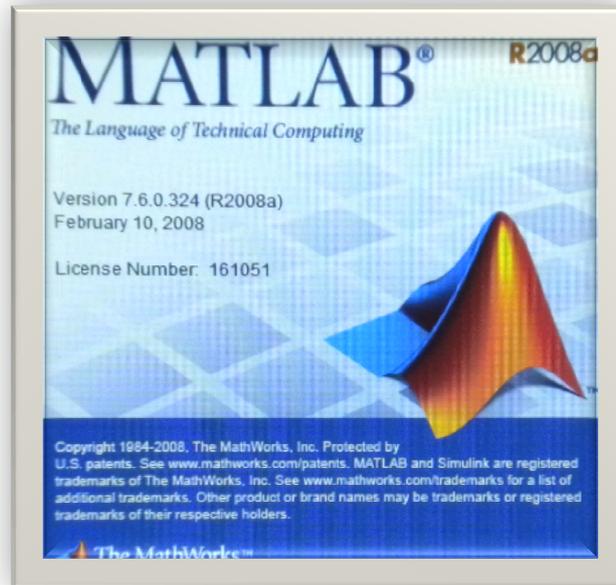


Figura II.01. Arrancando MATLAB Revisión 2008A.

Si tiene una computadora en su escritorio, descubrirá que usará MATLAB en lugar de su calculadora incluso para la más simple de sus aplicaciones matemáticas, por ejemplo para el balance de su chequera. En muchas clases ingeniería, la realización de cálculos con un programa de computación matemático como MATLAB sustituye la programación de computadora más tradicional. Actualmente MATLAB se ha convertido en una herramienta estándar para ingenieros y científicos.

Dado que MATLAB es tan fácil de usar, muchas tareas de programación se llevan a cabo con él. Sin embargo, MATLAB no siempre es la mejor herramienta para usar en una tarea de programación. El programa destaca en cálculos numéricos, especialmente en los relacionados con matrices y gráficas, MATLAB de hecho es un programa grande de aplicación que se escribió originalmente en FORTRAN y después se re-escribió en C, precursor del C++. Por lo general, los programas de alto nivel no ofrecen acceso fácil a la graficación, que es una aplicación en la que destaca MATLAB. El área principal de interferencia entre MATLAB y los programas de alto nivel es el “procesamiento de números”: programas que requieren cálculos repetitivos o el procesamiento de grandes cantidades de datos. Tanto MATLAB como los programas de alto nivel son buenos en el procesamiento de números.

2.2 ¿COMO SE USA MATLAB EN LA INDUSTRIA?

La habilidad para usar herramientas tales como MATLAB se convirtió rápidamente en un requisito para muchos puestos de ingeniería. En una reciente búsqueda de empleo usando un popular navegador, se encontró el siguiente anuncio:

“...se busca un ingeniero de sistema de pruebas con experiencia en procesamiento de análisis de señales (DSP)... Sus responsabilidades incluyen modificación de guiones de MATLAB (M-scripts), ejecución de simulaciones en Simulink y el análisis de los datos del resultado. El candidato debe estar familiarizado con MATLAB y Simulink.”

Este anuncio no es raro. La misma búsqueda arrojó alrededor de 75 compañías diferentes que requerirían específicamente de MATLAB para los ingenieros que entrarán al nivel de base.

MATLAB es particularmente popular para aplicaciones de ingeniería eléctrica, aunque se usa muchísimo en los campos de ingeniería y ciencias.

2.3 USO EN LA INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

MATLAB se utiliza mucho en ingeniería eléctrica para aplicaciones de procesamiento de señales. Por ejemplo en el diseño y fabricación de chips de computadora que detecta colisiones inminentes. Esto tiene una aplicación potencial en el diseño de robots autónomos que usen la visión para navegar y en particular en aplicaciones para la seguridad en automóviles. En el campo del estudio de señales en lo que se refiere a la Electrónica, que es el motivo del desarrollo de esta tesis, como es por ejemplo el estudio de las series de Fourier de señales cuadradas, triangulares, diente de sierra, o impulso, de idéntica forma las señales que se estudian en el área del control automático en donde MATLAB es de gran ayuda para la comprensión de los sistemas con o sin retroalimentación, gráficas de Bode, gráficas de Nyquist. Además en el estudio de antenas, radares, microondas y medios guiados o no guiados.

2.4 USO EN DINÁMICA DE FLUIDOS

Los cálculos que describen velocidades de fluidos (rapideces y direcciones) son importantes en varios campos. En particular, a los ingenieros aeroespaciales les interesa el comportamiento de los gases, tanto afuera de una aeronave o vehículo espacial como dentro de las cámaras de combustión. Visualizar el comportamiento tridimensional de los fluidos es difícil, pero MATLAB ofrece cierto número de herramientas que lo hacen más sencillo. Esto se aplica en el perfil de control industrial en lo que se refiere a la dinámica de fluidos, especialidad que se tiene ya implementada en la FIE, Escuela de Ingeniería Electrónica en Control y Redes Industriales.

2.5 RESOLUCIÓN DE PROBLEMAS EN INGENIERÍA Y CIENCIAS

En las disciplinas de ingeniería, ciencias y programación de computadoras, es importante tener un enfoque consistente para resolver los problemas técnicos. El enfoque que se plantea a continuación es útil en cursos tan distintos como en química, física, termodinámica y diseño de ingeniería. También se aplica para las ciencias sociales, como economía y sociología. Todas estas técnicas se podrán definir en este formato:

Plantear el problema

- ✚ En esta etapa con frecuencia es útil un dibujo.
- ✚ Si no se tiene una comprensión clara del problema, es improbable que pueda resolverlo.

Describir los valores de entrada (conocidas) y las salidas (incógnitas) que se requieren.

- ✚ Tenga cuidado de incluir las unidades conforme describe los valores de entrada y salida.
- ✚ El manejo descuidado de las unidades con frecuencia lleva a respuestas incorrectas.
- ✚ Identifique las constantes que tal vez requiera en el cálculo, como la constante de los gases ideales y la aceleración de la gravedad.
- ✚ Si es apropiado, en un dibujo escriba los valores que haya identificado o agrúpelos en una tabla.

Desarrollar un algoritmo para resolver el problema. En aplicaciones de cómputo, es frecuente que esto se logre con una prueba de escritorio. Para ello necesitará:

- ✚ Identificar cualesquiera ecuaciones que relacionen los valores conocidos con las incógnitas.
- ✚ Trabajar con una versión simplificada del problema, a mano o con calculadora.

Resolver el problema. En este trabajo de tesis, esto se intentará en cada uno de los laboratorios que se han pensado como sustento de la materia de Análisis de Señales, usando MATLAB.

Probar la solución

- ✚ ¿Sus resultados tienen sentido físico?
- ✚ ¿Coinciden con los cálculos de la muestra?
- ✚ ¿La respuesta es la que se pedía en realidad?
- ✚ Las gráficas con frecuencia son formas útiles de verificar que los cálculos son razonables.

Si utiliza en forma consistente un enfoque estructurado de resolución de problemas, como el que se acaba de describir, descubrirá que los problemas tipo “narración” son mucho más fáciles de resolver. (MOORE, 2007)

2.6 SIMULINK

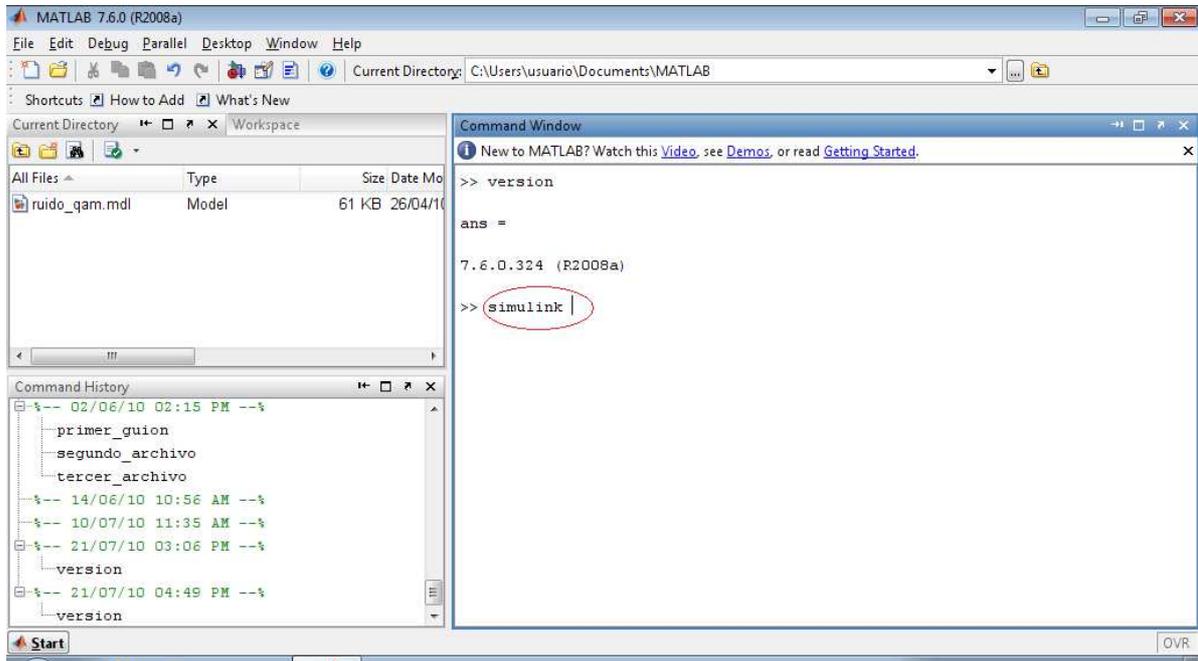


Figura II.02. Inicio de MATLAB y Simulink.

El programa Simulink es un software integrado en el entorno de MATLAB que nos permite modelar, simular y analizar sistemas dinámicos. Le habilita a proponer preguntas sobre un sistema, y ver que pasa durante su simulación.

Con Simulink, se puede rápidamente y de forma fácil construir modelos desde su inicio, o modificar modelos existentes que reúnan sus requerimientos. Simulink soporta sistemas lineales y no lineales, modelados en tiempo continuo, tiempo muestreado, o un modelo híbrido de los dos. Además se pueden crear sistemas que soporten múltiples tiempos de muestreo o que sean actualizados a diferentes velocidades.

Miles de científicos e ingenieros alrededor del mundo usan Simulink para modelar y resolver problemas reales en una variedad de industrias tales como:

- ✚ Aeroespacio y Defensa
- ✚ Industria Automotriz
- ✚ Comunicaciones
- ✚ Electrónica y Procesamiento de señales.
- ✚ Instrumentación médica.

2.7 HERRAMIENTA PARA EL DISEÑO BASADO EN MODELOS.

Con Simulink, se puede mover desde modelos idealizados lineales hasta explorar en modelos no lineales mucho más reales, como por ejemplo la resistencia del aire, arranque de motores y muchas otras cosas que se describen dentro de los fenómenos físicos reales.

Simulink provee una interfaz gráfica de usuario (GUI) para construir modelos como diagramas de bloques, permitiéndole dibujar modelos como si tuviera un lápiz y un papel. Simulink adicionalmente incluye una biblioteca de bloques de fuentes, componentes lineales o no lineales, y conectores. Si estos bloques no cumplen con sus expectativas o necesidades, se puede entonces crear sus propios bloques. El entorno gráfico interactivo simplifica el proceso de modelaje, eliminando la necesidad de formular ecuaciones diferenciales usando un lenguaje de programación o programa.

Los Modelos son jerárquicos, puede construir modelos usando las dos técnicas conocidas de Arriba-abajo o Abajo-arriba. Puede mirar un sistema a un alto nivel, entonces dar un doble clic para ver los diferentes niveles en detalle. Este acercamiento le provee la información de cómo está organizado el modelo y como sus partes interactúan.

2.8 HERRAMIENTA PARA SIMULACIÓN.

Después de definir un modelo. Puede simularlo, usando un conjunto de métodos de integración matemática, a través de sus menús interactivos o escribiendo comandos de la ventana de mandatos de MATLAB. Los menús son convenientes para un trabajo interactivo, cuando la línea de comando es útil para generar un grupo de simulaciones (por ejemplo, si está haciendo simulaciones consecutivas o quiere aplicar un parámetro dentro de un rango de valores).

Usando bloques parecidos a osciloscopios u otros bloques de visualización, se puede ver los resultados de la simulación en plena ejecución. Se puede entonces cambiar muchos parámetros y ver que pasa mediante una exploración “qué pasa si”. Los resultados de la simulación se pueden poner en el entorno de trabajo de MATLAB, llamado Workspace para procesos futuros y visualización.

2.9 HERRAMIENTA PARA ANÁLISIS.

Las herramientas de análisis incluyen herramientas para linealizar de sistemas, que pueden ser accesadas mediante la línea de mandatos de MATLAB, además de muchas herramientas en MATLAB y sus cajas de herramientas para aplicaciones. Porque MATLAB y Simulink están integrados, puede simular, analizar y revisar sus modelos en cualquiera de estos entornos en cualquier momento.

2.10 CÓMO SIMULINK INTERACTÚA CON EL ENTORNO DE MATLAB.

El programa de Simulink está completamente integrado con el entorno operativo de MATLAB. Lo requiere para correr, dependiendo de cómo se definen para su evaluación sus parámetros. Simulink puede además utilizar muchas características de MATLAB. Por ejemplo, Simulink puede usar el entorno de MATLAB para:

-  Definir modelos de entrada.
-  Almacenar salidas de modelos para análisis y visualización.

- ✚ Ejecutar funciones dentro de un modelo, a través de llamadas integradas de operadores y funciones MATLAB. (MATHWORKS, 2008)

CAPÍTULO III

DESARROLLO DE LAS PRÁCTICAS

3.1 PRÁCTICA SEÑALES Y ESPECTROS.

OBJETIVOS

Familiarizar al estudiante de la materia de Análisis de Señales, tanto de las carreras de Telecomunicaciones y Control con el entorno del software científico MATLAB.

MARCO TEÓRICO: SEÑALES Y ESPECTROS.

Las señales eléctricas de comunicaciones son cantidades que varían en el tiempo, al igual que el voltaje y la corriente. Aunque una señal existe físicamente en el dominio del tiempo, también se representa en el dominio de la frecuencia, donde se considera que la señal está integrada por componentes senoidales de diferentes frecuencias.

El análisis espectral, usando la serie y la transformada de Fourier, es uno de los métodos fundamentales de la ingeniería de comunicaciones. Permite tratar clases completas de señales que tienen propiedades similares en el dominio de la frecuencia, en vez de complicarse con un análisis detallado en el dominio del tiempo de señales individuales.

Este extracto obtenido de la obra de Sistemas de Comunicación (CARLSON, 2007) en su página 18, infiere que para el análisis de señales sinusoidales y su mejor entendimiento proponen en esta obra analizar las mismas en el dominio de la frecuencia. Sin embargo luego de la creación de nuestro primer modelo hecho con el programa SIMULINK de MATLAB comprobamos que resulta muy intuitiva para el lector la comprensión de estas señales en el dominio del tiempo, comprobando nuevamente la hipótesis planteada en este trabajo de tesis.

DESARROLLO DE LA PRÁCTICA.

PRIMEROS PASOS CON MATLAB

Para comenzar a usar MATLAB, sólo necesita prestar atención a la ventana de comandos. En ella se puede realizar cálculos en forma similar a como lo hace en una calculadora científica. Por ejemplo al calcular 5^2 , el coseno de π , o tal vez la raíz cuadrada de 3, así lo apreciamos en la figura III.03.

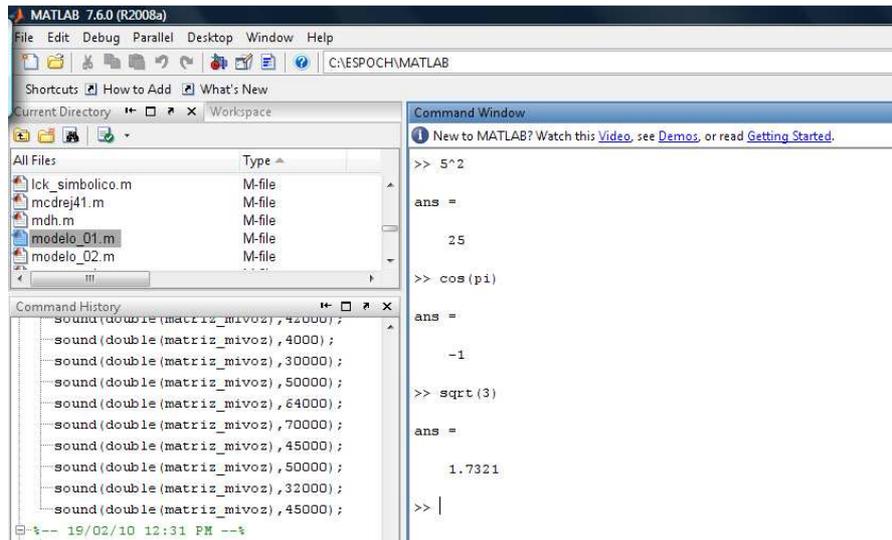


Figura III.03. Similitud a una Calculadora Científica.

Cuando se realizan todas estas operaciones desde la ventana de comandos de MATLAB, en el área de trabajo WORKSPACE, se crean las variables o resultados obtenidos los cuales se los puede borrar usando los mandatos CLC que borra la pantalla y CLEAR que borra estos valores de variables generados.

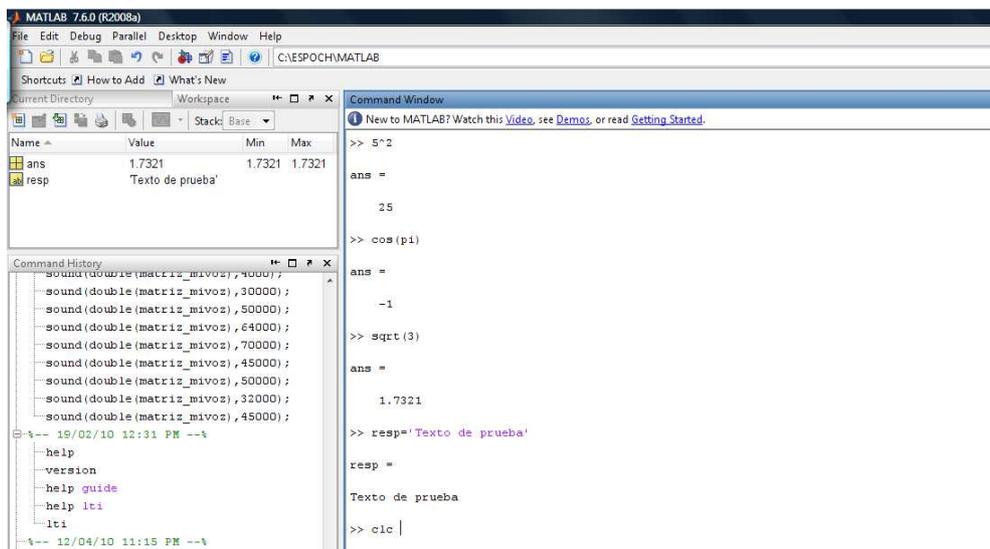


Figura III.04. Variable tipo texto, uso de CLC

En la figura III.04, hemos creado la variable **resp** que contiene la cadena 'Texto de prueba', en la ventana o área de trabajo se aprecia esta variable y la variable **ans** con el último contenido proporcionado por la función **SQRT(3)**, raíz cuadrada de tres.

Al escribir **CLC** en el indicador de la ventana de mandatos simplemente se borra la ventana, pero no los datos del área de trabajo, al escribir **CLEAR**, estos se borran de forma permanente.

Esto debe tomarse muy en cuenta cuando se desea escribir un archivo-M, ya que estos dos mandatos seguidos de un punto y coma [;¹] permitirán inicializar el entorno a operar. Esto se puede apreciar en la figura III.05, la escritura de un guión con los comandos ingresados anteriormente.

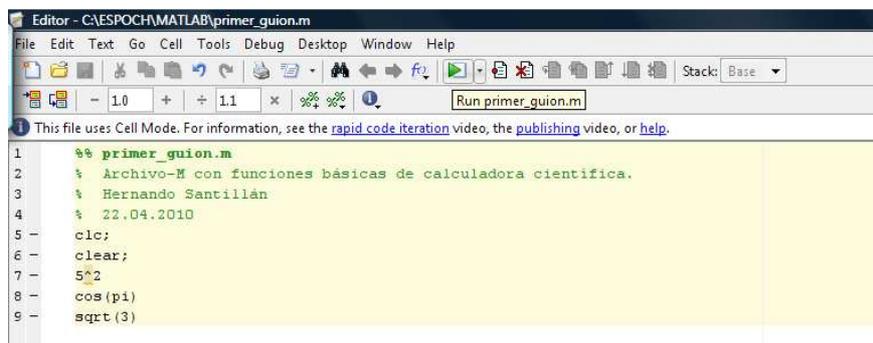


Figura III.05. Escritura primer archivo-M

Los pasos para crear este archivo-M son muy intuitivos **File/New/M-File** (Archivo/Nuevo/archivo-M), recuerde que se puede cambiar la carpeta de trabajo dando clic en los tres puntos correspondientes a Buscar nueva carpeta, como se aprecia en la figura III.06.

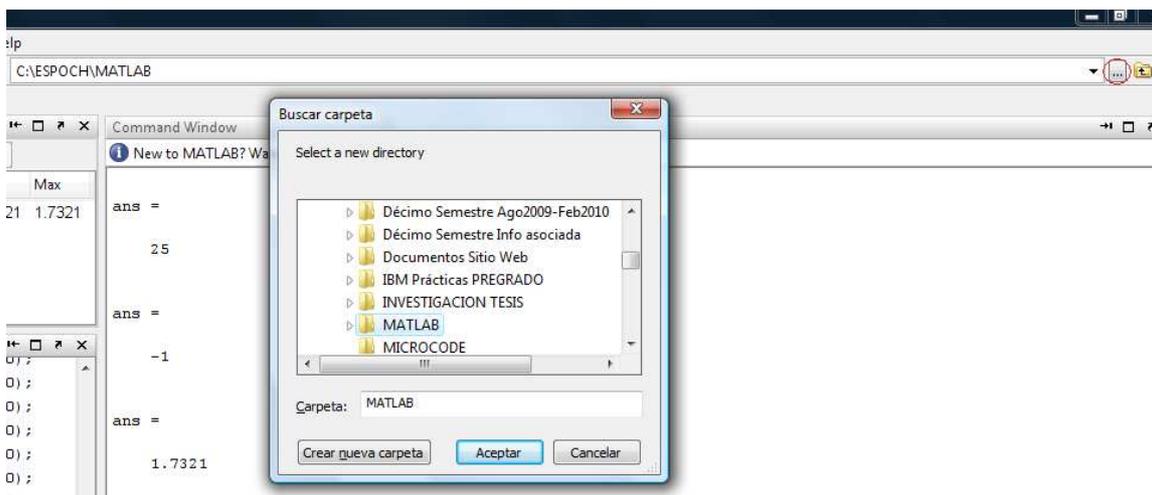


Figura III.06. Búsqueda de otra carpeta de trabajo para MATLAB.

¹ Permite que MATLAB no muestre el resultado en la ventana de comandos.

MATLAB es pues un laboratorio de matrices, por tanto todos sus datos están almacenados bajo esta premisa, a continuación dos ejemplos de cómo podemos crear un arreglo unidimensional y una matriz de $n \times m$ elementos desde la ventana de comandos.

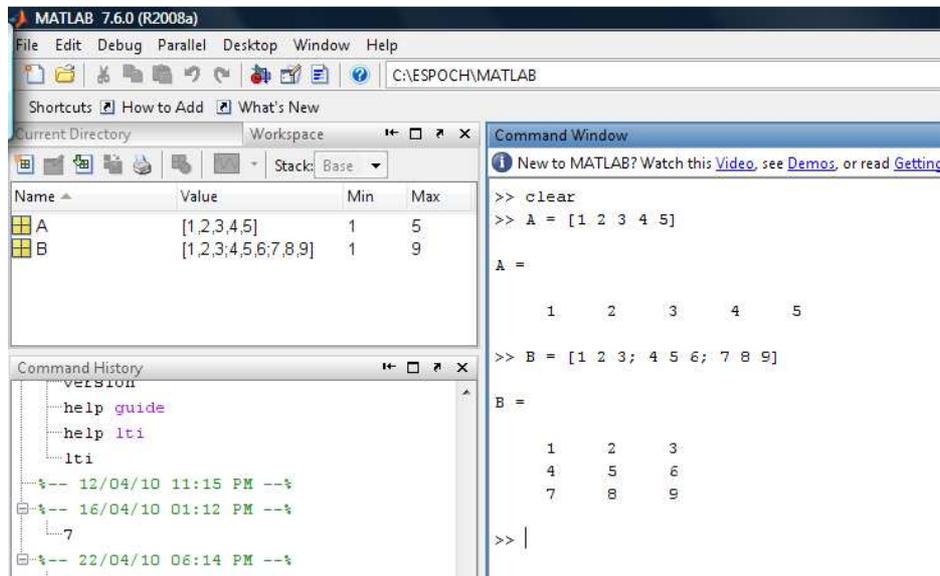


Figura III.07. Un vector y una matriz.

La figura III.07, nos muestra cómo trabaja MATLAB con los vectores o arreglos unidimensionales y las matrices de $n \times m$ elementos, en nuestro ejemplo el vector A, tiene 5 elementos siempre encerrados entre corchetes, mientras que nuestra matriz B es de 3×3 de 9 elementos, cada fila se separa mediante un punto y coma [;].

De una forma parecida se puede crear un vector de 100 elementos para poder simular o estimar un tiempo t de 100 segundos [0..99], de la forma que muestra la figura III.08.

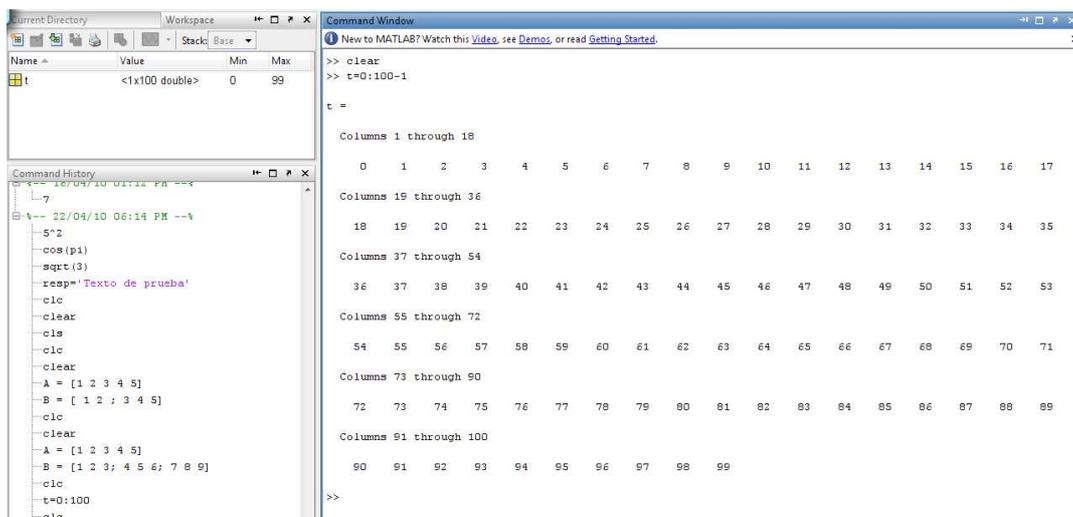


Figura III.08. Creación vector tiempo t, para simulación de señales.

Se nota que simplemente al escribir $t=0:100-1$ se tienen 100 elementos desde el 0 hasta el 99, es el motivo de que se debe restar 100-1, para que este resultado no se muestre en la ventana de comandos recuerde de usar el punto y coma al final de esta sentencia. Para luego usar la función PLOT, para graficar cualquier señal básica.

```
>> clear  
  
>> t=0:100-1;  
  
>> fm=2000; %Frecuencia de muestreo  
  
>> A=2; %Amplitud 2V  
  
>> w0=60; %Frecuencia 60Hz  
  
>> v=A*sin(2*pi*w0*t/fm); %Función Seno  
  
>> T=100; %Período  
  
>> plot(-T/2:T/2-1,v); %Gráfico función2.  
  
>>
```

Se tiene entonces como resultado el siguiente gráfico correspondiente a la señal seno. Figura III.09.

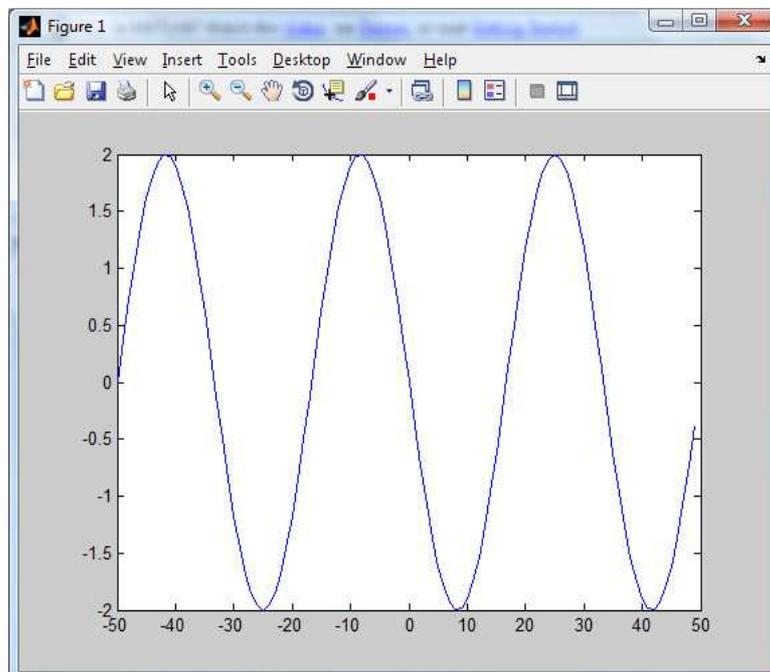


Figura III.09. Función señal Seno sencilla.

² Todos los vectores deben tener la misma cantidad de elementos o longitud.

Se recomienda escribir y correr los archivos M, **segundo_archivo.m** y **tercer_archivo.m** que se listan a continuación, como un complemento del análisis de señales básicas sinusoidales (seno y coseno).

```
Editor - C:\ESPOCH\MATLAB\segundo_archivo.m
File Edit Text Go Cell Tools Debug Desktop Window Help
- 1.0 + 1.1 x
This file uses Cell Mode. For information, see the rapid code iteration video, the publishing video, or help.
1 %% segundo_archivo.m
2 % Hernando Santillán.
3 % 23.04.2010 0942Hrs
4 -
5 - clear
6 - T=100; %Periodo
7 - t=0:T-1; %tiempo muestreo
8 - A=2; %Amplitud 2V
9 - f=60; %frecuencia operación 60Hz.
10 - fm=2e3; %frecuencia de muestreo.
11 - v1=A*sin(2*pi*f*t/fm); %Función seno.
12 - v2=A*cos(2*pi*f*t/fm); %Función coseno.
13 - subplot(2,1,1), plot(-T/2:T/2-1,v1,'r'), grid;
14 - hold;
15 - subplot(2,1,1), plot(-T/2:T/2-1,v2,'g');
16 - legend('Señal Seno','Señal Coseno');
17 - title('Señales Seno y Coseno a 60Hz');
```

Figura III.10. Listado: SEGUNDO_ARCHIVO.M

Que nos genera el siguiente gráfico de estas señales seno y coseno básicas muestreadas en una frecuencia de 60Hz, para efectos demostrativos así lo muestra la figura III.11.

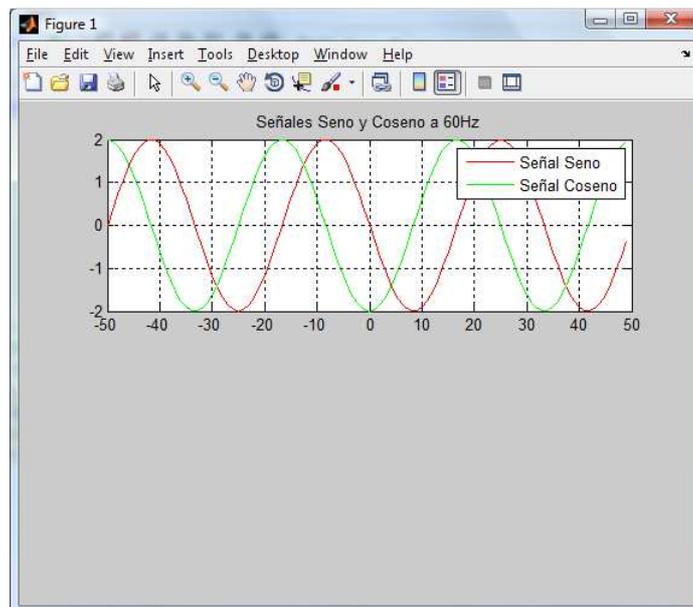
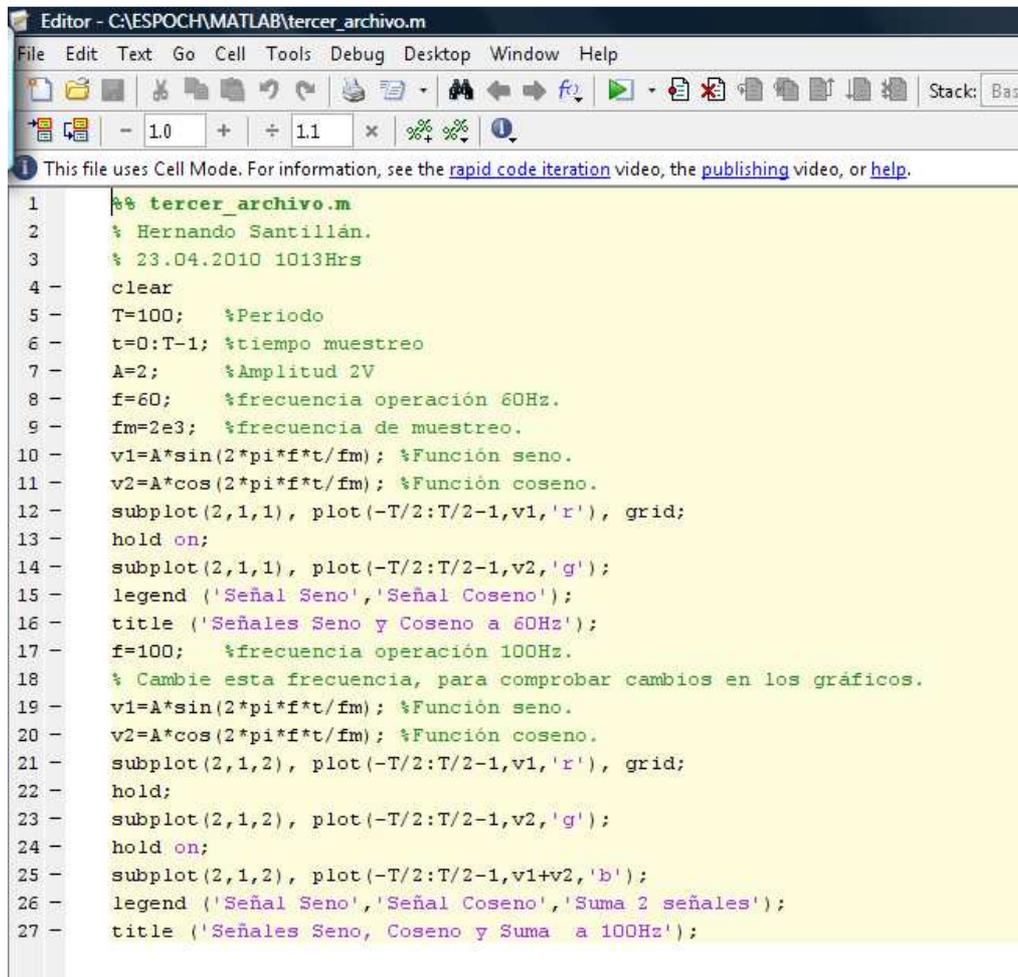


Figura III.11. Señales Seno y Coseno

A continuación se puede dibujar la suma de estas dos señales, por lo que nos referimos al listado del guión TERCER_ARCHIVO.M



```
1 %% tercer_archivo.m
2 % Hernando Santillán.
3 % 23.04.2010 1013Hrs
4 clear
5 T=100; %Periodo
6 t=0:T-1; %tiempo muestreo
7 A=2; %Amplitud 2V
8 f=60; %frecuencia operación 60Hz.
9 fm=2e3; %frecuencia de muestreo.
10 v1=A*sin(2*pi*f*t/fm); %Función seno.
11 v2=A*cos(2*pi*f*t/fm); %Función coseno.
12 subplot(2,1,1), plot(-T/2:T/2-1,v1,'r'), grid;
13 hold on;
14 subplot(2,1,1), plot(-T/2:T/2-1,v2,'g');
15 legend('Señal Seno','Señal Coseno');
16 title('Señales Seno y Coseno a 60Hz');
17 f=100; %frecuencia operación 100Hz.
18 % Cambie esta frecuencia, para comprobar cambios en los gráficos.
19 v1=A*sin(2*pi*f*t/fm); %Función seno.
20 v2=A*cos(2*pi*f*t/fm); %Función coseno.
21 subplot(2,1,2), plot(-T/2:T/2-1,v1,'r'), grid;
22 hold;
23 subplot(2,1,2), plot(-T/2:T/2-1,v2,'g');
24 hold on;
25 subplot(2,1,2), plot(-T/2:T/2-1,v1+v2,'b');
26 legend('Señal Seno','Señal Coseno','Suma 2 señales');
27 title('Señales Seno, Coseno y Suma a 100Hz');
```

Figura III.12. Listado: TERCER_ARCHIVO.M

La gráfica correspondiente a este guión la tenemos representada en la figura III.14, aquí se puede apreciar cómo se retienen los gráficos a través de la sentencia HOLD ON, además de que ya hemos separado dos áreas para mostrar los gráficos usando el mandato SUBPLOT, para mejor información del uso de cada uno de los comandos de MATLAB, se puede acceder a HELP, de la siguiente forma:

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

>> help subplot
SUBPLOT Create axes in tiled positions.
H = SUBPLOT(m,n,p), or SUBPLOT(mnp), breaks the Figure window
into an m-by-n matrix of small axes, selects the p-th axes for
the current plot, and returns the axis handle. The axes are
counted along the top row of the Figure window, then the second
row, etc. For example,

SUBPLOT(2,1,1), PLOT(income)
SUBPLOT(2,1,2), PLOT(outgo)
```

Figura III.13. Ayuda en MATLAB.

Se aprecia que luego de escribir HELP mandato, MATLAB da una extensa información en inglés con ejemplos detallados acerca de la función o mandato requeridos.

A continuación, la salida del guión TERCER_ARCHIVO.M

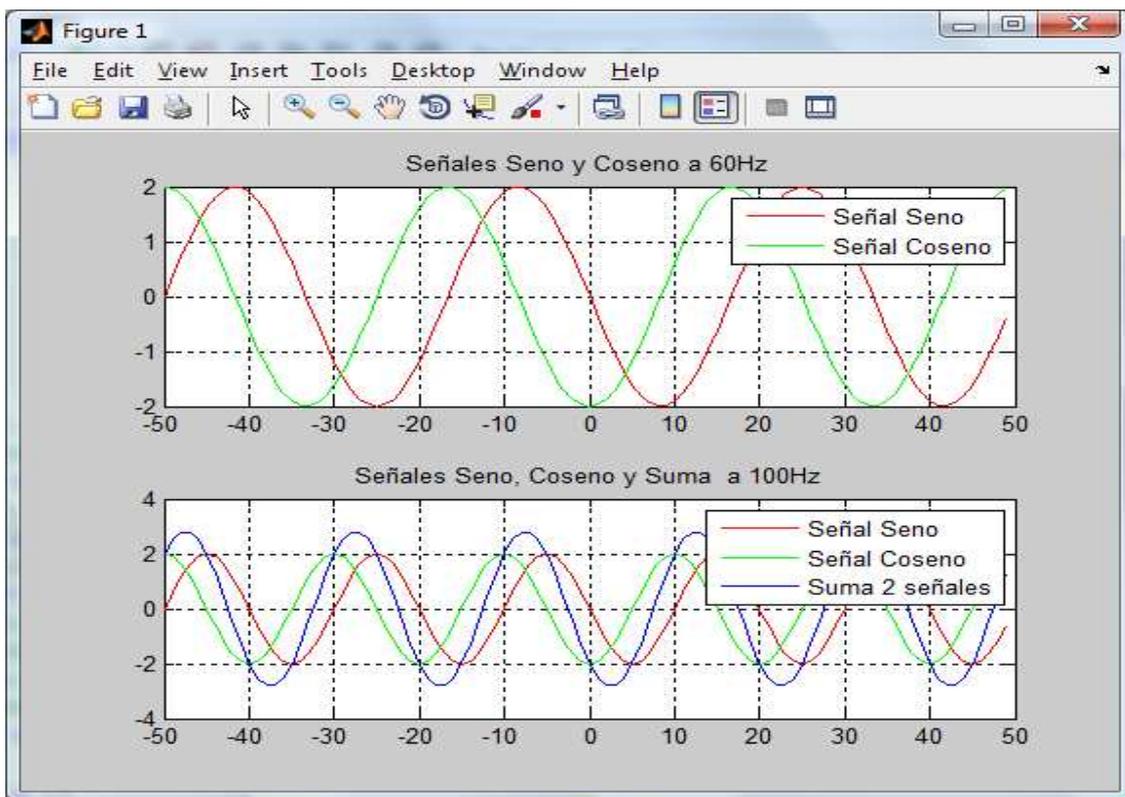


Figura III.14. Salida TERCER_ARCHIVO.M

El segundo y tercer archivo contienen como se nota el uso de nuevos comandos para mostrar los gráficos ya representados en las figuras III.11 y III.14, pero debemos explicar cuál es la forma de su uso.

El mandato SUBPLOT, divide la ventana de gráficos en nuestros ejemplos en dos partes, luego de entender la explicación de la ayuda SUBPLOT(m,n,p), nos interesa saber que el valor p se refiere a la ventana 1 y 2 respectivamente; m y n son los puntos o coordenadas iniciales.

HOLD ON, me permite guardar o copiar el gráfico para que pueda ser contrastado con el siguiente, LEGEND, en función del número de señales se lo coloca el final con el objetivo de esclarecer cada una de nuestras salidas. Y finalmente en el mandato PLOT se añadió el color de las señales, tomando en cuenta que 'r' equivale a rojo (red en inglés), 'g' a verde (green en inglés) y 'b' equivale a azul (de blue del inglés). GRID, me permite mostrar la figura dividida en una grilla, y por supuesto TITLE me permite dar un título a mi gráfico correspondiente. Al mirar detalladamente estos gráficos, se nota que el período de simulación de estas ondas está dado luego de dividir en dos partes nuestro eje x, en la mitad negativa y la mitad positiva de T, cuando se use SIMULINK estas demostraciones serán más intuitivas, ya que recuerde que esta herramienta está orientada a una mejor simulación de los eventos de análisis de señales, motivo de nuestro estudio. Es importante recalcar que SIMULINK sirve además para muchas más aplicaciones en el área científica y de ingeniería.

Primeros pasos en el uso del programa SIMULINK de MATLAB

Objetivo: Modelar y simular dos señales básicas de una onda Seno y Coseno mediante un osciloscopio. Ondas sinusoidales básicas para el estudio de señales y la modelación posterior en una serie trigonométrica de Fourier.

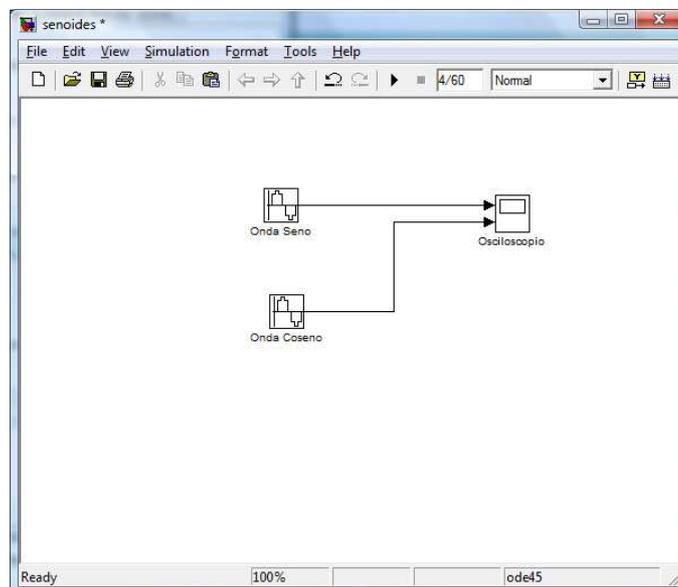


Figura III.15. Señales Básicas

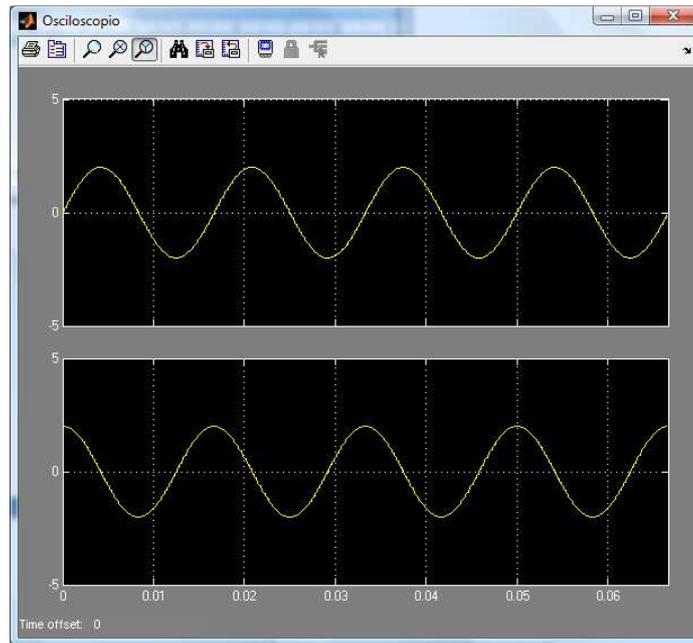


Figura III.16. Salida bloque SCOPE a) Onda Seno b) Onda Coseno

Para esta salida se ha tenido que tomar en cuenta los siguientes parámetros configurados en el bloque Seno:

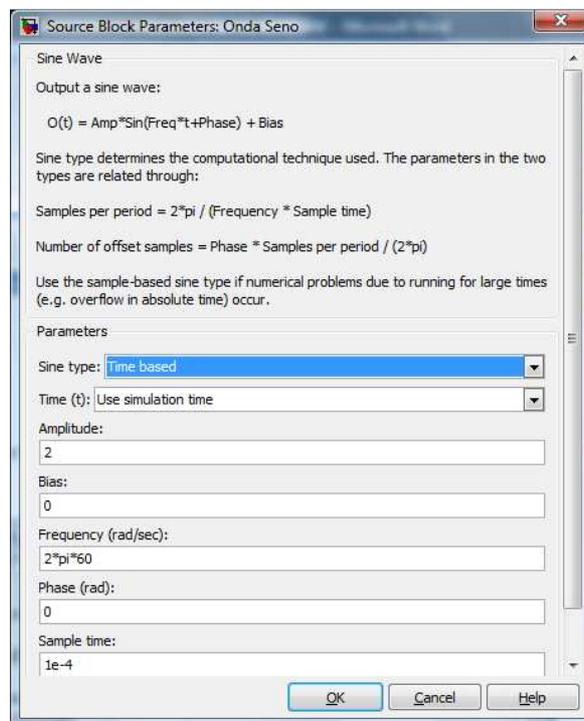


Figura III.17. Configuración del bloque DE LA FUNCIÓN Seno(t)

En primer parámetro a tomar en cuenta es la amplitud de la onda sinusoidal, que aleatoriamente la configuramos en 2V (por omisión se tiene 1). El parámetro inclinación (Bias) se lo mantiene en su valor por omisión 0. La frecuencia es el parámetro $\omega = 2\pi f$, por tanto si $f = 60\text{Hz}$ entonces es por este motivo que en el campo Frecuencia (frequency (rad/sec)) escribimos $2 * \pi * 60$. En el siguiente casillero por omisión la fase es 0 radianes correspondiente a la función seno (t), y el tiempo de muestreo es de **1e-4 s o sea 0.1ms** que es un tiempo apropiado para la muestra. Para la onda coseno sabemos que está desfasada en 180 grados o $\pi/2$ radianes, por tanto, al configurarla tenemos:

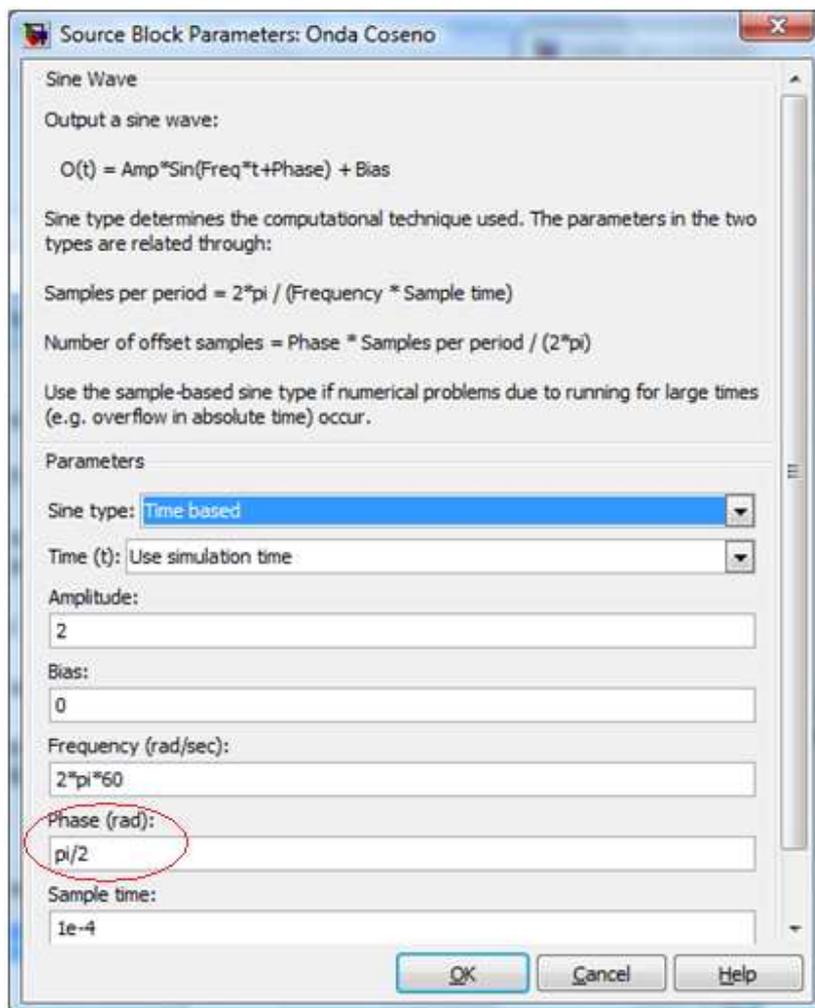


Figura III.18. Fase $\pi/2$ de una onda Seno transformada en Coseno (t)

Ahora para conseguir estos bloques, simplemente al lanzar el programa SIMULINK desde el indicador de la ventana de mandatos de MATLAB, escribiendo:

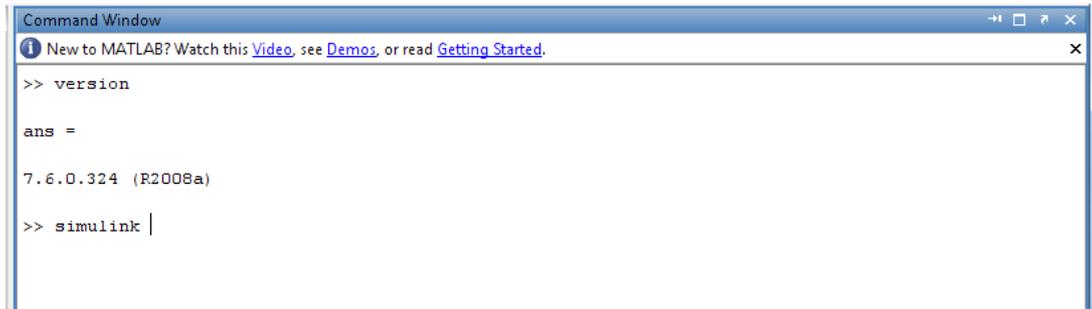


Figura III.19. Inicio de Simulink.

SIMULINK es el programa que nos permite modelar en tiempo real varios sistemas, inicialmente estamos revisando la parte básica de las ondas sinusoidales o senoides, elementales para el estudio de señales en comunicaciones de datos, cuando hablamos de las Series de Fourier.

Escogemos de la biblioteca **Sources** (fuentes) en donde hallamos el bloque Seno (Sine wave), de ahí sólo hay que arrastrar este bloque al área de diseño.

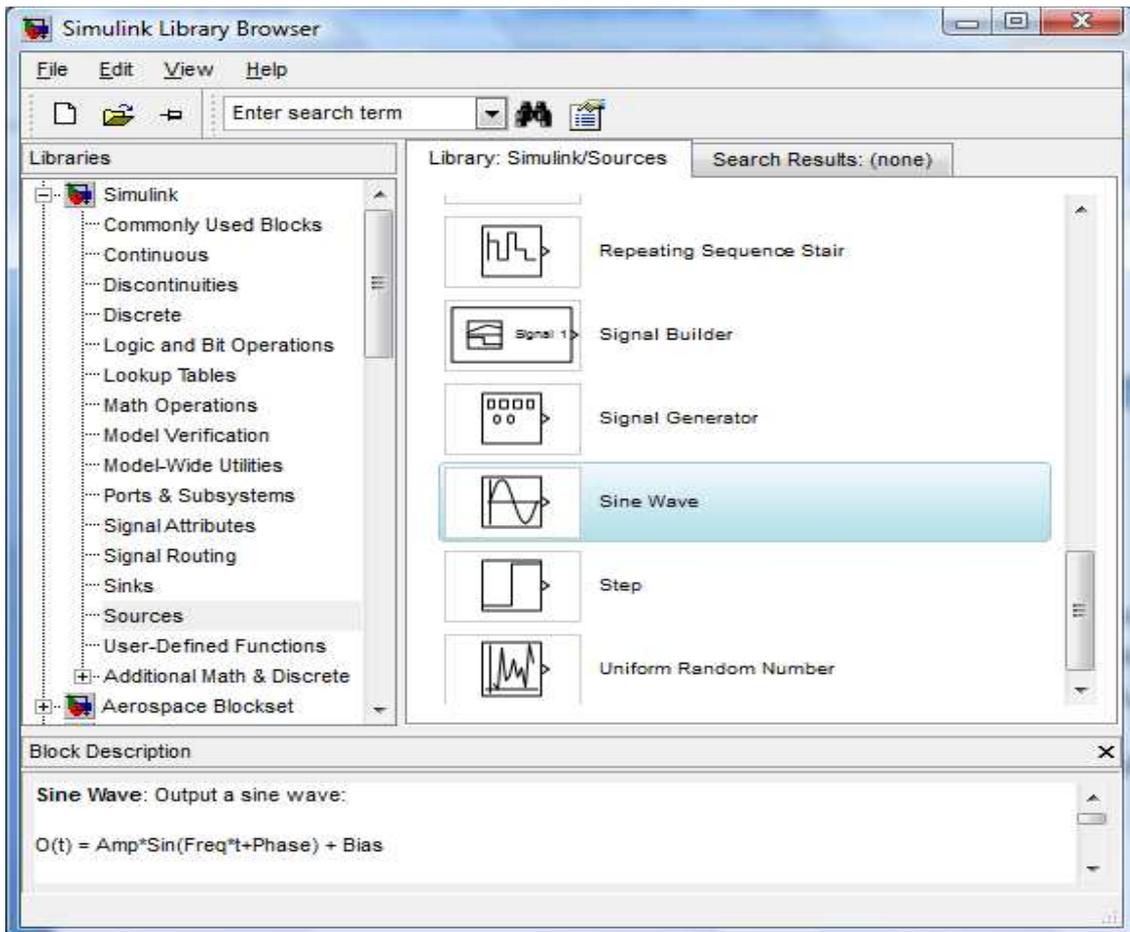


Figura III.20. Biblioteca Sources/Sine Wave (Fuentes/Onda Seno)

El osciloscopio (**Scope**) se lo consigue de la biblioteca **Commonly Used Blocks** (bloques comúnmente usados).

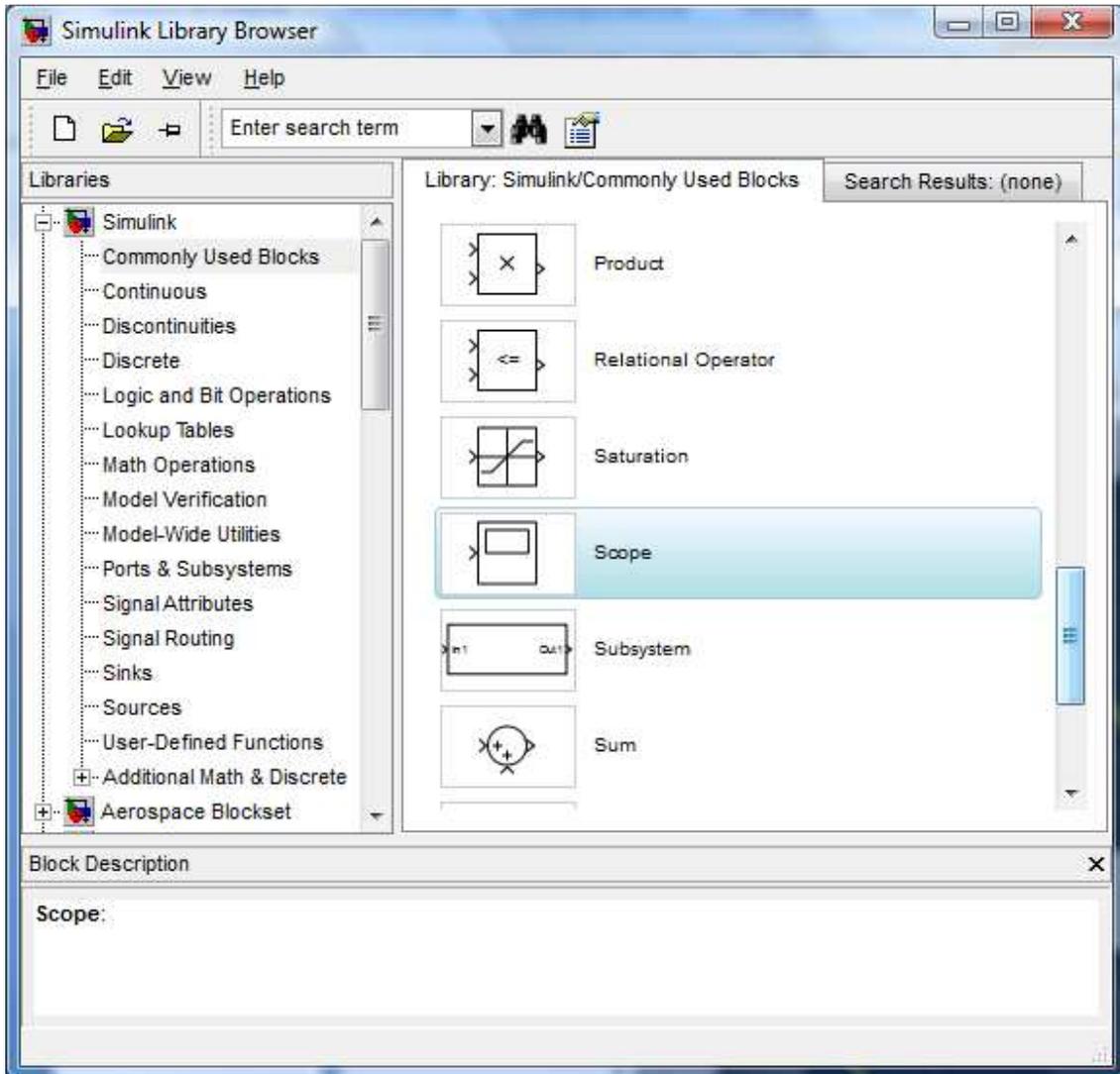


Figura III.21. Biblioteca Commonly Used Blocks/Scope (Bloques Comúnmente usados/Osciloscopio)

Hemos ido en este ejemplo, partiendo de nuestro objetivo final hasta como se forma nuestro primer ejemplo de Simulación de las ondas básicas en el tratamiento de señales, las ondas sinusoidales Seno y Coseno.

Resumiendo, lance el programa SIMULINK desde la ventana de mandatos de MATLAB, clic en el ícono de **Nuevo modelo** (New model), por omisión tendríamos:

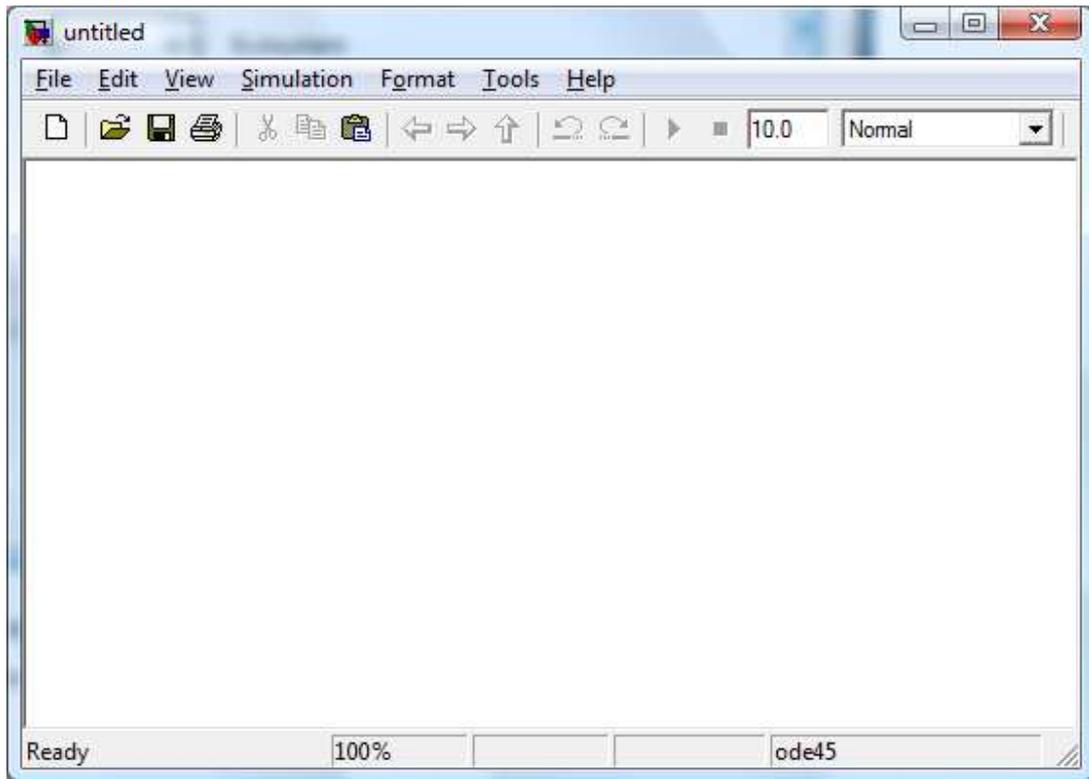


Figura III.22. Ventana Nuevo modelo.

Luego añadamos los 2 bloques Seno y el bloque Osciloscopio de las respectivas bibliotecas mencionadas.

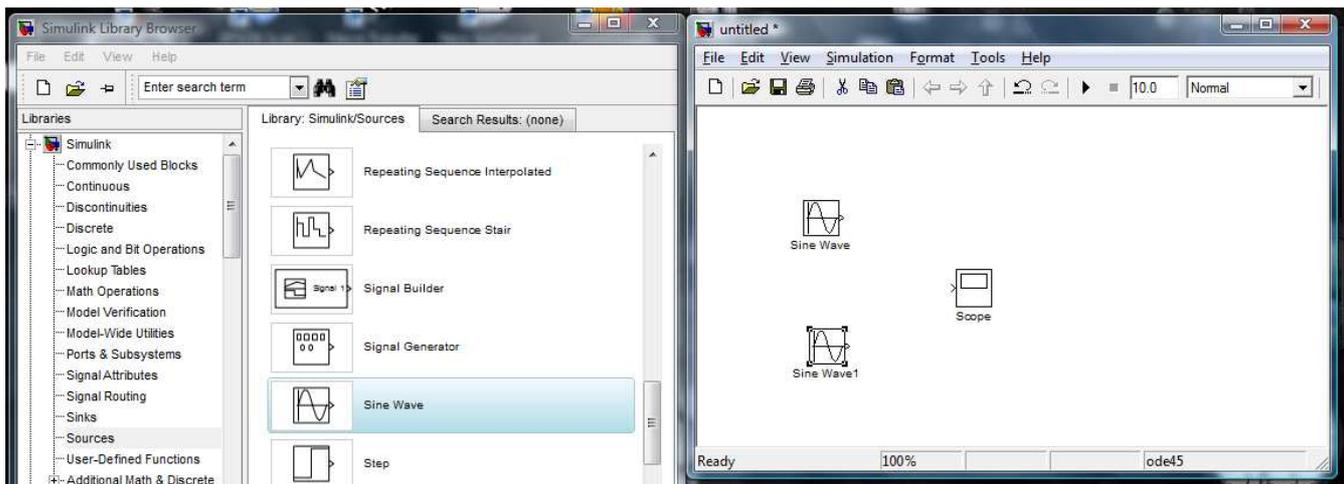


Figura III.23. Implementación de nuestra simulación.

Se puede cambiar los nombres de estos bloques para mejor comprensión, es decir Sine Wave por Onda Seno, Sine Wave1 por Onda Coseno y finalmente Scope por Osciloscopio. Luego en los bloques Onda Seno y Coseno que tienen estos valores por omisión:

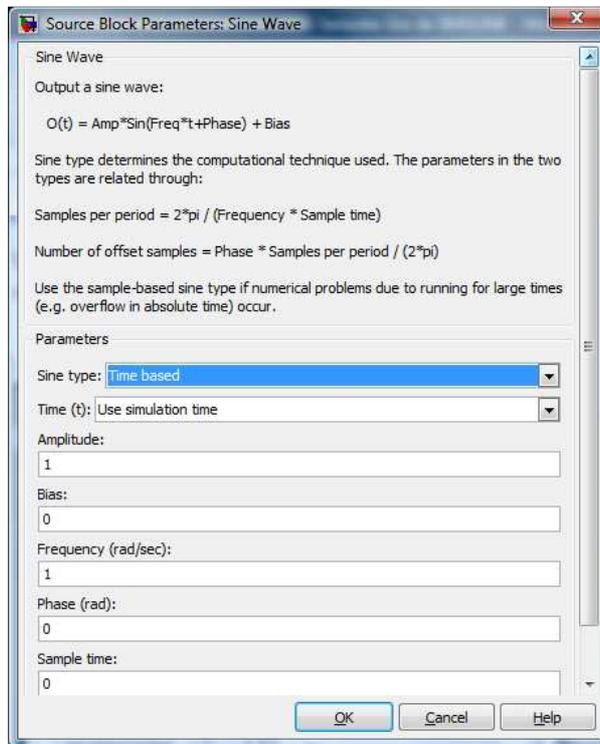


Figura III.24. Valores por omisión onda seno (Sine Wave) de la biblioteca Fuentes (Sources).

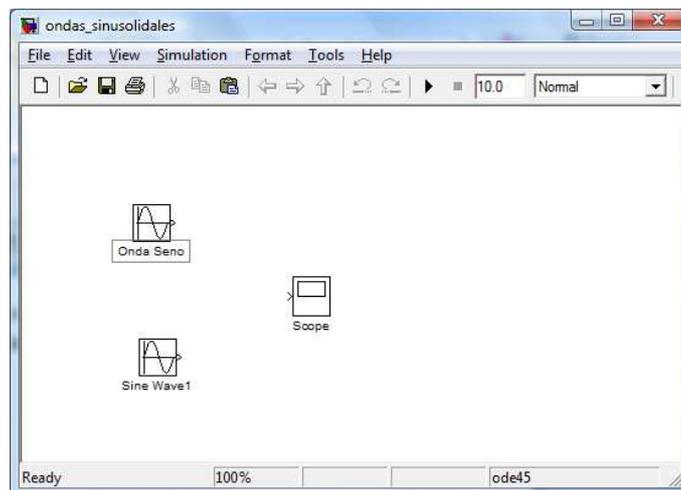


Figura III.25. Cambio de nombre³ al bloque Sine Wave por Onda Seno

³ Clic sobre el nombre Sine Wave1 para cambiar el nombre a Onda Coseno.

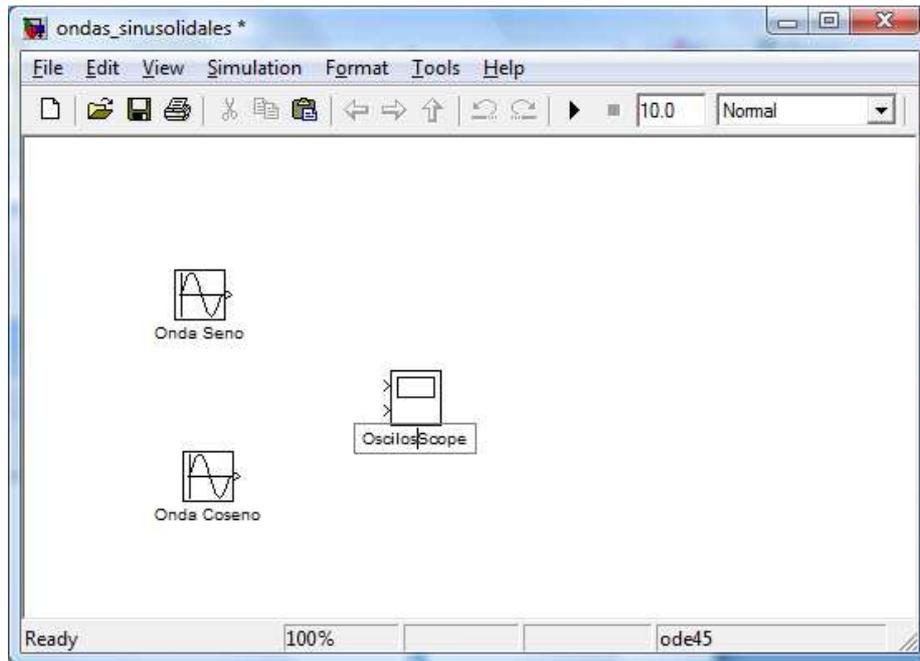


Figura III.26. Cambio de nombre del bloque Scope a Osciloscopio.

Para hacer que nuestro bloque Scope cambiado el nombre a Osciloscopio pueda tener más de una salida, damos doble clic en el bloque Osciloscopio y clic en el ícono correspondiente a Parámetros, allí cambiamos el número de cortes de pantalla a **2**, y poder de esta forma evaluar las dos señales al mismo tiempo.

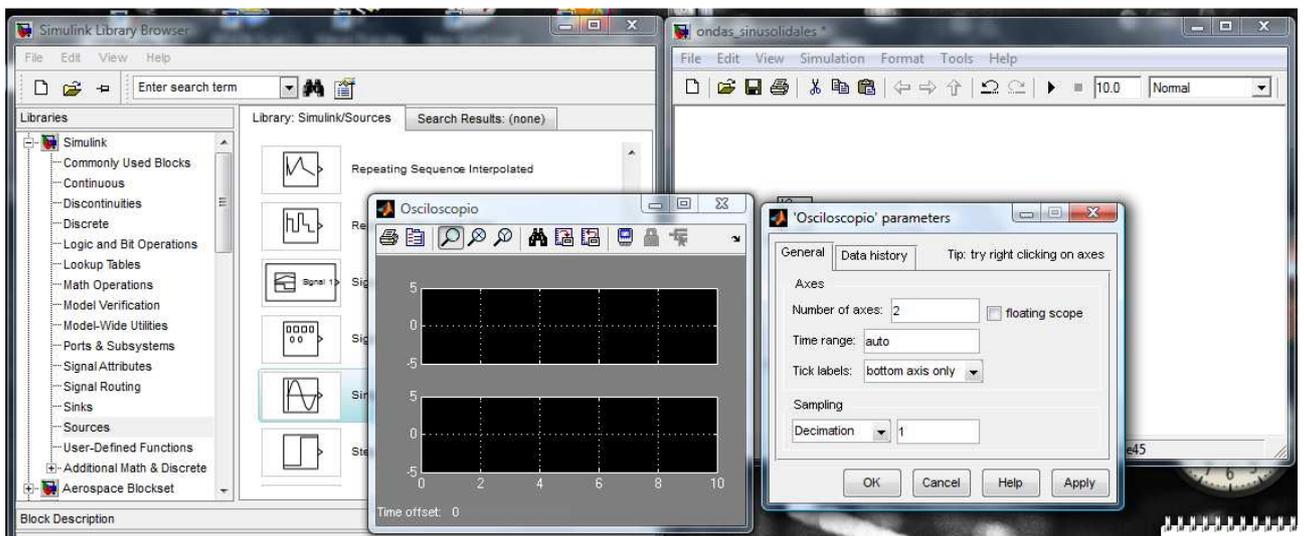


Figura III.27. Corte de pantalla a 2 salidas

Finalmente los bloques hay que unirlos mediante flechas desde cada uno de ellos: Onda Seno y Onda Coseno hasta el bloque Osciloscopio.

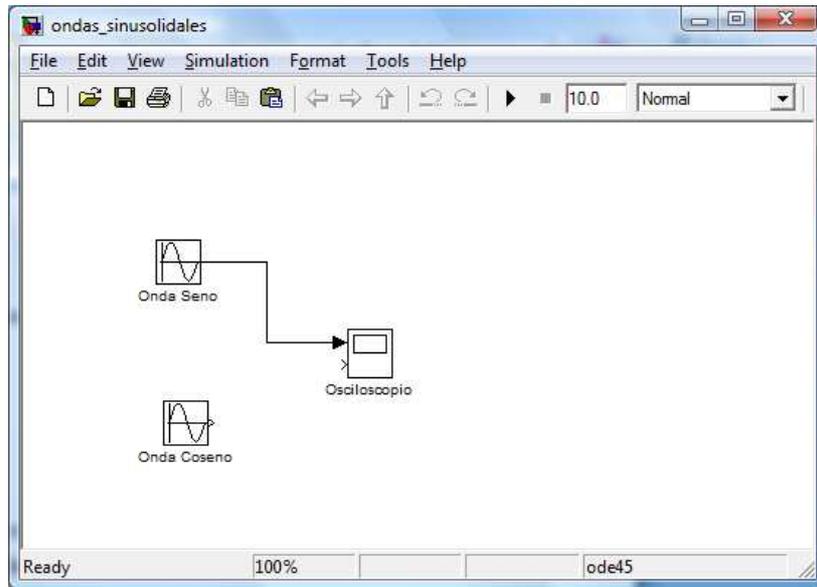


Figura III.28. Conexión de señales

Ya creado el modelo, procedemos a configurar la salida del osciloscopio, dando clic derecho en Autoescala.

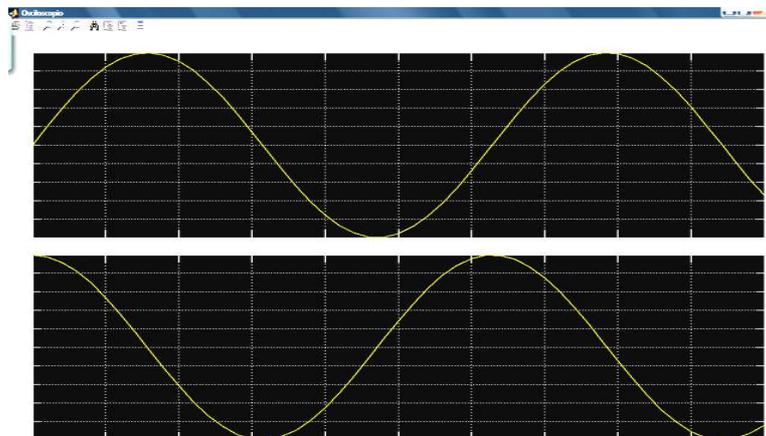


Figura III.29. Autoescala de la salida de la Simulación

Recordemos que nuestro modelo final lo tenemos en el CD adjunto a esta tesis con el nombre de **ONDAS_SINUSOIDALES.MDL**, la extensión .MDL corresponde a los archivos de SIMULINK.

Para recuperar este modelo en una fecha posterior, recuerde primero lanzar el programa SIMULINK como ya se mencionó simplemente escribiendo **simulink** en la línea de mandatos, y posterior a esto dar clic en el ícono de carpeta Abrir Modelo, escoger este archivo y dar clic en el botón Abrir.

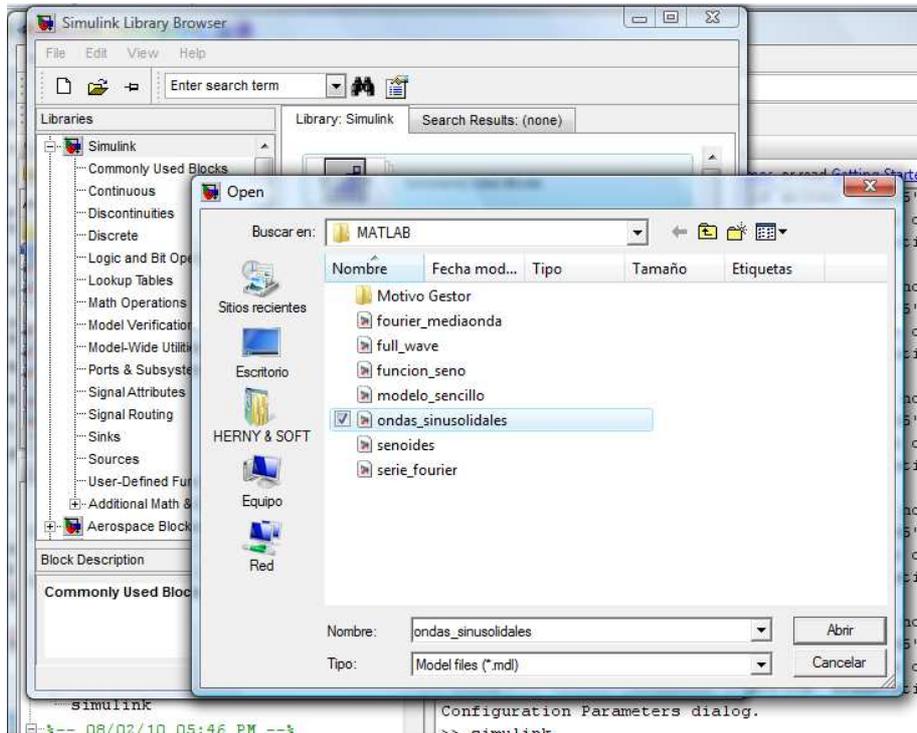


Figura III.30. Abrir modelo ya creado.

Como muestra la figura III.30, se ha creado una carpeta denominada MATLAB, que contiene todos los archivos que se explican en este trabajo de tesis.

3.2 PRÁCTICA ONDAS DETERMINÍSTICAS Y ALEATORIAS.

OBJETIVOS

Usar MATLAB como interface entre el estudiante de Análisis de Señales y el marco teórico planteado en el tema de las señales determinísticas y aleatorias junto al cálculo de la fórmula de capacidad del canal C de Claude Shannon, a través de una simulación básica de intercambio de información desde un sistema transmisor a un sistema receptor, añadido ruido blanco o gaussiano.

MARCO TEÓRICO

Formas de onda determinísticas y aleatorias.

Dos clases generales de formas de onda son de importancia en los sistemas de comunicación: determinísticas y aleatorias (o estocásticas).

DEFINICIÓN: Una forma de onda determinística puede modelarse como una función de tiempo completamente especificada.

Por ejemplo, si:

$$\omega(t) = A \cos(\omega_0 t + \varphi_0)$$

Describe una forma de onda, donde A, ω_0 , y φ_0 son constantes conocidas, se dice que esta forma de onda es determinística debido a que para cada valor de t, el valor de $\omega(t)$ puede evaluarse. Si cualquiera de las constantes es desconocida, entonces el valor de $\omega(t)$ no puede calcularse y, por consecuencia, $\omega(t)$ no es determinística.

DEFINICIÓN: Una forma de onda aleatoria (o estocástica) no se puede especificar completamente como una función de tiempo y debe modelarse estadísticamente.

Esto presenta inmediatamente un dilema al analizar sistemas de comunicación. Las formas de onda que representan a la fuente no pueden ser determinísticas. Por ejemplo, en un sistema de comunicación digital, puede enviarse información correspondiente a cualesquier letra del alfabeto español. Cada letra puede representarse mediante una forma de onda determinística, pero cuando se examine la forma de onda emitida por la fuente se encuentra que es aleatoria, ya que no se sabe exactamente que caracteres se transmitirán. Por consiguiente, se necesita diseñar el sistema de comunicación utilizando una forma de onda de señal aleatoria. El ruido también puede ser descrito por una forma de onda aleatoria. Esto requiere el uso de de conceptos de probabilidad y estadística que complican el procedimiento de análisis y diseño. Sin embargo, si

representamos la forma de onda de la señal mediante una forma de onda determinística “típica” se puede obtener la mayoría de los resultados esperados, aunque no todos. (COUCH, 2008)

DESARROLLO

Abra el entorno de SIMULINK y vaya a la biblioteca Conjunto de Bloques referente a Comunicaciones/Modulación/Modulación de Banda base digital/Frecuencia Modulada⁴ (**Communications Blockset/Modulation/Digital Baseband Modulation/FM**), esta señal FM digital es una señal determinística, que como se aprecia en la figura 1.1.1 se tienen bloques para su modulación y demodulación, es decir sus correspondientes sistemas de transmisión (TX) y recepción (RX).

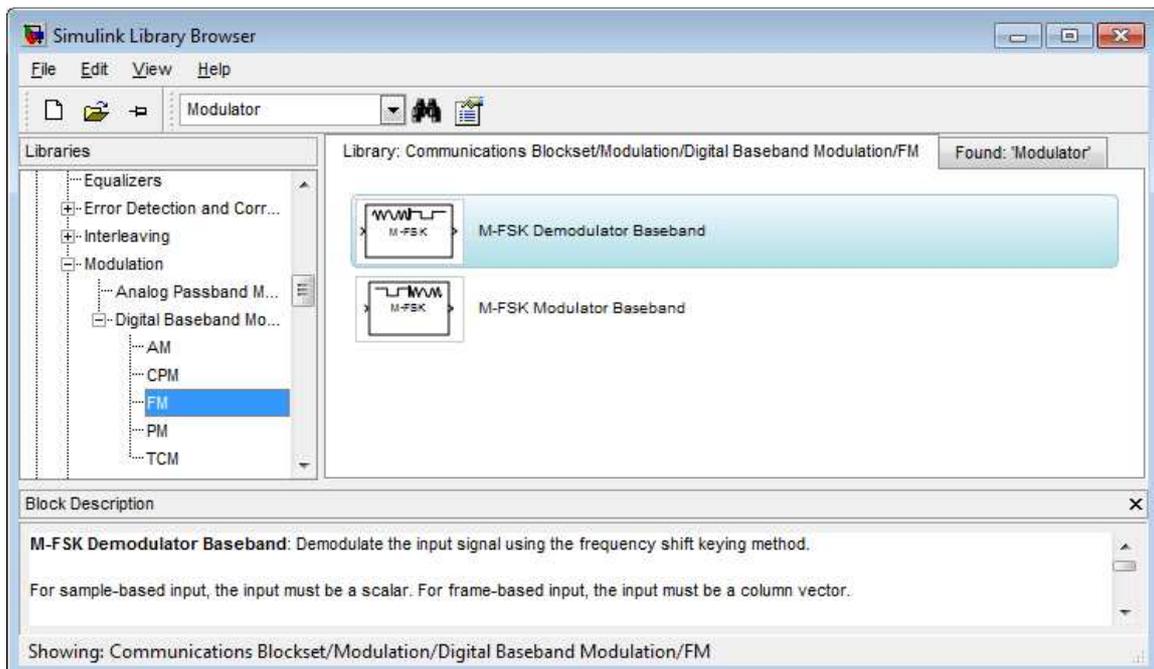


Figura III.31. Modulador/Demodulador Digital por Desplazamiento de Frecuencia FSK (Frequency Shift Keying).

Necesitaremos además una fuente generadora de pulsos aleatoria, que lo conseguimos a través del bloque ubicado en la misma biblioteca de Comunicaciones: Fuentes de comunicaciones/fuentes de datos aleatorios. Que se muestra en la figura III.32.

Communications Blockset/Comm sources/Random Data Sources

⁴ Digital.

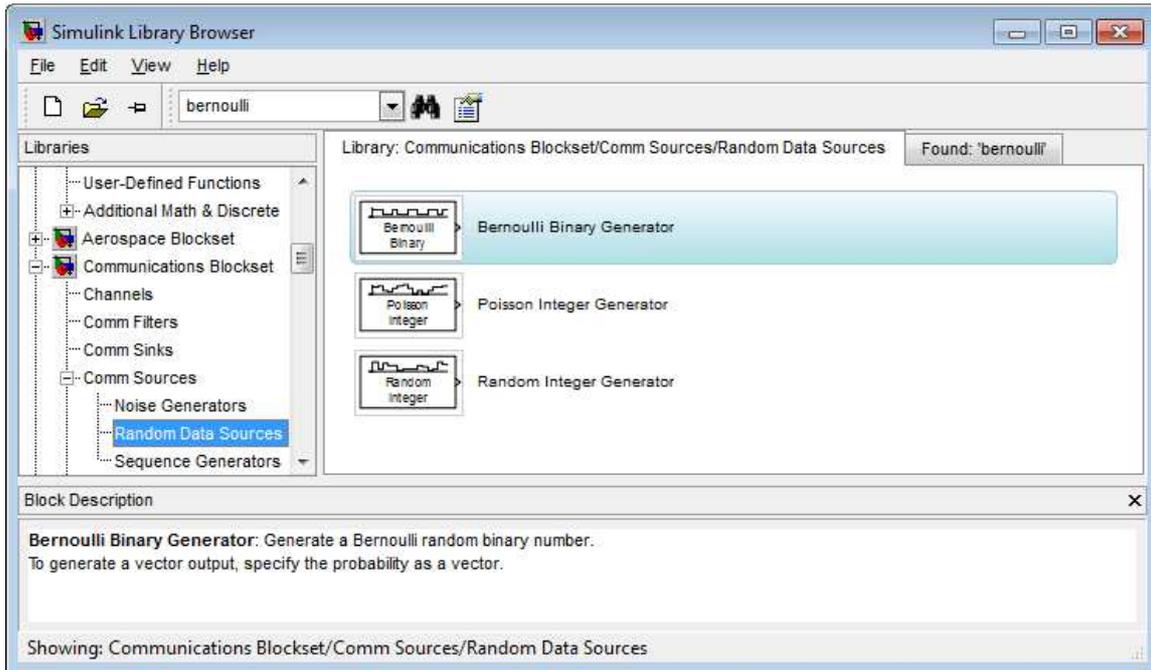


Figura III.32. Bloque Generador Binario Aleatorio de Bernoulli.

Adicionalmente a esta simulación añadimos un bloque que nos representa una señal aleatoria (o estocástica), como es un canal de ruido blanco gaussiano, que se ubica también en la biblioteca de Comunicaciones (Communication Blockset)/Canales (Channels), que se muestra en la figura III.33. **Communication Blockset/Channels**

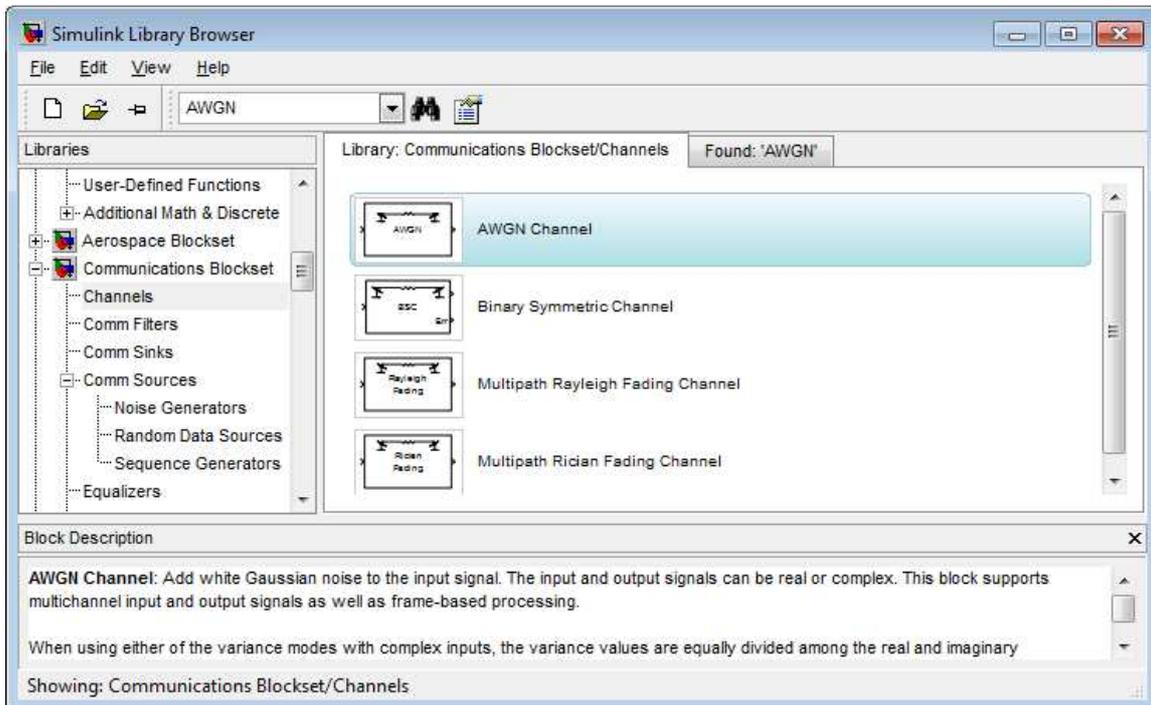


Figura III.33. Bloque que Añade Ruido Blanco Gaussiano a la Simulación.

Para poder comparar estas dos señales simuladas, utilizamos un osciloscopio, que se lo consigue en la biblioteca **Simulink/Commonly Used Blocks/Scope** (Bloques comúnmente usados/Osciloscopio). Figura III.34.

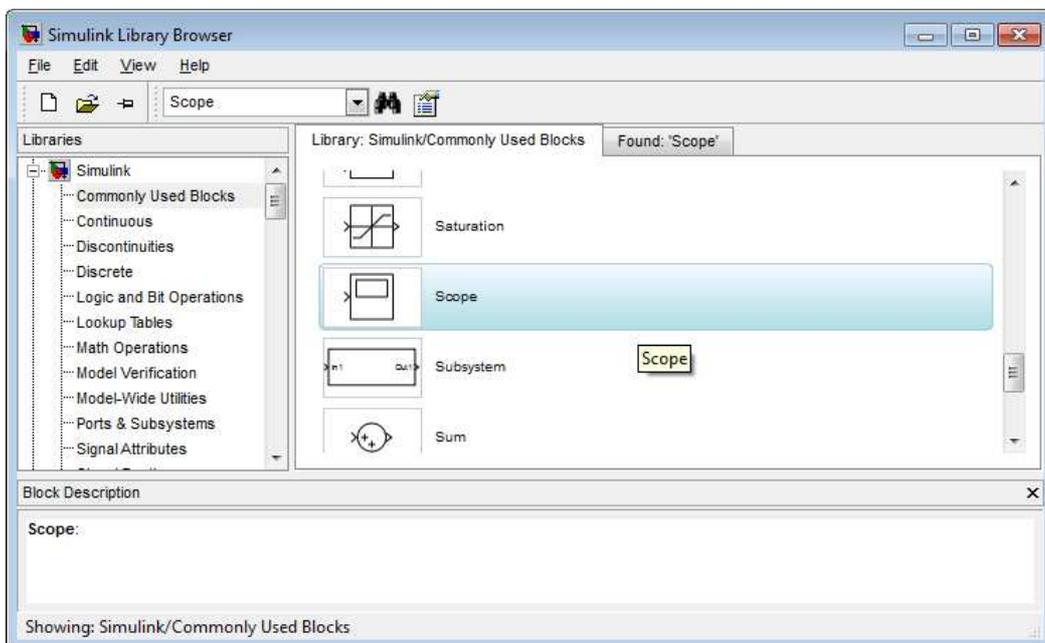


Figura III.34. Bloque Scope (Osciloscopio).

Como ya practicó en el capítulo de Introducción, el paso siguiente es arrastrar y soltar estos bloques, después de dar clic en New (nuevo modelo) para su posterior simulación, entonces nuestro modelo quedaría: (figura III.35)

Recuerde que además en el CD adjunto a esta tesis se encuentra este modelo ya terminado para su respectiva comparación.

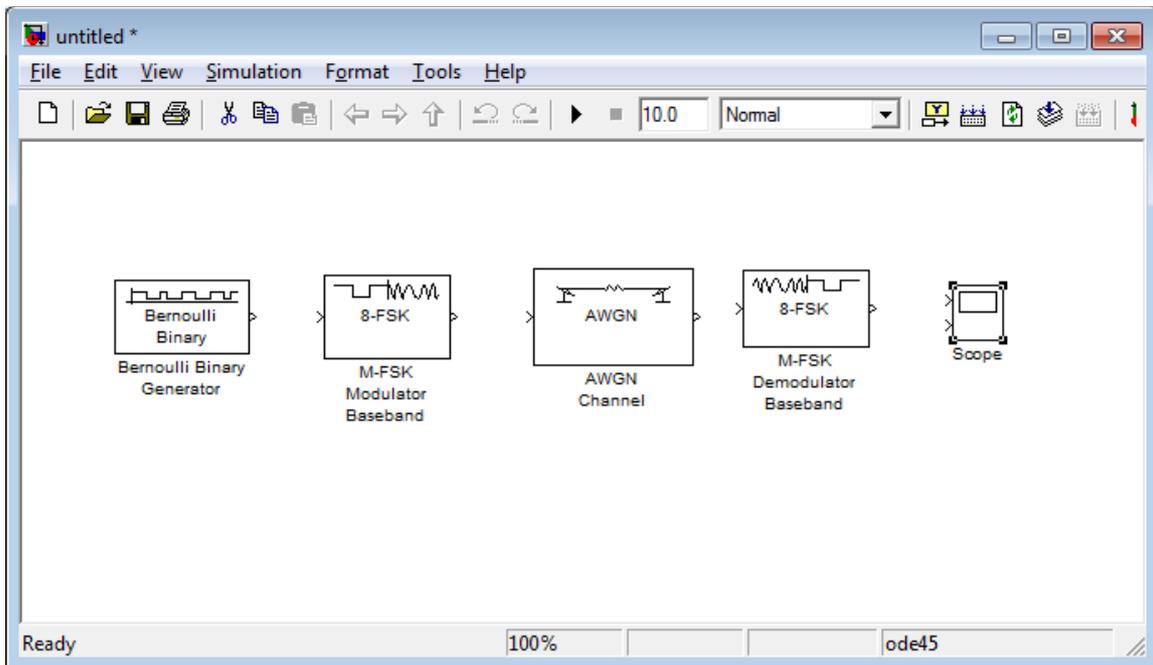


Figura III.35. Modelo Casi terminado.

Por omisión el bloque Scope (Osciloscopio) tiene sólo una entrada, al dar doble clic sobre el bloque y posteriormente en Parámetros, podemos cambiar a 2 entradas como lo muestra la figura III.36.

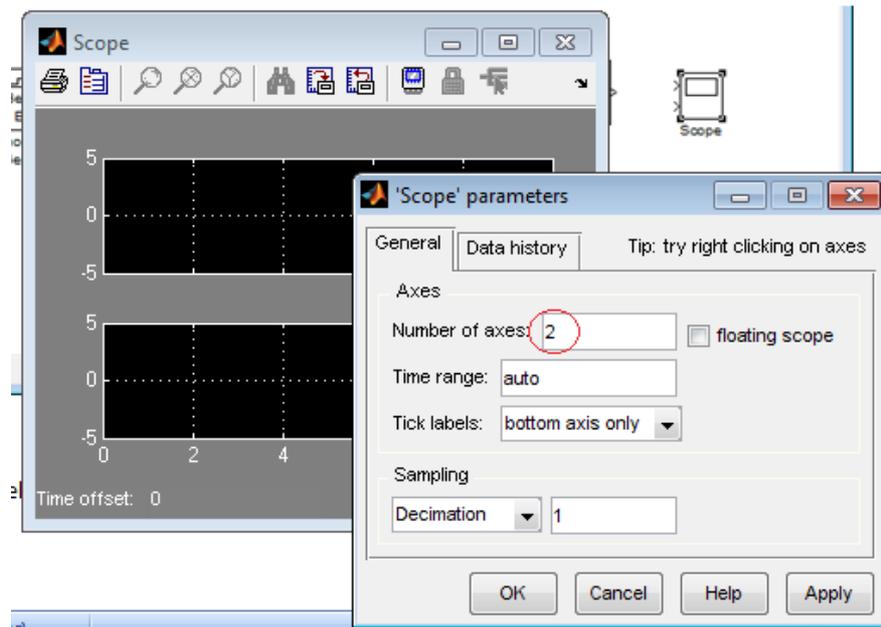


Figura III.36. Configuración Bloque Scope.

Este modelo se guarda como **ONDAS_DETERMINISTICAS_Y_ALEATORIAS.MDL**, como lo muestra la figura III.37.

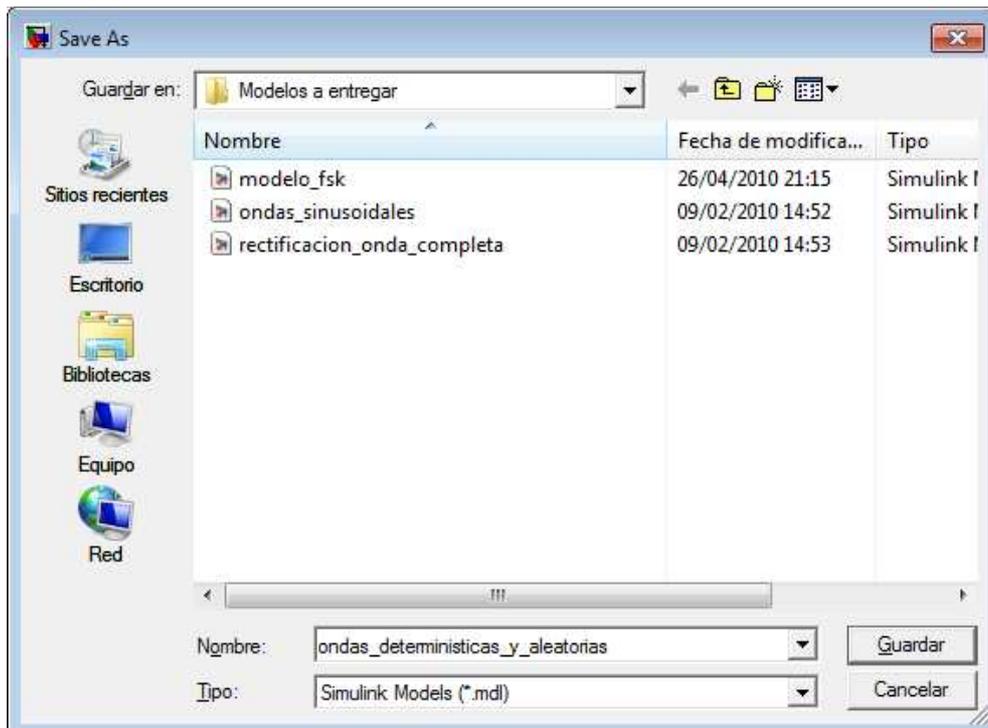


Figura III.37. Guardar Modelo Simulink Creado.

Finalmente conectamos los bloques con las señales, representadas en Simulink como flechas, esto lo muestra la figura III.38.

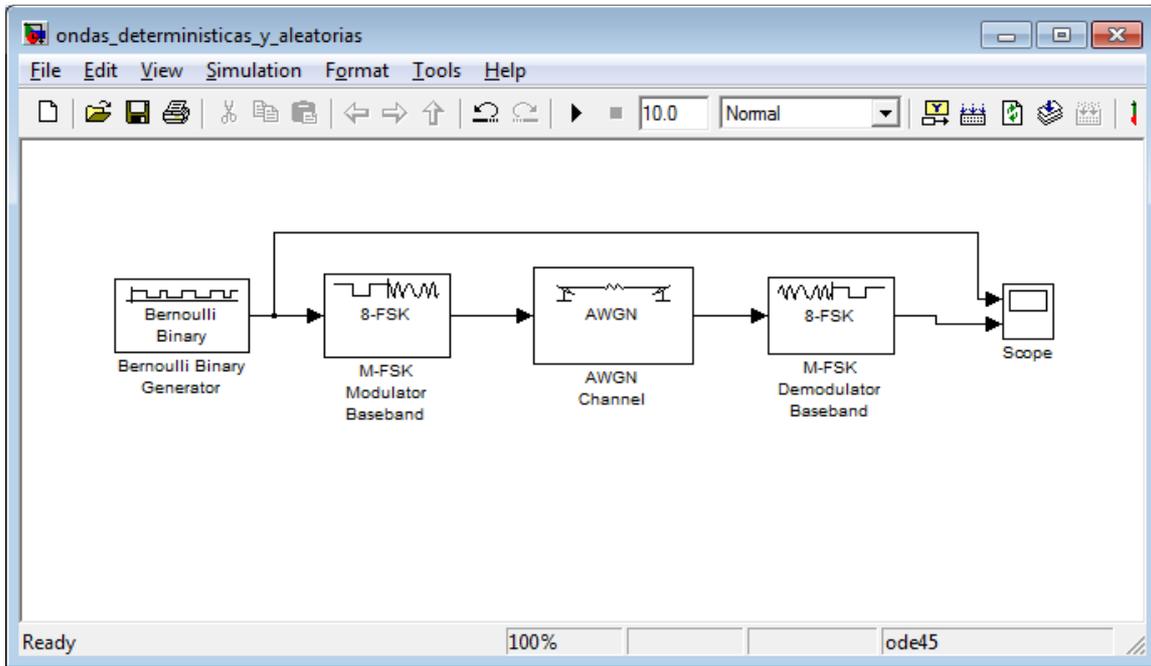


Figura III.38. Modelo Final listo para simularse.

Corremos la simulación y comprobamos en el bloque Scope la salida de la información con ruido gaussiano, y la fuente original de información (figura III.32). Podemos cambiar el parámetro del tiempo a 20 segundos o más y comparar los resultados de las formas de onda emitidas. (Figura III.39).



Figura III.39. Cambio interactivo del tiempo de ejecución de la simulación.

La señal primera que ingresa en el bloque de osciloscopio genera pulsos en forma aleatoria, pero de forma determinística. En la segunda entrada, a causa del canal de ruido blanco gaussiano, se genera un pulso que ya no es determinístico, por lo tanto es aleatorio, en el **segundo 18**, la amplitud sube al pico como lo muestra la figura III.40, esto claro está puede variar cuando construyan el modelo y apliquen el tiempo que requieran necesario de simulación.

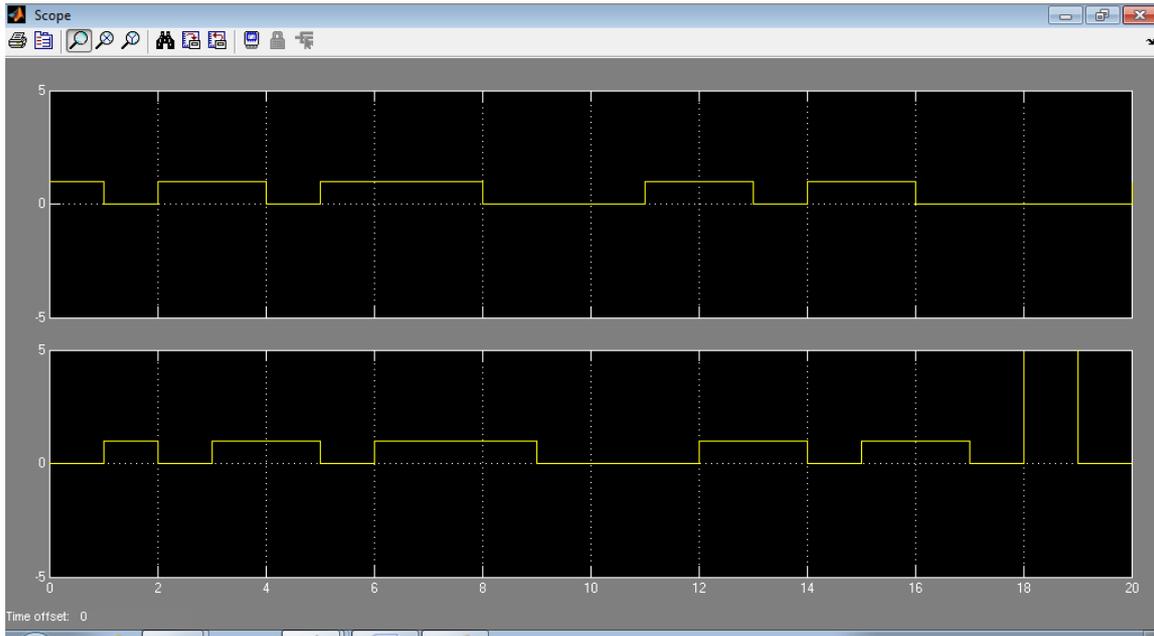


Figura III.40. Salida del bloque Scope (Osciloscopio) de una señal determinística y de una señal aleatoria.

Conclusiones y Recomendaciones

Se concluye que se ha podido distinguir la diferencia entre estos dos tipos de formas de onda, la determinística y la aleatoria, gracias a esta simulación de Simulink

Se recomienda tratar de usar otros bloques de entrada y canales o señales de ruido, y ver los resultados, recordando que se han usado bloques de comunicación FM digital (FSK-Desplazamiento de Frecuencia).

3.3 PRÁCTICA CAPACIDAD DE CANAL Y SISTEMAS DE COMUNICACIONES IDEALES.

OBJETIVOS

Evaluar el teorema de Claude Shannon, en lo que se refiere a la capacidad del canal.

MARCO TEÓRICO.

Capacidad de Canal y Sistemas de Comunicaciones Ideales

Para verificar un sistema de comunicación es ideal o perfecto, puede utilizarse muchos criterios. Para los sistemas digitales, el sistema óptimo es aquel que minimiza la probabilidad de error de bit a la salida del sistema sujeto a las restricciones de la energía transmitida y del ancho de banda del canal. Por lo tanto, el error de bit y el ancho de banda de la señal son de primordial importancia. Ello origina la siguiente pregunta: ¿Acaso es posible la invención de un sistema sin error de bit a la salida aun cuando se introduzca ruido en el canal? Claude Shannon respondió a esta pregunta en 1948-1949. La respuesta es afirmativa bajo ciertas suposiciones. Shannon demostró que (para el caso de una señal con ruido blanco gaussiano añadido) se puede calcular una capacidad de canal C (bits/s) tal que si la velocidad de información R (bits/s) era menor a C , entonces la probabilidad de error se aproxima a cero. La ecuación para C es

$$C = B \log_2 \left(1 + \frac{S}{N} \right)$$

Donde B es el ancho de banda del canal en Hertz (Hz) y S/N es la relación de potencia de señal a ruido (vatios/vatios, no dB) a la entrada del receptor digital. Shannon no describe cómo construir este sistema, pero si demuestra que tener tal sistema es teóricamente posible. Por tanto, Shannon aporta un límite de rendimiento teórico que no puede alcanzarse con sistemas de comunicación prácticos. Los sistemas que se aproximan a este límite a menudo incorporan codificación para la corrección de errores.

El sistema óptimo en los sistemas analógicos es aquel que puede alcanzar la relación más grande de señal a ruido a la salida del receptor, sujeta a las restricciones de diseño tales como el ancho de banda del canal y la potencia transmitida. En este caso la evaluación de la relación de señal a ruido de salida es de suma importancia. Se puede preguntar si es posible diseñar un sistema con una relación de señal a ruido infinita a la salida cuando se introduce ruido al canal. La respuesta es negativa.

Otros límites fundamentales para la señalización digital fueron descubiertos por Nyquist en 1924 y Hartley en 1928. Nyquist demostró que si un pulso representa un bit de datos, se puede enviar

pulsos no interferentes a través de un canal a una velocidad no mayor a $2B$ pulsos/s, donde B es el ancho de banda del canal en hertz. Esto se conoce ahora como el teorema de dimensionalidad. Hartley generalizó el resultado de Nyquist para el caso de una señalización de pulso multinivel.

Medición de Información

Como se ha visto, el propósito de los sistemas de comunicación es transmitir información de una fuente a un receptor. Sin embargo, ¿qué se entiende exactamente por información y cómo se mide?

Se sabe que cuantitativamente está relacionada a la sorpresa que se siente cuando se recibe un mensaje. Por ejemplo, el mensaje: “El océano ha sido destruido por una explosión nuclear” contiene más información que el mensaje: “Hoy está lloviendo”.

Definición: La información enviada de una fuente digital cuando se transmite el j -ésimo mensaje está dada por

$$I_j = \log_2 \left(\frac{1}{P_j} \right) \text{ bits} \quad (1.2.1)$$

Donde P_j es la probabilidad de transmitir el j -ésimo mensaje.

De esta definición se deduce que los mensajes que tienen una menor probabilidad de ocurrencia (valores más pequeños para P_j) suministran más información (valores grandes de I_j). También se observa que la medida de observación depende sólo de la probabilidad de envío del mensaje y no de la posible interpretación del contenido en referencia a si se tiene sentido o no.

La base del logaritmo determina las unidades utilizadas para la medida de información. Por lo tanto se emplea el logaritmo de base 2 para unidades de “bits”. Si se usa el logaritmo natural, las unidades son “nats”, y para logaritmos de base 10 la unidad es el “hartley”, llamado así en honor a R.V. Hartley, quien fue el primero en sugerir el uso de la media logarítmica en 1928.

Para facilitar la evaluación de I_j en una calculadora o en nuestro caso usando MATLAB se puede escribir la ecuación 1.2.1 en términos del logaritmo de base 10 o logaritmo natural, que corresponde a la **parte c**, de nuestro archivo-M. (COUCH, 2008)

Desarrollo

Transcribir o copiar del CD anexo a esta tesis el siguiente archivo-M y comprobar la capacidad del canal usando el teorema de Claude Shannon. Al correr el gui3n se pueden dar algunos valores, en lo que se refiere a la informaci3n a ser enviada y el ancho de banda en hercios del canal. Esto lo muestra la figura 2.2.1.

%% PRACTICA2_2.M

```
% Capacidad del canal, ley de Shannon.
% Fuente: Sistemas de Comunicación Digitales y Analógicos.
% Leon W. Couch II, 7ma Ed. Pearson Educación.
% ISBN: 978-970-26-1216-2 pp.18
%% TESIS DE GRADO
% Hernando Santillán
% Lunes, 26 Julio 2010 1122Hrs

clc;
clear;

% Parte a.
% chars = Número disponible de caracteres.
chars = input('Entre el número disponible de caracteres (chars): ');
% Determinando el número de bits requeridos para representar un caracter.
b = ceil(log(chars)/log(2));
fprintf('\n\nEl número de bits requerido para representar un caracter =
%G\n\n',b);

% Parte b.
% B = Ancho de Banda del Canal en Hz.
% SNRdB = relación Señal a Ruido en dB.
B = input('Entre el ancho de banda del canal en Hz: ');
SNRdB = input('Entre la relación Señal a ruido en dB: ');

SNR = 10^(SNRdB/10);

% Usando el teorema de capacidad del canal de Shannon.
C = B*log(1 + SNR)/log(2); %% página 18 del libro Sistemas de
Comunicaciones
%Digitales y Analógicas, Couch. 7ma Ed.
fprintf('\n\nLa capacidad del canal es %G (bits/seg)\n',C);
fprintf('o su equivalente %G (chars/seg)\n\n',C/b);

% Parte c.
% Cálculo de la información contenida en cada carácter (char) asumiendo
que
% cada carácter es igual al que se envió.

% p = probabilidad de un carácter dado que fue enviado.
p = 1/chars;
I = -log(p)/log(2);
fprintf('La información contenida en cada caracter es %G (bits)\n',I);
% info parecida al cálculo de b (número de bits para representar un
% caracter).
```

Correr el archivo-M y probar algunas opciones de envío de información y ancho de banda del canal en hercios.

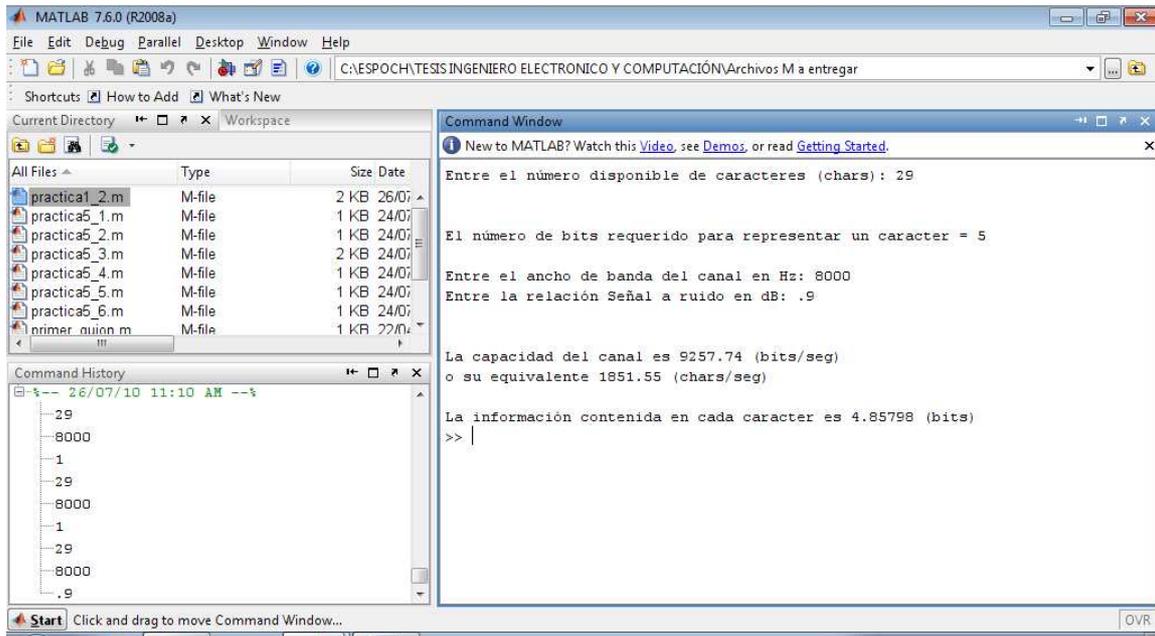


Figura III.41. Ejecución archivo-M cálculo capacidad del canal usando el teorema de Shannon.

Conclusiones y Recomendaciones

Se ha comprobado la parte teórica concerniente al teorema de Claude Shannon, que se tiene dentro del marco teórico de esta práctica.

Se concluye además que este tipo de programación estructurada es de amplio beneficio para el ingeniero de comunicaciones en lo que se refiere al cálculo de la capacidad del canal de cualquier medio sea guiado o no guiado.

Se recomienda aplicar o escribir otros archivos-M para el cálculo de otros factores importantes en lo que se refiere al envío y recepción de información por canales analógicos o digitales. Tales como por ejemplo la evaluación de la información y entropía. (COUCH, 2008) pp. 17.

3.4 PRÁCTICA POTENCIA.

OBJETIVOS

Determinar la Potencia en función del tiempo.

MARCO TEÓRICO.

Potencia

En los sistemas de comunicación, si la potencia (promedio) de señal recibida es suficientemente grande en comparación a la del ruido, entonces la información podrá recuperarse. Este concepto se demostró mediante la fórmula de capacidad de canal de Shannon. Por consiguiente, la potencia promedio es un concepto importante que requiere comprensión. Gracias a la física se conoce que la potencia está definida como trabajo por unidad de tiempo; el voltaje es el trabajo por unidad de carga y la corriente es carga por unidad de tiempo. Esta es la base para la definición de la potencia en términos de cantidades eléctricas.

Definición

Suponga que $v(t)$ denote el voltaje a través de un conjunto de terminales de circuito, y que $i(t)$ represente la corriente a la terminal, como en la figura III.42 lo muestra. La potencia instantánea (trabajo incremental dividido por el tiempo incremental) asociado con el circuito está dada por

$$p(t) = v(t)i(t)$$

Donde la potencia instantánea fluye dentro del circuito cuando $p(t)$ es positiva y fluye fuera del circuito cuando $p(t)$ es negativa. La potencia promedio es (COUCH, 2008)

$$P = \langle p(t) \rangle = \langle v(t)i(t) \rangle$$

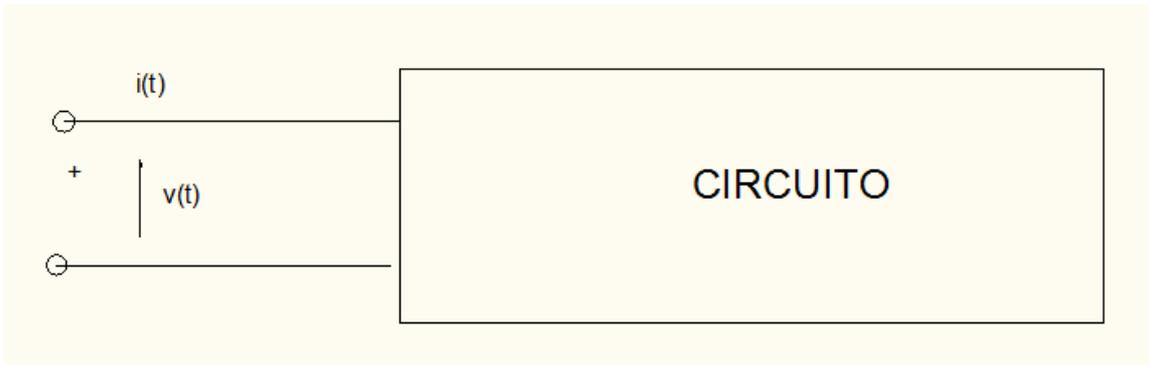


Figura III.42. Convención de polaridad utilizada para el voltaje y la corriente.

Desarrollo

Transcribir o copiar del CD anexo a esta tesis el siguiente archivo-M y comprobar la relación entre la potencia y el tiempo, considerando que Potencia = Voltaje por Corriente, según la ley de Ohm. Al correr el gui3n se puede observar la gr3fica que representa estas dos funciones contrastadas: la potencia y el tiempo. Esto lo muestra la figura III.43.

```
%% PRACTICA3_1.M
% Potencia en funci3n del tiempo.
%% TESIS DE GRADO
% Hernando Santill3n
% Jueves, 29 Julio 2010 1133Hrs

clc;
clear;

fo = 60; %frecuencia de operaci3n.
t = 0:0.0002:0.03; %vector tiempo.
wo = 2*pi*fo; %velocidad angular omega 0.

% Seleccione el 3ngulo teta para que sea el desplazamiento de fase de la
% corriente en grados.
teta = 20; %20 grados.

corriente = 5*cos(wo*t + teta*(pi/180));
voltios = 10*cos(wo*t);

corriente = corriente(:);
voltios = voltios(:);

potencia = corriente.*voltios;

plot(t,potencia);
xlabel('tiempo');
ylabel('potencia');
```

```
title('Potencia en función del tiempo');  
grid;
```

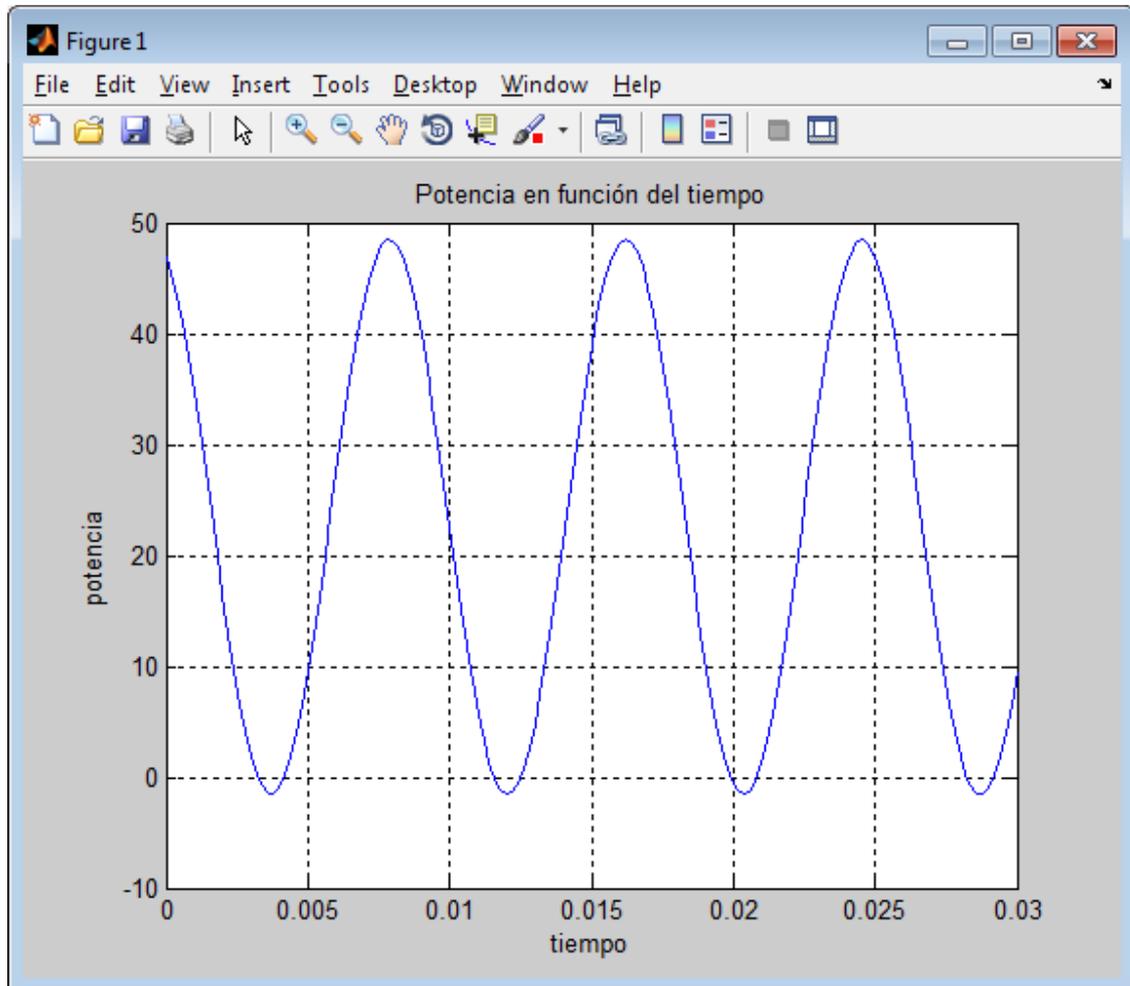


Figura III.43. Salida relación Potencia en función del tiempo.

Conclusiones y recomendaciones

Se concluye de esta práctica que la relación voltaje-corriente aplicado a la potencia, según la ley de Ohm se las puede analizar usando una gráfica de MATLAB que nos muestra una señal sinusoidal.

Se recomienda graficar otras relaciones físicas conocidas.

3.5 PRÁCTICA SERIE DE FOURIER.

Rectificación de una Onda Completa utilizando el método de la Serie de Fourier con MATLAB.

Objetivo: Ver una ejecución con SIMULINK de una onda senoidal con rectificación de onda completa utilizando el método de Fourier.

Antecedentes Históricos.

En 1768, nació Joseph Fourier. Como estudiante mostró talento para las matemáticas pero se dedicó a ellas profesionalmente, sólo cuando su nivel social (era hijo de un sastre) le hizo imposible obtener una comisión militar.

La descripción matemática de la conducción del calor era un problema muy importante al inicio del siglo XIX. En 1807, Fourier envió un artículo sobre este tema a la prestigiosa Academia de Ciencias de París, para competir por un premio que se había ofrecido al estudio más exitoso de este problema. Algunos gigantes matemáticos como Laplace, Lagrange y Legendre evaluaron el trabajo de Fourier y lo rechazaron por falta de rigor. Sin embargo, alentaron a Fourier para continuara con su investigación y a que intentara completar los detalles que había omitido. En 1811, Fourier envió una versión revisada de su artículo y obtuvo el premio concedido por el comité. Sin embargo, todavía la academia declinó publicar el artículo porque muchos detalles todavía no estaban claros. Finalmente en 1822, Fourier publicó *Théorie analytique de la chaleur*, hoy en día su trabajo clásico, incorporando la mayor parte de sus resultados de 1811 junto con algunos nuevos.

En este artículo, Fourier consideró la ecuación de calor en tres dimensiones

$$\frac{\partial u}{\partial t} = k \left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right]$$

En donde $u(x, y, z, t)$ es la temperatura de un objeto en el tiempo t y en el punto (x, y, z) . Usando lo que hoy se conoce como separación de variables, encontró las representaciones en series trigonométricas de las soluciones. En ellas utilizó las fórmulas para los coeficientes que se conocen ahora como coeficientes de Fourier.

Este epígrafe tomado de las páginas 588-589 del texto (O'NEIL, 2004) , nos explica que alrededor de 15 años le tomo a Jean Joseph Marie de la Fourier demostrar su teoría de calor en la Academia de Ciencias de París, en la que la suma y resta de sus coeficientes aplicados a sus series

trigonométricas nos permiten demostrar muchas señales que son básicas en el estudio del comportamiento de sistemas electrónicos de comunicaciones y control.

Antecedentes teóricos.

MATLAB y su programa SIMULINK se usan para modelar y simular varios comportamientos de sistemas. Usamos SIMULINK para ejecutar la ecuación siguiente:

$$v(t) = \frac{2V}{\pi} - \frac{4V \cos(2\omega t)}{\pi(1)(3)} - \frac{4V \cos(4\omega t)}{\pi(3)(5)} - \frac{4V \cos(6\omega t)}{\pi(5)(7)} - \frac{4V \cos(8\omega t)}{\pi(7)(9)} \quad (ec. 4.1)$$

La cual representa los primeros cinco términos de la serie de Fourier de una onda senoidal con rectificación de onda completa. El valor **V** corresponde a la amplitud de la onda, que para nuestro modelo es 1 voltio. (BLAKE, 2004) página 12.

Desarrollo del modelo usando SIMULINK

Como ya revisamos anteriormente al crear nuestro modelo de ondas sinusoidales en la sección anterior (primeros pasos en el uso de MATLAB y SIMULINK), en donde utilizaremos los bloques ya conocidos como son: Onda Seno (Sin Wave) de la biblioteca Fuentes (Sources) para las 4 señales Coseno de la ecuación 2.1 , Osciloscopio (Scope) de la biblioteca Bloques Comúnmente Usados (Commonly Used Blocks), para obtener su salidas. Nos resta ahora tener un bloque que nos permita sumar estas 4 señales llamado Suma (**Sum**), que se lo encuentra en la biblioteca de bloques comúnmente usados, como se muestra en la figura III.44.

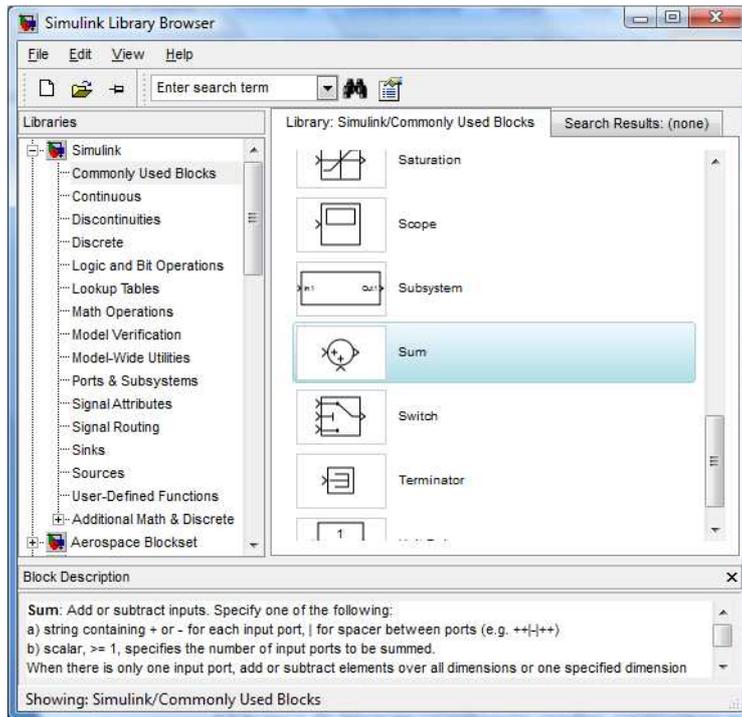


Figura III.44. Bloque Suma biblioteca Bloques comúnmente usados.

Aquí un pequeño tip, cuando solo recordemos el nombre del bloque que deseemos usar en nuestros modelos y no recordemos la biblioteca a la que corresponde, así como ahora necesitamos buscar los bloques Constante (**Constant**), que me permite representar el primer coeficiente de la serie, es decir $\frac{2V^5}{\pi}$ y el bloque Ganancia (**Gain**) para poder aplicar a cada uno de los coeficientes de las cuatro señales Coseno de la serie.

En el campo de búsqueda escribimos **Constant** y tenemos como resultado algunos bloques que contienen la palabra buscada, como se muestra en la figura III.45.

⁵ V=1 voltio, este es el valor medio de continua de la onda.

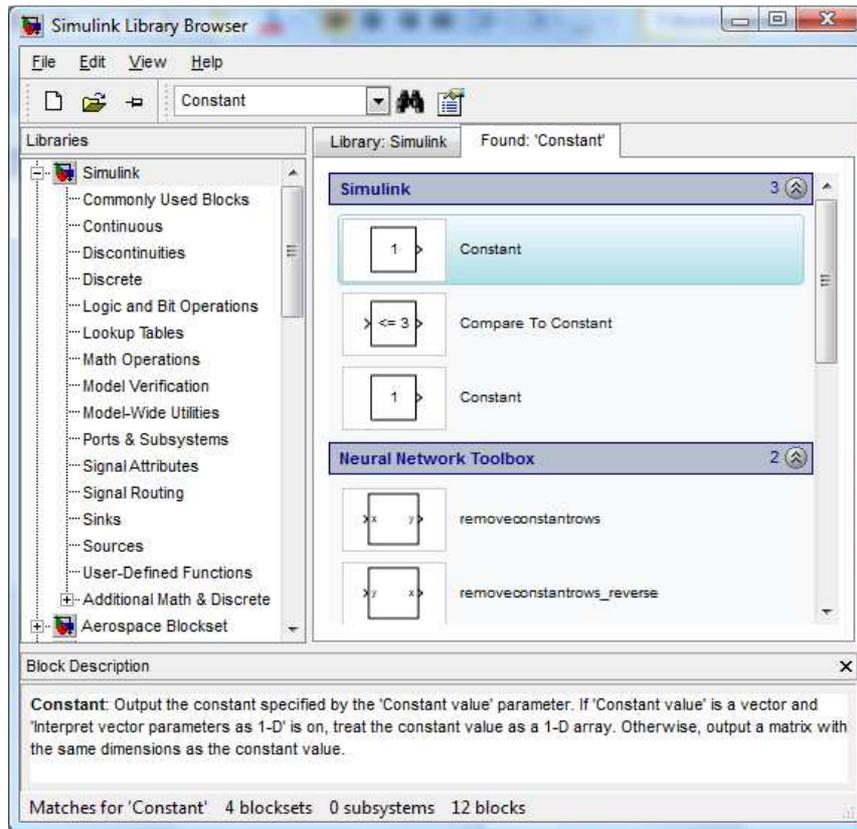


Figura III.45. Búsqueda de bloque Constante

Sin embargo, los tres bloques adicionales que se requieren para la construcción de este nuevo modelo que representa esta serie de Fourier: **Constante**, **Ganancia** y **Suma** se los encuentra en la biblioteca de bloques comúnmente usados (**Commonly Used Blocks**).

Esto lo podemos apreciar en la figura III.46, posterior a esto ya con nuestros elementos o bloques procederíamos a cambiar de nombres a las ondas Senos por Cosenos, la constante, ganancias y osciloscopio, para tener nuestro modelo como la que tenemos en la figura III.47.

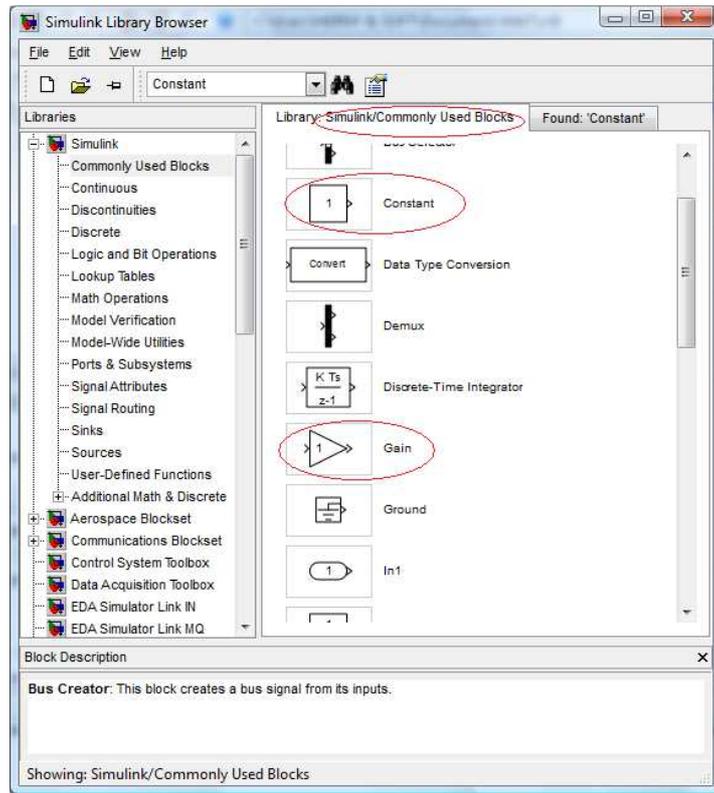


Figura III.46. Bloques Constante y Ganancia de la biblioteca Bloques Comúnmente usados de Simulink.

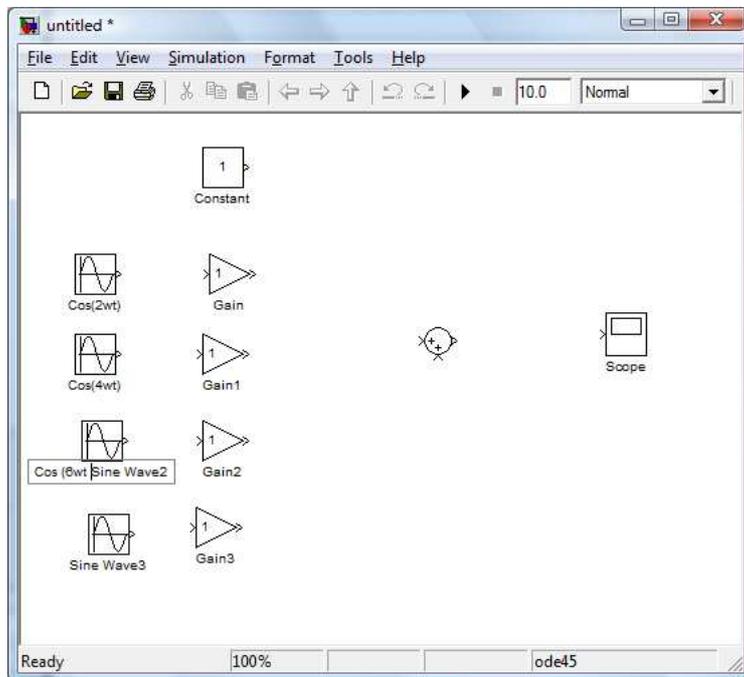


Figura III.47. Bloques para modelar la Serie de Fourier de una Onda Senoidal con Rectificación de Onda Completa.

Como ya lo revisamos en el apartado 1 de este capítulo, debemos configurar los valores de cada uno de los bloques, por tanto el bloque coseno ($2\omega t$), por ejemplo se configura de acuerdo a como lo muestra la figura III.48.

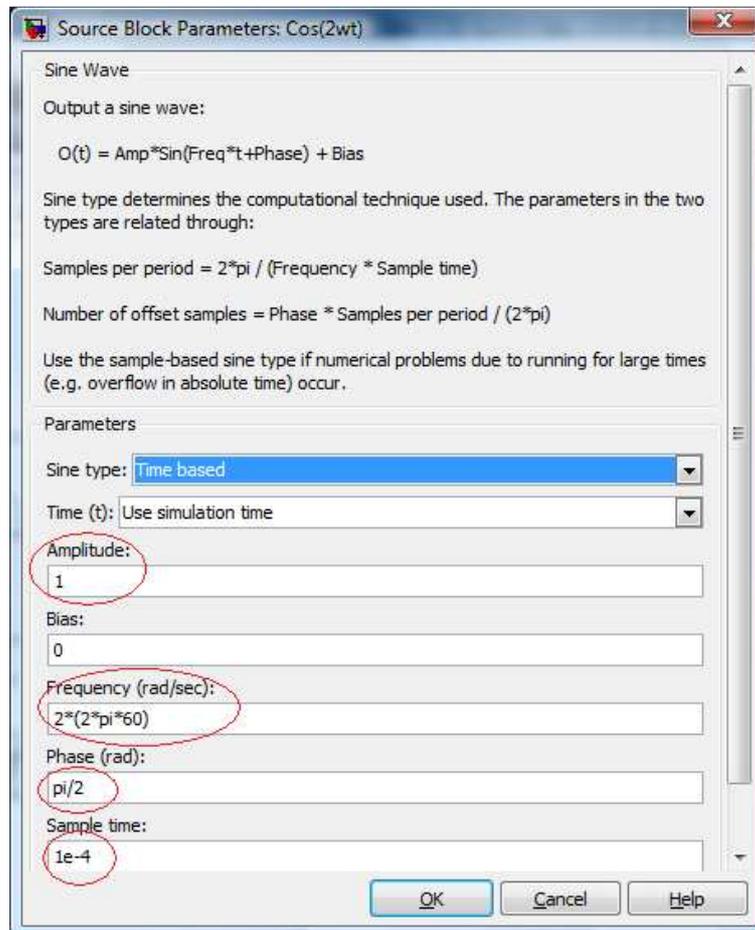


Figura III.48. Representación Coeficiente $\cos(2\omega t)$ en el bloque Sin Wave

Salta a la vista que en campo de la frecuencia hemos considerado el $\cos(2\omega t)$ de la forma $2 * (2 * \pi * 60)$ en radianes, recordemos siempre que $\omega = 2\pi f$ y la frecuencia para nuestros ejemplos se ha considerado $f = 60\text{Hz}$, este parámetro puede variar de acuerdo a las necesidades de los sistemas a modelar. Está claro además que este valor pudiese ser cambiado a $240 * \pi$ sin alterar el resultado, de hecho al representar el coeficiente $\cos(4\omega t)$ tendríamos en el campo frecuencia $4 * (120 * \pi)$ o $480 * \pi$ o $4 * 120 * \pi$ con el mismo efecto deseado. Esto lo mostramos en la figura III.49.

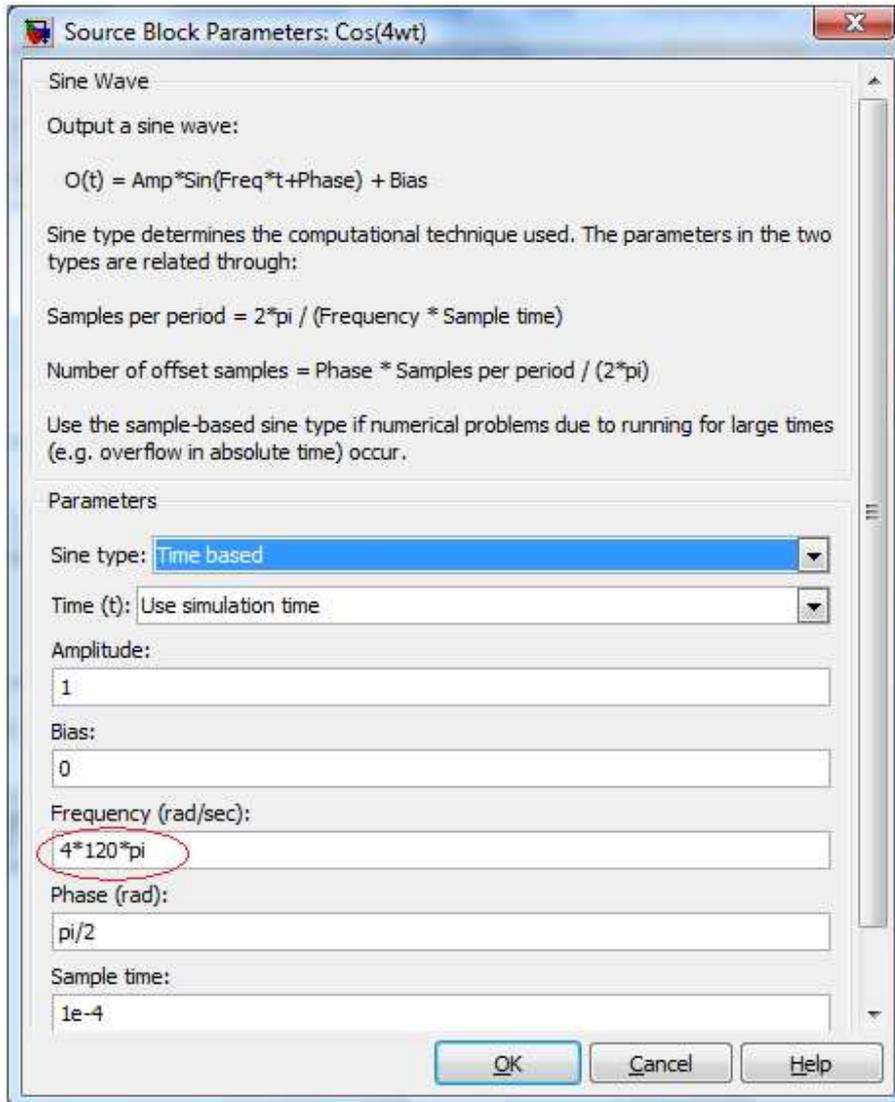


Figura III.49. $\cos(4\omega t)$, siendo $\omega = 2\pi f$

Con los dos restantes coeficientes, siguiendo estas pautas tendríamos en el campo frecuencia de la onda coseno el valor de: $6 * (120 * \pi)$ o $6 * 120 * \pi$ para el término $\cos(6\omega t)$ y $8 * 120 * \pi$ para el término $\cos(8\omega t)$.

Recordemos que en cada bloque para que sea una onda coseno la fase (**Phase**) es de $\frac{\pi}{2}$ radianes, y el tiempo de muestreo (**Sample time**) es de 0.1 ms o sea $1e-4$ s.

Ya definidas las 4 señales coseno, debemos definir el voltaje constante correspondiente a $\frac{2V}{\pi}$ para esto al dar doble clic en bloque Constant que lo hemos cambiado de nombre a Voltaje Constante, procedemos a configurarlo como se muestra en la figura III.50.

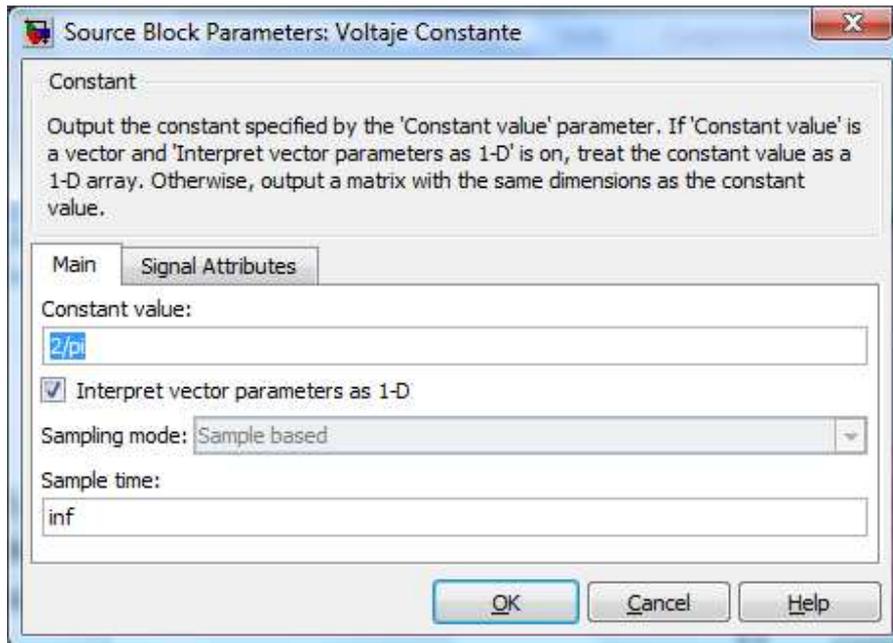


Figura III.50. Cambio de valor constante

Simplemente cambiamos el valor por omisión 1 por el valor que deseemos, puede ser un número entero o como en nuestro caso un valor real $2/\pi$ ⁶, los demás parámetros especificados como el tiempo de la muestra (**Sample time**) y el casillero marcado Interprete los parámetros del vector de una dimensión (**Interpret vector parameters as 1-D**) se mantienen como nos muestra la figura III.50.

Ahora debemos cada una de las cuatro señales coseno aplicar una ganancia negativa de $-\frac{4V^7}{x\pi}$ siendo $x = 3,3 * 5,5 * 7,7 * 9$ o $x = 3,15,35,63$. Por lo tanto en cada bloque de Ganancia tendríamos que colocar este valor en el casillero de Ganancia representado en la figura III.51, previo a esto podemos cambiar como sabemos la etiqueta del bloque **Gain** a una etiqueta más significativa para nuestra mejor comprensión, es decir Gain por $-4/(3 * \pi i)$.

⁶ El valor de pi en MATLAB es de:

```
>> pi()
```

```
ans =
```

```
3.1416
```

```
>>
```

⁷ Recuerde el valor de V es 1 voltio, por lo que no se lo considera como factor de multiplicación.

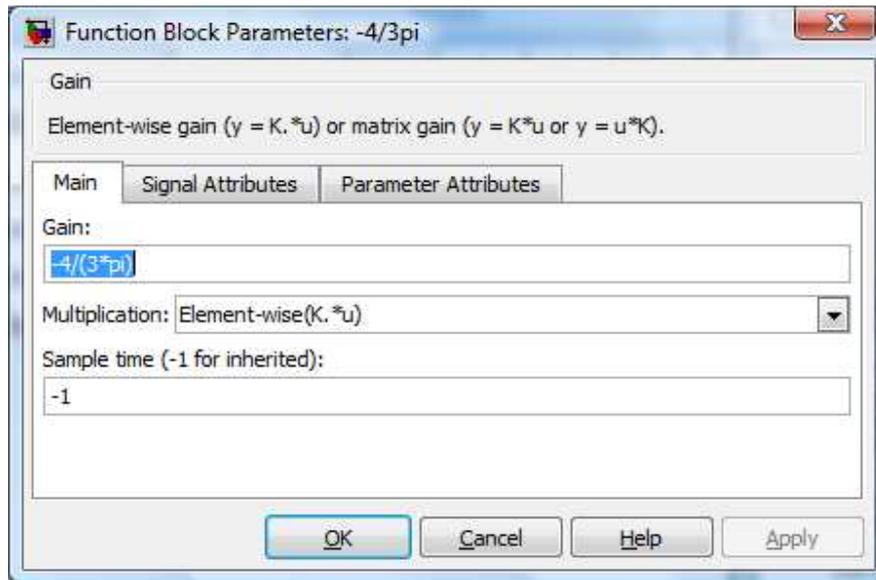


Figura III.51. Ganancia del término $\cos(2\omega t)$

Aquí igual que el bloque de constantes sólo cambiamos el valor de la ganancia, notamos que en este campo se escribe para representar $-\frac{4}{3\pi}$ de la forma $-4/(3 * \pi i)$, este bloque acepta además la notación $-4/\pi i/3$ con el mismo resultado (figura III.51). Una forma equivalente la tenemos en la figura III.52 donde la ganancia $-\frac{4}{(3)(5)\pi}$ correspondiente al coeficiente $\cos(4\omega t)$ se representa con esa otra notación.

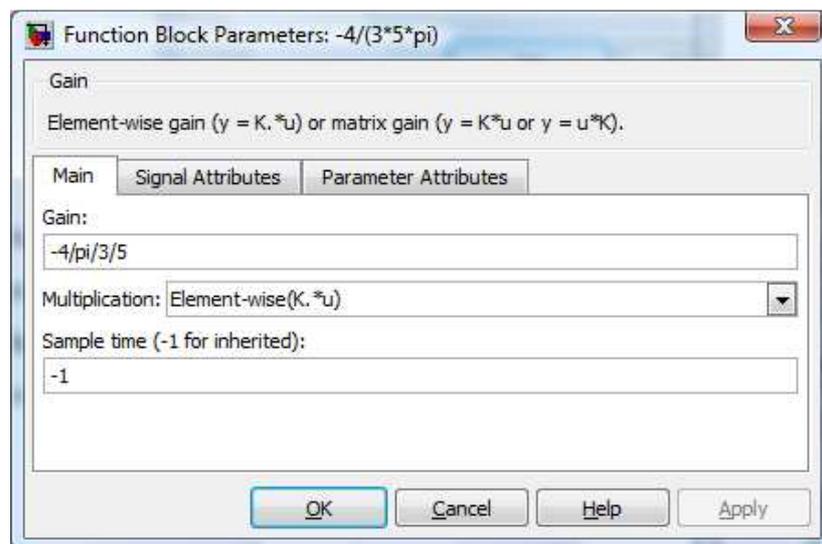


Figura III.52. Otra forma de representar la ganancia $-\frac{4}{\pi(3)(5)}$ de $\cos(4\omega t)$ de la ecuación 4.1.

Cualesquier de los métodos presentados para denotar la ganancia en estos bloques nos sirve para continuar y determinar la ganancia para los coeficientes de la función $\cos(6\omega t)$ de ganancia $-\frac{4}{(5)(7)\pi}$ y $\cos(8\omega t)$ de ganancia $-\frac{4}{(7)(9)\pi}$. Esto se registra en lo que muestra la figura III.53 en sus acápite a) y b).

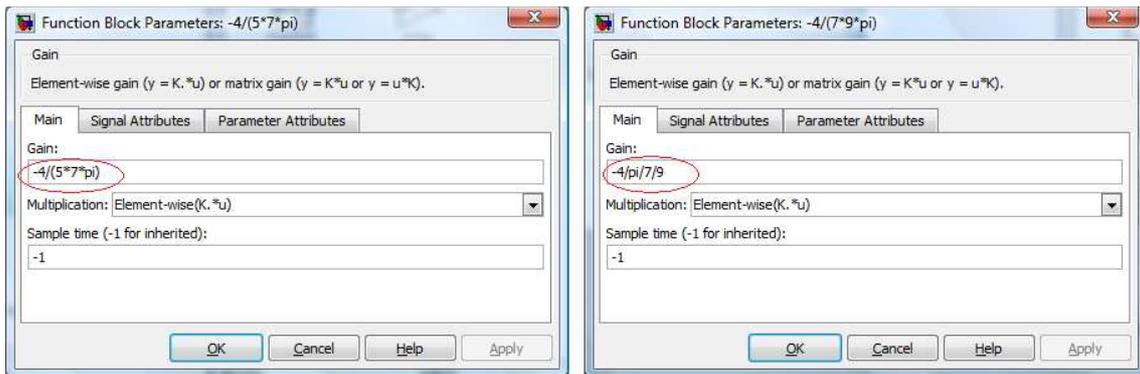


Figura III.53. Contraste de las dos formas de representar las ganancias de los 2 últimos coeficientes de esta serie. a) ganancia de $\cos(6\omega t)$ b) ganancia de $\cos(8\omega t)$.

Ya nuestro modelo debe parecerse al que se muestra en la figura III.54. Sin embargo todavía faltan algunos detalles, como definir más puntos de suma, ya que por omisión el bloque Suma (**SUM**)⁸ sólo tiene 2 entradas y al momento necesitamos 5 sumandos. Además recordemos como ya se vio en la primera sección el bloque de osciloscopio (**Scope**) igual va a necesitar 6 subdivisiones para mostrar todas y cada una de las cinco señales, más la señal sumada, es decir la serie trigonométrica completa de Fourier deseada del rectificador de onda completa. Refiérase a la figura III.55.

⁸ Este bloque corresponde a la biblioteca Bloques Comúnmente Usados.

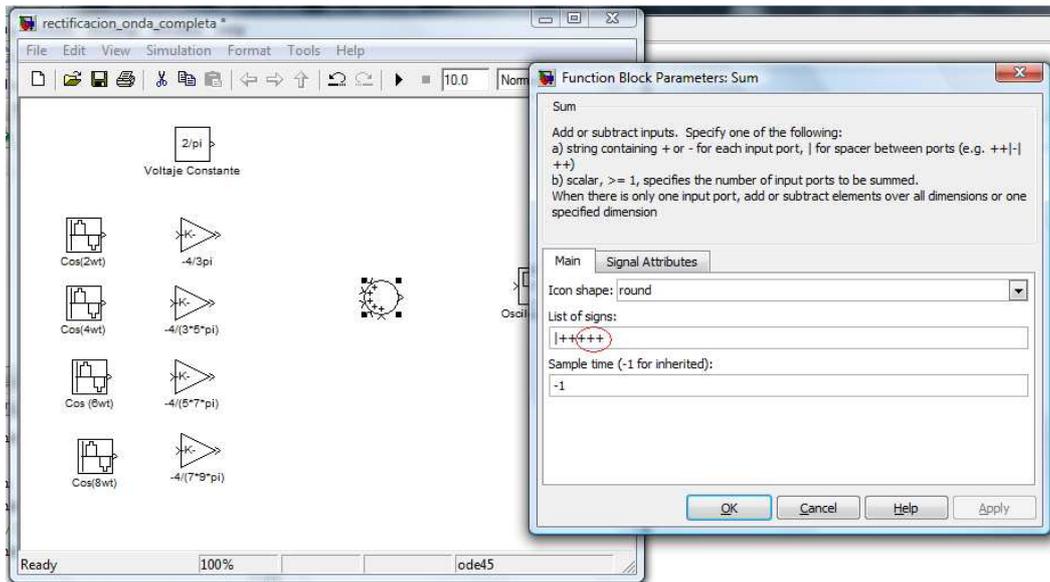


Figura III.54. Añadiendo señales de entrada al bloque Suma (Sum)

En el campo lista de signos (**List of Signs**) del bloque **Sum** que contiene dos signos + por omisión añade los tres signos (+) adicionales para poder representar la serie trigonométrica de Fourier de una onda senoidal rectificada en onda completa (full wave, en inglés) con cinco componentes (figura III.54).

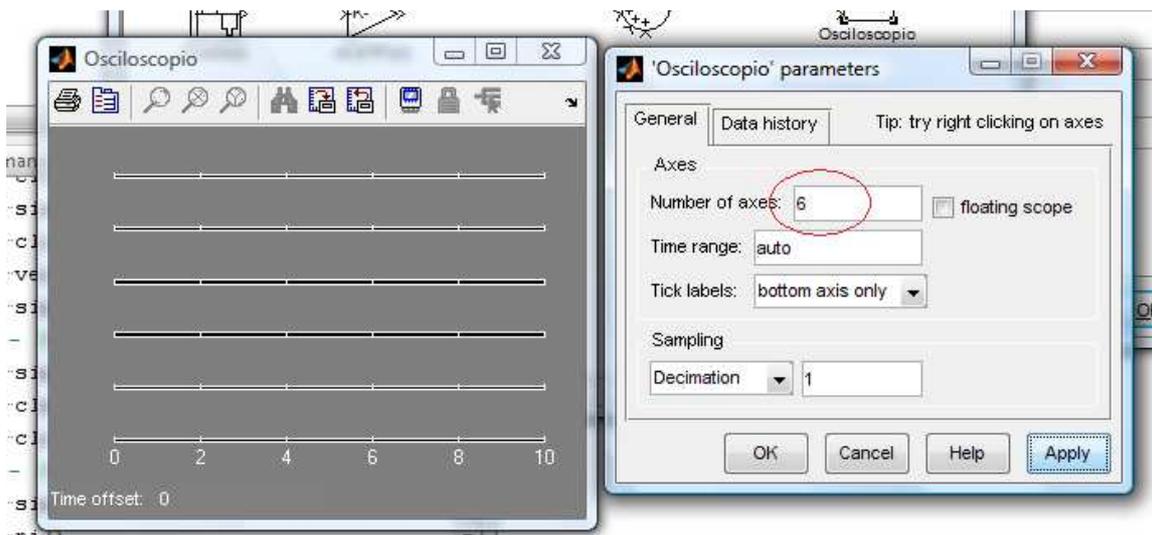


Figura III.55. División de las pantallas del Osciloscopio simulado.

Finalmente debemos unir cada uno de los bloques para obtener nuestro modelo ya finalizado tal como lo apreciamos en la figura III.57. Seguidamente corremos la simulación y concentrarnos nuestra atención en la salida 6 del osciloscopio que finalmente corresponde a la suma de las señales y comprobar que se trata de una onda sinusoidal rectificada en onda completa.

Al empezar a conectar los bloques aparecerá este tip indicativo (figura III.56), para realizar conexiones de forma más rápida al presionar la tecla de Control seleccionando cada uno de los bloques y luego dando clic izquierdo en el bloque de destino.



Figura III.56. Tip Uso de la tecla CONTROL

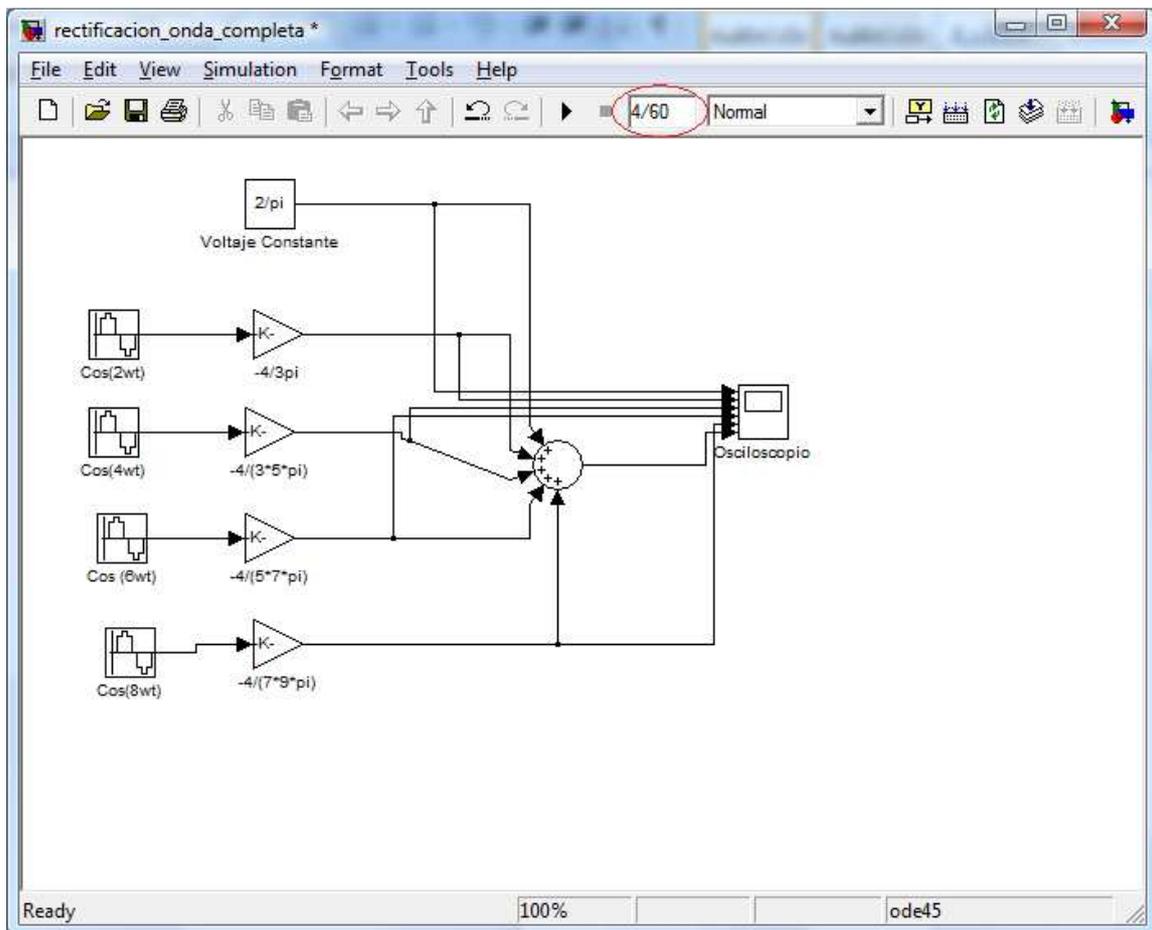


Figura III.57. Modelo terminado corresponde a la ec 2.1 $v(t) = \frac{2V}{\pi} - \frac{4V \cos(2\omega t)}{\pi(1)(3)} - \frac{4V \cos(4\omega t)}{\pi(3)(5)} - \frac{4V \cos(6\omega t)}{\pi(5)(7)} - \frac{4V \cos(8\omega t)}{\pi(7)(9)}$

Es importante determinar el tiempo de ejecución del modelo, hemos escogido **4/60** este valor puede ser cambiado dependiendo de la velocidad que se desea para analizar los datos. Como conclusión al correr este modelo se obtiene la siguiente salida del osciloscopio simulado según se muestra en la figura III.58.

Nótese además que la ecuación 4.1 tiene un valor variable de amplitud **V** que lo hemos definido como 1 voltio, sin embargo este valor podría tener cualesquier valor 2, 5 (voltios), etcétera. Sin embargo para efectos de simulación el modelo presentado con 1 voltio, responde al resultado deseado.

Recuerde siempre en cada salida aplicar la **auto-escala** al dar clic derecho sobre el osciloscopio simulado, especialmente en la sexta pantalla que tiene la onda rectificada en onda completa.

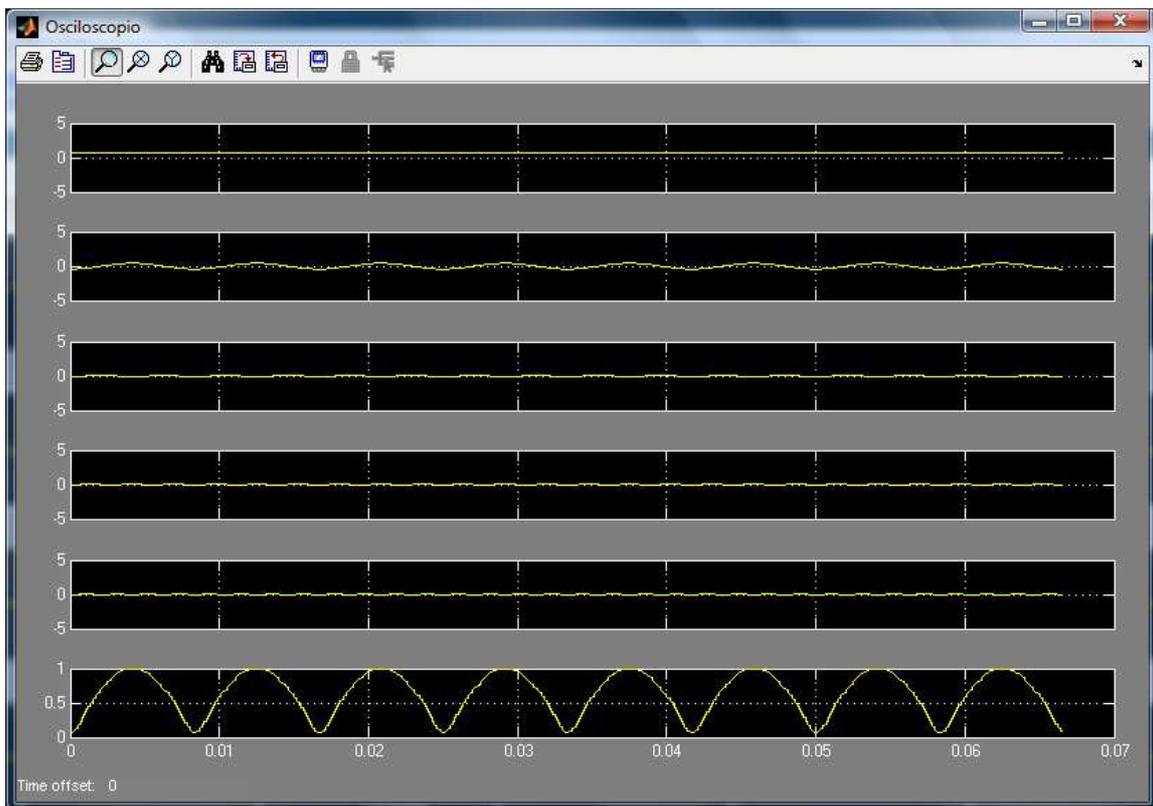


Figura III.58. Senoide con Rectificación de onda completa y sus componentes.

Apreciamos en la figura III.58 que nuestro objetivo se ha cumplido en forma exitosa, son necesarios tan sólo cinco coeficientes de la serie de Fourier para mostrar la onda rectificada en onda completa.

Refiérase al modelo **rectificación_onda_completa.mdl** del CD anexo a esta tesis.

Aplicación del modelo simulado al mundo real.

Uno de los puntos importantes al desarrollar este trabajo de investigación es llevar este tipo de demostraciones de modelos matemáticos como es la serie trigonométrica de Fourier a los campos de la ingeniería electrónica práctica. En consecuencia se pone a consideración dos aplicaciones que las conocemos muy bien, estas son: el circuito de un rectificador de onda completa convencional con dos diodos y el rectificador en puente con cuatro diodos.

Se menciona entonces la teoría correspondiente en la que se puede determinar que el estudio asistido por el computador de las señales de Fourier corresponde a estos dos típicos circuitos o dispositivos electrónicos que son usados muy frecuentemente en muchas aplicaciones reales de ingeniería electrónica.

Los siguientes epígrafes corresponden a un resumen de (ALBERT MALVINO, 2007), en sus páginas 91-95, que nos sirven para enlazar lo que se demostrado con el programa SIMULINK de MATLAB y aplicaciones de ingeniería electrónica del mundo real.

Circuito Rectificador de Onda Completa

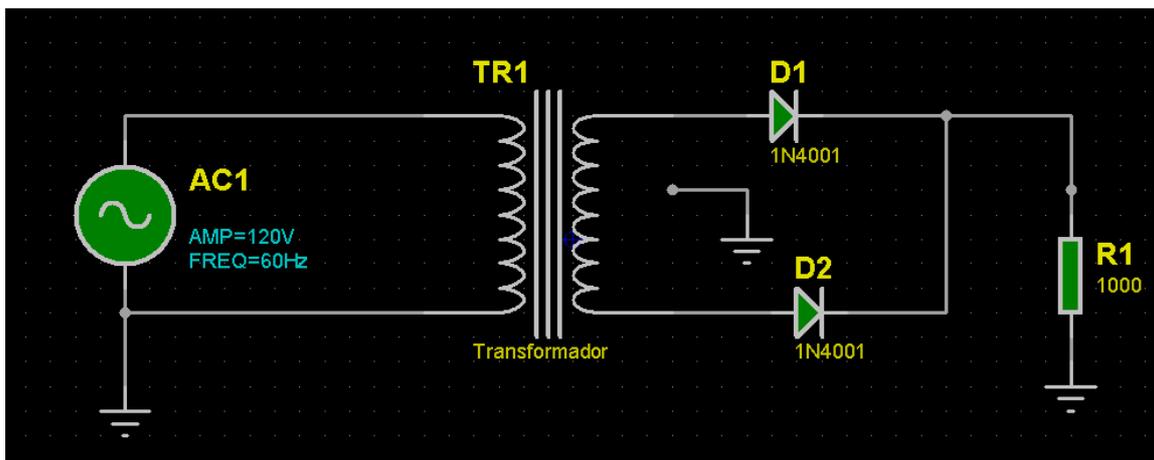


Figura III.59. Circuito Rectificador de Onda completa

La figura III.59 muestra un rectificador de onda completa. Observe que la conexión intermedia del devanado del secundario está conectada a masa (tierra). El circuito de onda completa es equivalente a dos rectificadores de media onda. Debido a la conexión central, cada uno de estos rectificadores tiene una tensión de entrada igual a la mitad de la tensión del secundario. El diodo D1 conduce durante el semiciclo positivo y el diodo D2 conduce durante el semiciclo negativo. Como resultado, la corriente de carga rectificada fluye durante ambos semiciclos. El rectificador de onda completa se comporta como dos rectificadores de media onda en oposición.

El circuito se denomina rectificador de onda completa, porque ha transformado la tensión alterna de entrada en una tensión de salida pulsante continua (figura III.58 primera pantalla del

osciloscopio $\frac{2V}{\pi}$). Esta forma de onda tiene algunas propiedades interesantes que se exponen a continuación.

Valor medio de continua

Dado que la señal de onda completa tiene el doble de ciclos positivos que la señal de media onda, el valor de continua o valor medio es el doble como mucho y vienen dado por:

$$\text{Onda completa: } V_{DC} = \frac{2V_p}{\pi} \text{ } ^9 \text{ (ec. 4.2)}$$

Puesto que $\frac{2}{\pi} = 0.636$ podemos escribir la ecuación 2.2 como:

$$V_{DC} \approx 0.636V_p$$

De esta forma podemos ver que el valor de continua o valor medio es igual al 63.3% del valor pico. Por ejemplo, si la tensión de pico de la señal de onda completa es 100V, el valor de continua o valor medio es igual a 63.6V.

Frecuencia de salida

Con un rectificador de media onda, la frecuencia de salida es igual a la de entrada. Pero con un rectificador de onda completa, algo inusual sucede a la frecuencia de salida. La tensión alterna de la red eléctrica tiene una frecuencia de 60Hz (50Hz en Europa). Por tanto el período es igual a:

$$T_{ent} = \frac{1}{f} = \frac{1}{60\text{Hz}} = 16.7[\text{ms}]$$

A causa de la rectificación de onda completa, el período de la señal de onda completa es la mitad que el período de entrada:

$$T_{sal} = 0.5(16.7 \text{ ms}) = 8.33[\text{ms}]$$

Cuando calculamos la frecuencia de salida tenemos:

$$f_{sal} = \frac{1}{T_{sal}} = \frac{1}{8.33\text{ms}} = 120[\text{Hz}]$$

La frecuencia de la señal de onda completa es el doble de la frecuencia de entrada. Esto es lógico: una señal de salida de onda completa tiene el doble de ciclos que una entrada sinusoidal. El rectificador de onda completa invierte cada semiciclo negativo, así se obtienen el doble de ciclos. El efecto de esto es que la frecuencia se duplica. Como una derivación tenemos que:

$$f_{sal} = 2f_{ent} \text{ (ec. 4.3)}$$

El rectificador en Puente

⁹ Que corresponde al valor constante de nuestra simulación.

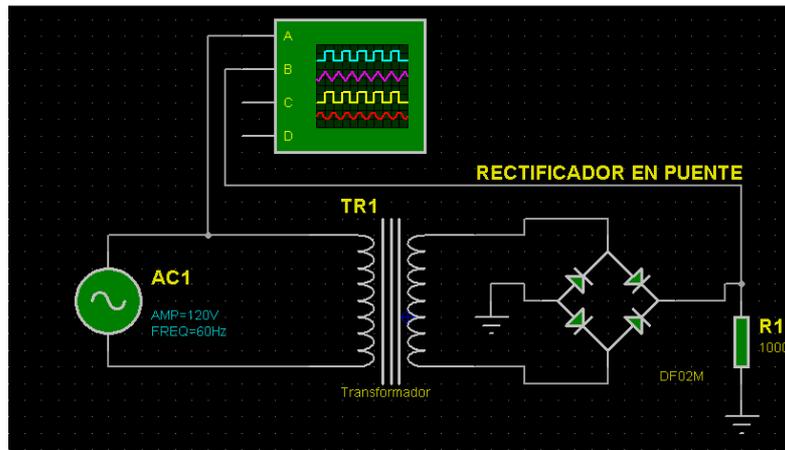


Figura III.60. a). Rectificador en puente.

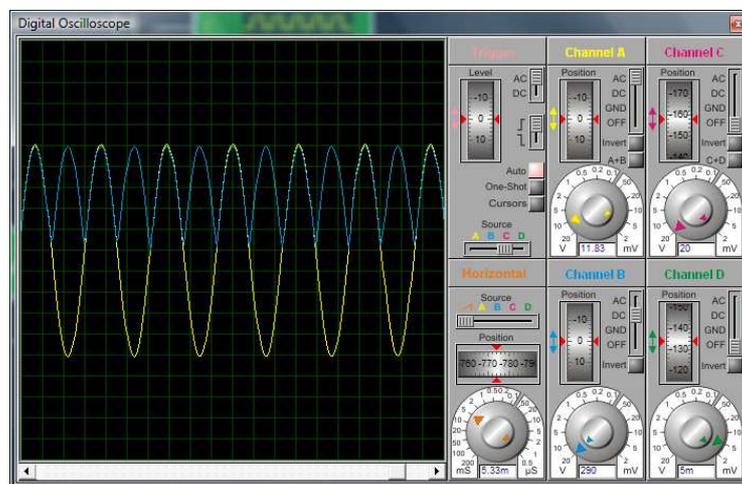


Figura III.61. b) Osciloscopio digital simulado muestra la onda rectificada de onda completa¹⁰.

La figura III.60 muestra un rectificador en puente, que es muy similar a un rectificador de onda completa porque genera una tensión de salida de onda completa. Los diodos D1 y D2 conducen en el semiciclo positivo y los diodos D3 y D4 conducen en el semiciclo negativo. Como resultado, la corriente por la carga rectificadora circula durante ambos semiciclos.

Durante ambos semiciclos, la tensión en la carga tiene la misma polaridad y la corriente de carga circula en la misma dirección. El circuito ha transformado la tensión de entrada alterna en una señal de salida continua pulsante, como se muestra en la figura III.58 (f).

Observe la ventaja de esta rectificación de onda completa sobre la versión con conexión central de la figura III.59. En donde se puede emplear toda la tensión del secundario.

Dado que un rectificador en puente produce una salida de onda completa, las ecuaciones para el valor medio y frecuencia de salida son las mismas que para el rectificador de onda completa:

¹⁰ Software ISIS 7.1 Pro.

$$V_{DC} = \frac{2V_p}{\pi} \quad y \quad f_{sal} = 2f_{ent}$$

El valor medio es el 63.6% del valor pico y la frecuencia de salida es 120Hz para una frecuencia de red de 60Hz.

Una ventaja del rectificador en puente es que toda la tensión del secundario se usa como entrada al rectificador. Dado el mismo transformador, obtenemos el doble de la tensión pico y el doble de la tensión continua con un rectificador en puente que con un rectificador de onda completa. Duplicar la tensión de salida continua compensa el uso de dos diodos extra. Por regla general verá el rectificador puente en muchas más aplicaciones que el rectificador de onda completa.

El rectificador de onda completa se usó durante muchos años antes de que apareciera el rectificador en puente. Por esta razón, ha mantenido el nombre de rectificador de onda completa incluso aunque el rectificador en puente tiene una salida de onda completa. Para distinguir el rectificador de onda completa del rectificador en puente, en algunos textos el rectificador de onda completa se denomina rectificador de onda completa convencional, rectificador de onda completa de dos diodos o rectificador de onda completa con conexión central.

Finalmente se muestra en la figura III.62 un dispositivo que encapsula los cuatro diodos¹¹ en puente llamado KBU.



Figura III.62. Encapsulado KBU de 4 diodos en puente.

Como recomendación de esta sección sería implementar un laboratorio real y usar cualesquier de los circuitos planteados y probar toda esta teoría y simulación. De esta forma conjugaremos la enseñanza teórica que hoy es asistida por computador como es el caso de software como MATLAB y la electrónica práctica, herramientas útiles hardware y software para la formación de un ingeniero en electrónica mejor capacitado.

CONCLUSIONES Y RECOMENDACIONES DE LA PRÁCTICA.

¹¹ Un solo diodo común es el 1N4001 que sirve para propósitos de rectificación.

Se concluye que realizando modelos en Simulink se puede analizar las ondas generadas por dispositivos reales, tales como un puente de diodos.

Se recomienda usar otras series de Fourier para aplicarlas usando el esquema demostrado, por ejemplo una señal diente de sierra, que corresponde a la siguiente serie trigonométrica: (BLAKE, 2004)

3.6 PRÁCTICA SERIES DE FOURIER USANDO ARCHIVOS-M. SERIES DE FOURIER USANDO ARCHIVOS-M DE MATLAB

Objetivo: Probar y cambiar datos concernientes a los archivos-M presentados para entender mejor el desarrollo de las diferentes series de Fourier que generan señales típicas de sistemas electrónicos.

Se sugiere transcribir o simplemente usar estos archivos-M desde el CD entregado como anexo a esta tesis.

Ejemplo1.

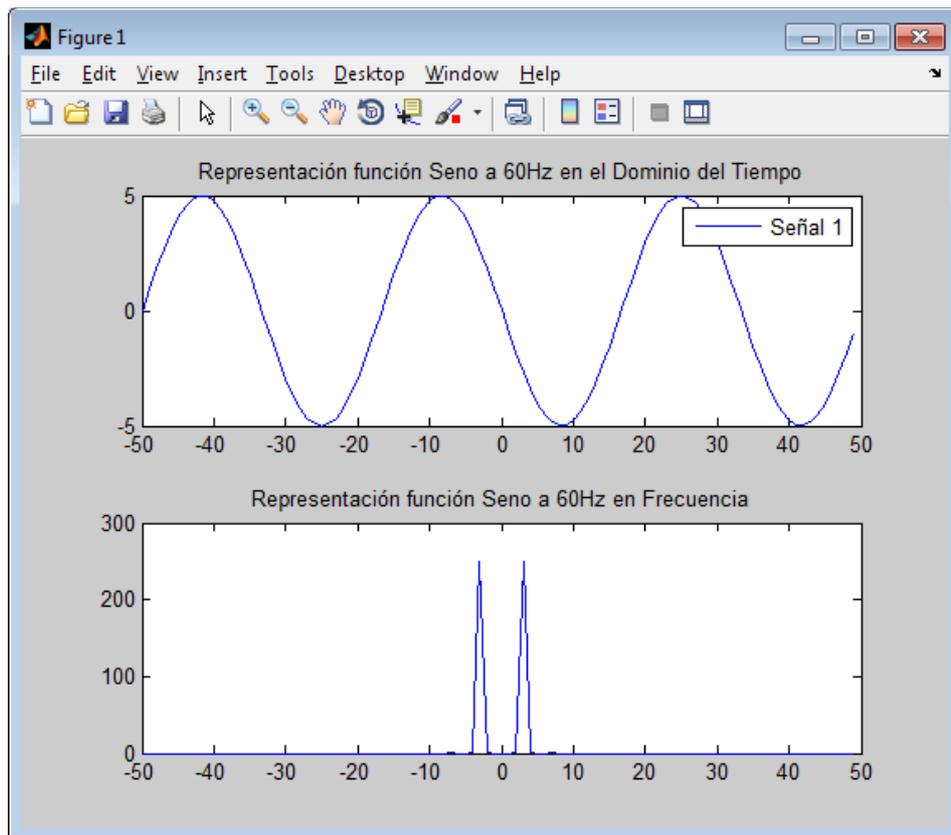


Figura III.63. Salida del archivo-M PRACTICA4_2_1.M

```
%% PRACTICA4_2_1.M  
%% Hernando Santillán
```

```
%% Tesis Práctica 4.2-Series de Fourier.  
%% Domingo, 24 de julio 2010  
clc  
clear  
T=100;      %%Período.  
t=0:T-1;    %%tiempo  
A=5;        %%amplitud pico 5V  
w0=60;      %%frecuencia Hz  
fm=2000;    %%frecuencia de muestreo.  
  
v=A*sin(2*pi*w0*t/fm); %%Función  
subplot(2,1,1),plot(-T/2:T/2-1,v);  
title('Representación función Seno a 60Hz en el Dominio del Tiempo');  
legend('Señal 1');  
subplot(2,1,2),plot(-T/2:T/2-1,abs(fftshift(fft(v)))); %%fftshift()  
sirve para centrar el espectro.  
title('Representación función Seno a 60Hz en Frecuencia');
```

Ejemplo2.

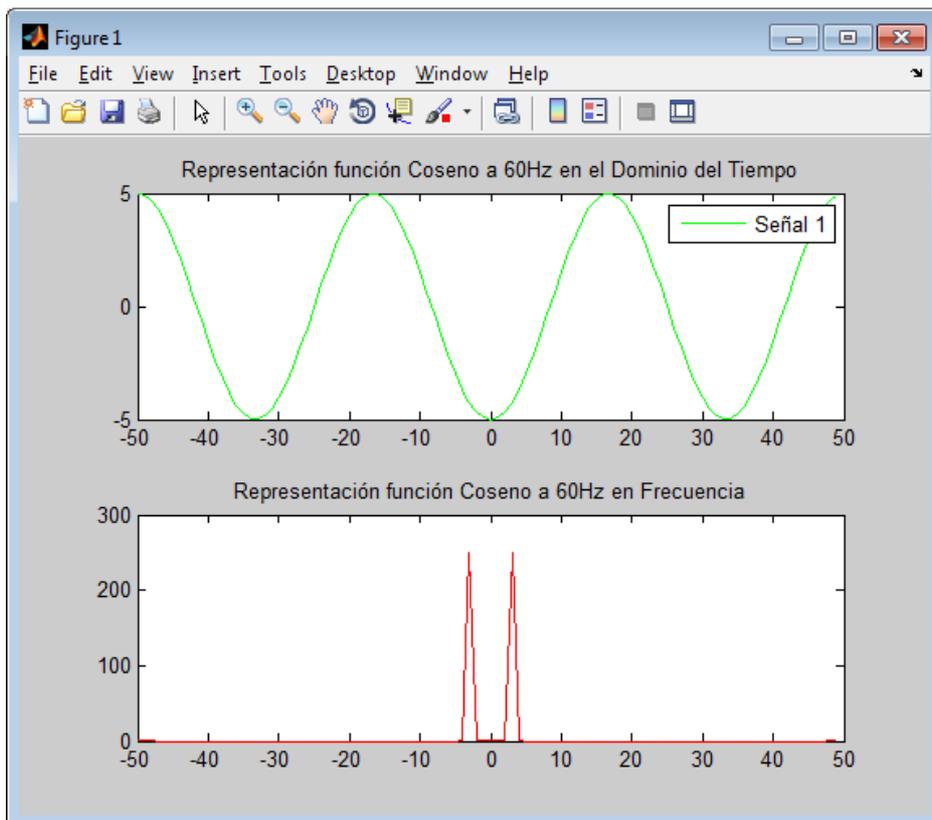


Figura III.64. Salida del archivo-M PRACTICA4_2_1.M

```
%% PRACTICA4_2_2.M  
%% Hernando Santillán
```

```
%% Tesis. Práctica 4.2. Series de Fourier
%% Sábado, 24 de julio 2010
clc
clear
T=100;      %%Período.
t=0:T-1;    %%tiempo
A=5;        %%amplitud pico 5V
w0=60;      %%frecuencia Hz
fm=2000;    %%frecuencia de muestreo.

v=A*cos(2*pi*w0*t/fm); %%Función
subplot(2,1,1),plot(-T/2:T/2-1,v,'g');
%% g muestra la gráfica en color verde.
title('Representación función Coseno a 60Hz en el Dominio del Tiempo');
legend('Señal 1');
subplot(2,1,2),plot(-T/2:T/2-1,abs(fftshift(fft(v))), 'r');
%%fftshift() sirve para centrar el espectro.
%% r muestra la gráfica en color rojo.
title('Representación función Coseno a 60Hz en Frecuencia');
```

Ejemplo3

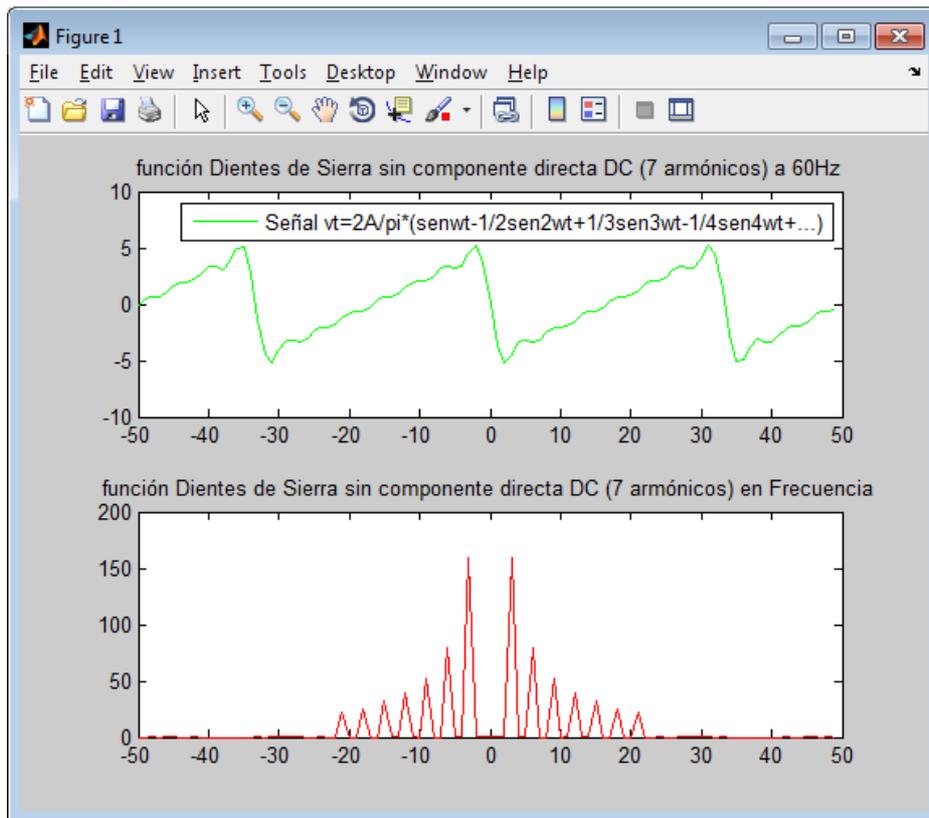


Figura III.65. Salida del archivo-M PRACTICA4_2_3.M

```
%% PRACTICA4_2_3.M
%% Hernando Santillán
```

```
%% Tesis. Práctica 4.2. Series de Fourier
%% Sábado, 24 de julio 2010
clc
clear
T=100;      %%Período.
t=0:T-1;   %%tiempo
A=5;       %%amplitud pico 5V
w0=60;     %%frecuencia Hz
fm=2000;   %%frecuencia de muestreo.
%%Señal Dientes de Sierra Sin componente directa DC.
a1=sin(2*pi*w0*t/fm);    %%primer armónico.
a2=0.5*sin(4*pi*w0*t/fm); %%segundo armónico.
a3=.33*sin(6*pi*w0*t/fm); %%tercer armónico.
a4=.25*sin(8*pi*w0*t/fm); %%cuarto armónico.
a5=.2*sin(10*pi*w0*t/fm); %%quinto armónico.
a6=.16*sin(12*pi*w0*t/fm); %%sexto armónico.
a7=.14*sin(14*pi*w0*t/fm); %%séptimo armónico.
v=(2*A/pi)*(a1-a2+a3-a4+a5-a6+a7); %%Función de dientes de sierra.
subplot(2,1,1),plot(-T/2:T/2-1,v,'g');
%% y muestra la gráfica en color amarillo.
title('función Dientes de Sierra sin componente directa DC (7 armónicos)
a 60Hz');
legend('Señal vt=2A/pi*(senwt-1/2sen2wt+1/3sen3wt-1/4sen4wt+...)');
subplot(2,1,2),plot(-T/2:T/2-1,abs(fftshift(fft(v))),'r');
%%fftshift() sirve para centrar el espectro.
%% k muestra la gráfica en color negro.
title('función Dientes de Sierra sin componente directa DC (7 armónicos)
en Frecuencia');
```

Ejemplo4

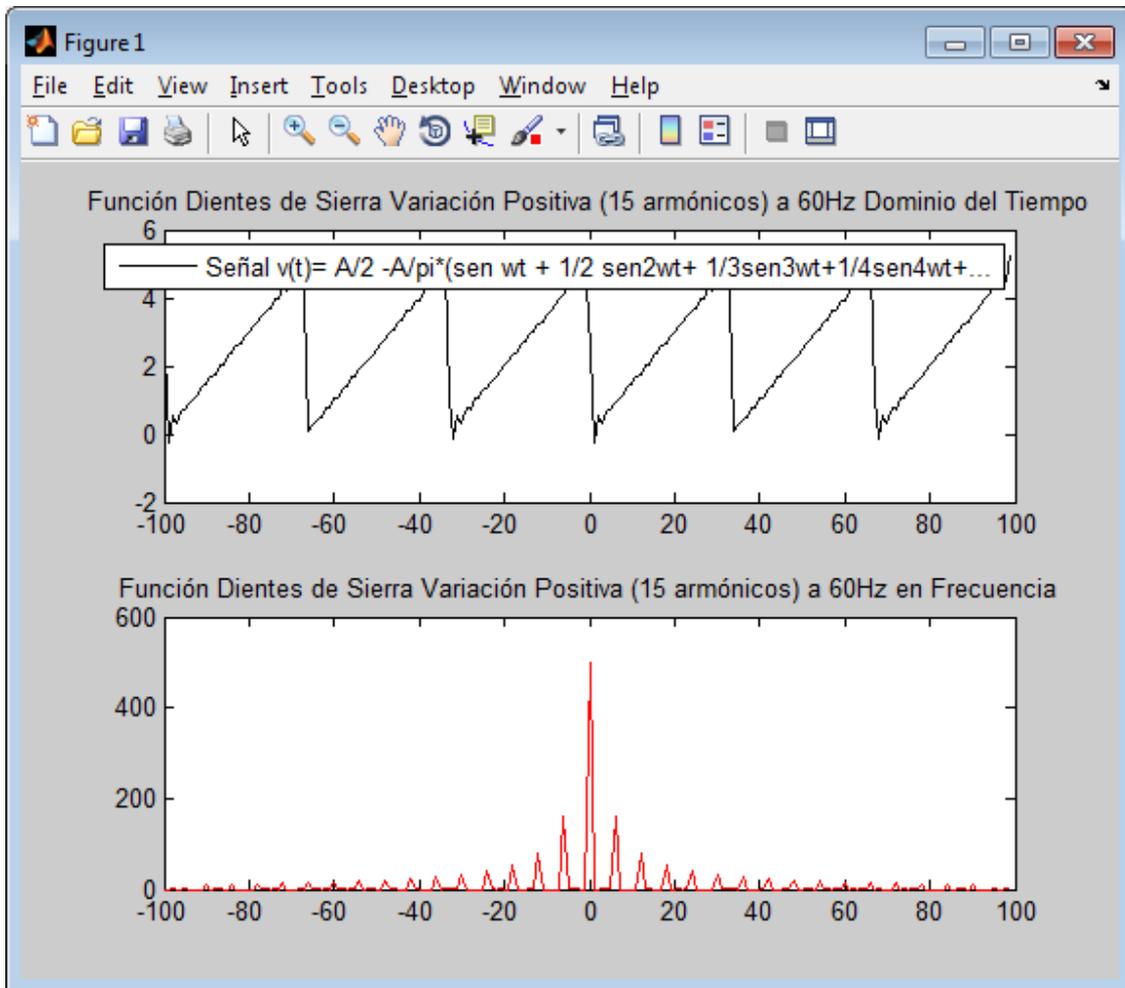


Figura III.66. Salida del archivo-M PRACTICA4_2_4.M

```
%% PRACTICA4_2_4.M
%% Hernando Santillán
%% Tesis. Práctica 4.2. Series de Fourier
%% Sábado, 24 de julio 2010
clc
clear
T=200;      %%Período.
t=0:T-1;    %%tiempo
A=5;        %%amplitud pico 5V
w0=60;      %%frecuencia Hz
fm=2000;    %%frecuencia de muestreo.

n=15; %%número de armónicos.
s=0;
for i=1:n
    armonico=1/i*sin(2*i*pi*w0*t/fm);
    s = s+armonico;
end
```

```
v=A/2-A/pi*(s); %%Función de dientes de sierra.  
subplot(2,1,1),plot(-T/2:T/2-1,v,'k');  
%% y muestra la gráfica en color amarillo.  
title('Función Dientes de Sierra Variación Positiva (15 armónicos) a 60Hz  
Dominio del Tiempo');  
legend('Señal v(t)= A/2 -A/pi*(sen wt + 1/2 sen2wt+  
1/3sen3wt+1/4sen4wt+...');  
subplot(2,1,2),plot(-T/2:T/2-1,abs(fftshift(fft(v))),'r');  
%%fftshift() sirve para centrar el espectro.  
%% k muestra la gráfica en color negro.  
title('Función Dientes de Sierra Variación Positiva (15 armónicos) a 60Hz  
en Frecuencia');  
%%
```

Ejemplo5.

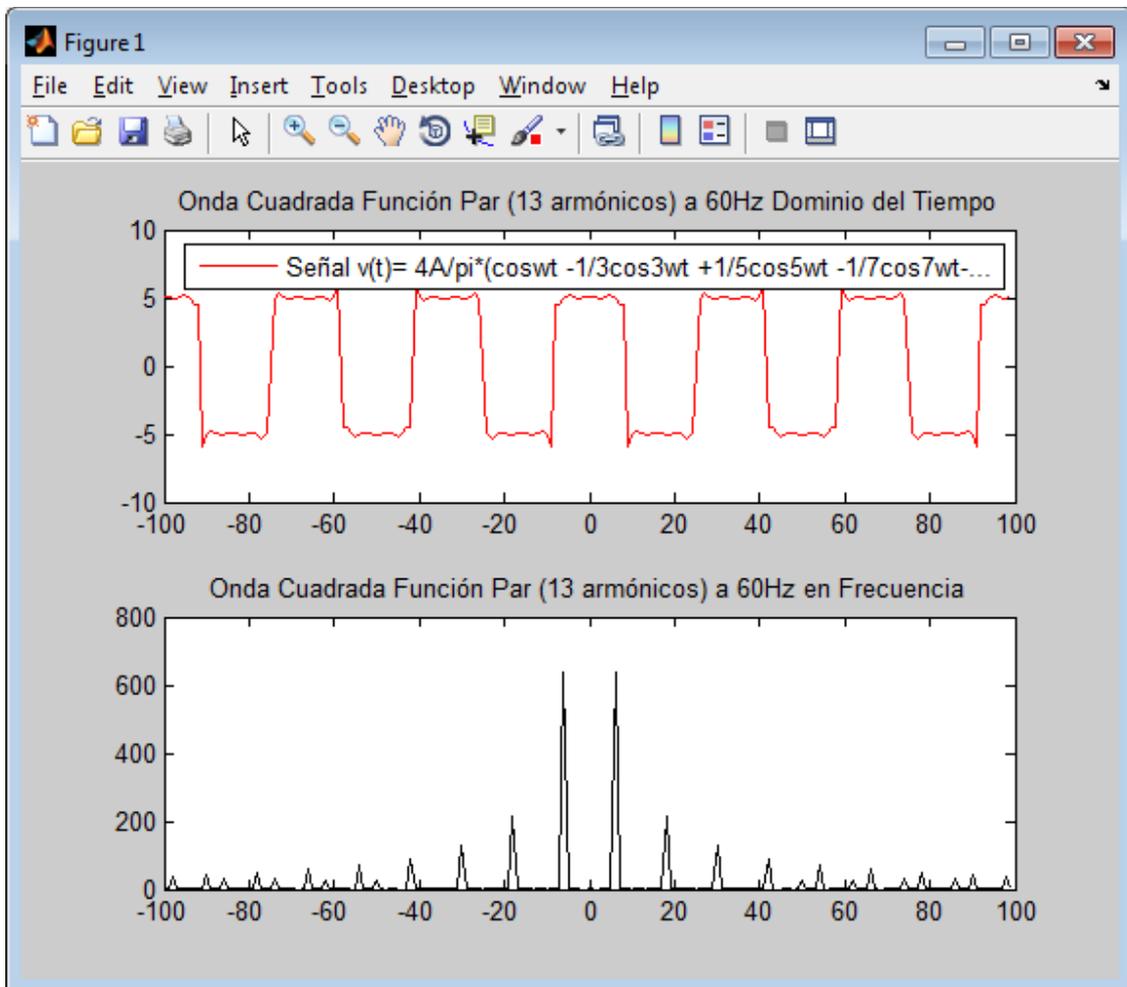


Figura III.67. Salida del archivo-M PRACTICA4_2_5.M

```
%% PRACTICA4_2_5.M  
%% Hernando Santillán  
%% Tesis. Práctica 4.2. Series de Fourier
```

```
%% Sábado, 24 de julio 2010
clc
clear
T=200;      %%Período.
t=0:T-1;    %%tiempo
A=5;        %%amplitud pico 5V
w0=60;      %%frecuencia Hz
fm=2000;    %%frecuencia de muestreo.

n=25; %%número de armónicos/2. => 13 armónicos
s=0;
signo=1;
for i=1:2:n %%solo toma armónicos impares
    armonico=1/i*cos(i*2*pi*w0*t/fm);
    s = s+armonico*signo;
    signo=signo*-1;
end
v=(4*A/pi)*(s); %%Función Par Onda Cuadrada.
subplot(2,1,1),plot(-T/2:T/2-1,v,'r');
%% b muestra la gráfica en color rojo.
title('Onda Cuadrada Función Par (13 armónicos) a 60Hz Dominio del
Tiempo');
legend('Señal v(t)= 4A/pi*(coswt -1/3cos3wt +1/5cos5wt -1/7cos7wt-...');
subplot(2,1,2),plot(-T/2:T/2-1,abs(fftshift(fft(v))),'k');
%%fftshift() sirve para centrar el espectro.
%% k muestra la gráfica en color negro.
title('Onda Cuadrada Función Par (13 armónicos) a 60Hz en Frecuencia');
%%
```

Ejemplo6

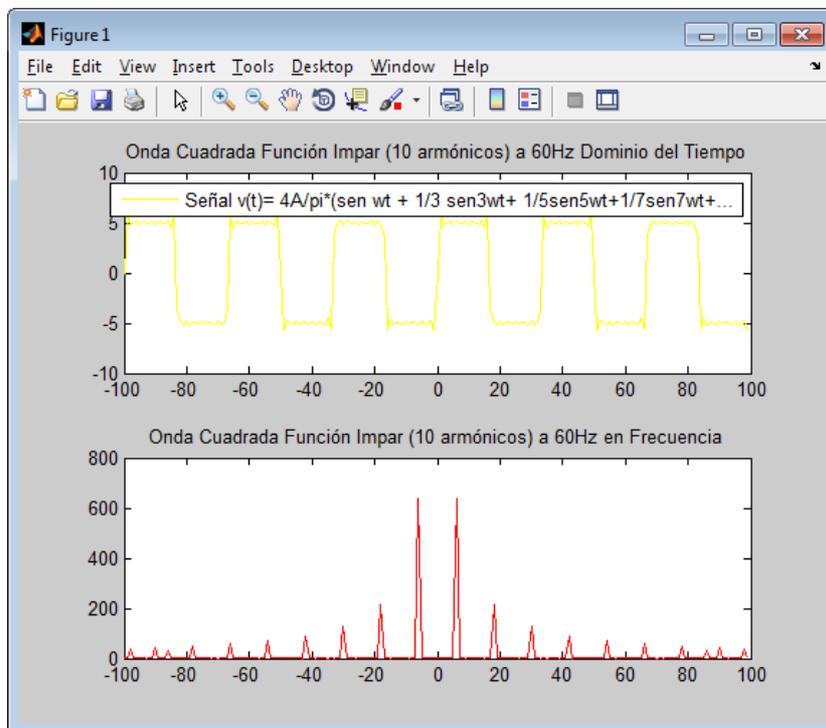


Figura III.68. Salida del archivo-M PRACTICA4_2_6.M

```
%% PRACTICA4_2_6.M
%% Hernando Santillán
%% Tesis. Práctica 4.2. Series de Fourier
%% Sábado, 24 de julio 2010
clc
clear
T=200;      %%Período.
t=0:T-1;   %%tiempo
A=5;       %%amplitud pico 5V
w0=60;     %%frecuencia Hz
fm=2000;   %%frecuencia de muestreo.

n=20; %%número de armónicos. (son 10 armónicos)
s=0;
for i=1:2:n %%van paso 2
    armonico=1/i*sin(i*2*pi*w0*t/fm);
    s = s+armonico;
end
v=(4*A/pi)*(s); %%Función impar Onda Cuadrada.
subplot(2,1,1),plot(-T/2:T/2-1,v,'y');
%% y muestra la gráfica en color amarillo.
title('Onda Cuadrada Función Impar (10 armónicos) a 60Hz Dominio del
Tiempo');
legend('Señal v(t)= 4A/pi*(sen wt + 1/3 sen3wt+
1/5sen5wt+1/7sen7wt+...');
subplot(2,1,2),plot(-T/2:T/2-1,abs(fftshift(fft(v))),'r');
%%fftshift() sirve para centrar el espectro.
%% k muestra la gráfica en color negro.
title('Onda Cuadrada Función Impar (10 armónicos) a 60Hz en Frecuencia');
%%
```

CONCLUSIONES Y RECOMENDACIONES

Se concluye que el uso de estos archivos-M o guiones programados que se entregan como un anexo a esta tesis, que sirven para un total esclarecimiento del concepto de sumas y restas de señales en el capítulo que se refiere a las series de Fourier.

El cambio en los datos concernientes al número de armónicos nos permiten ver y analizar cómo se forman estas señales a menos armónicos, menor será su resolución y detección en los osciloscopios simulados.

Se recomienda tratar de pasar estos ejercicios a modelos Simulink, tomando como referencia la práctica de laboratorio 3.6, que desarrolla una rectificación de onda completa.

3.7 PRÁCTICA TEOREMA DE MUESTREO.

TEOREMA DE MUESTREO USANDO ARCHIVOS-M DE MATLAB

Objetivo: Probar los archivos-M presentados en relación al Teorema de Muestreo, y usar señales de ingreso típicas o características como son las señales Impulso y Escalón.

Se sugiere transcribir o simplemente usar estos archivos-M desde el CD entregado como anexo a esta tesis. Además se propone hacer cambios a los diferentes scripts con el objetivo de optimizarlos en caso fuere necesario.

Marco teórico.

Teorema de Muestreo

El teorema de muestreo es uno de los de mayor utilidad porque se aplica a los sistemas digitales de comunicación, y es otra aplicación de una expansión de series ortogonales.

Teorema de Muestreo. Cualquier forma de onda física puede representarse sobre el intervalo $-\infty < t < \infty$ mediante

$$\omega(t) = \sum_{n=-\infty}^{n=\infty} a_n \frac{\sin\{\pi f_s [t - (n/f_s)]\}}{\pi f_s [t - (n/f_s)]} \quad (ec. 5.1)$$

Donde

$$a_n = f_s \int_{-\infty}^{\infty} \omega(t) \frac{\sin\{\pi f_s [t - (n/f_s)]\}}{\pi f_s [t - (n/f_s)]} dt \quad (ec. 5.2)$$

Y f_s es un parámetro al cual se le asigna algún valor conveniente mayor a cero. Aún más, si $\omega(t)$ está limitada en banda a B hercios y $f_s \geq 2B$, entonces la ecuación (5.1) se convierte en la representación de la función de muestreo, donde

$$a_n = \omega(n/f_s) \quad (ec. 5.3)$$

Esto es que, para $f_s \geq 2B$, los coeficientes de la serie ortogonal son simplemente valores de la forma de onda generados cuando se obtiene una muestra cada $1/f_s$ segundos.

La serie dada por las ecuaciones (5.1) y (5.2) es llamada algunas veces serie cardinal.

Los matemáticos han sabido de esta serie desde por lo menos el año 1915 y los ingenieros desde el descubrimiento de Shannon, quién conectó la serie con la teoría de información.

De la ecuación

$$a_n = \int_{-f_s/2}^{f_s/2} W(f) e^{+j2\pi f(n/f_s)} df \quad (ec. 5.4)$$

Obtenida luego de algunos artificios matemáticos, notamos que $W(f)$ es cero para $|f| > B$, donde $B \leq f_s/2$, los límites de la integral pueden extenderse a $(-\infty, \infty)$ sin cambiar el valor de la integral. Esta integral con límites infinitos es sólo la transformada inversa de Fourier de $W(f)$ evaluada a $t = n/f_s$. Por consecuencia, $a_n = \omega(n/f_s)$ que es igual a la ecuación (5.3).

De la ecuación (5.4) resulta obvio que la mínima velocidad de muestreo permitida para la reconstrucción de una forma de onda limitada por banda sin errores está dada por

$$(f_s)_{\min} = 2B \quad (ec. 5.5)$$

Esto se conoce como la frecuencia de Nyquist.

Ahora se examina el problema de la reproducción de una forma de onda de banda limitada utilizando N valores de muestra. Suponga que sólo está interesado en reproducir la forma de onda sobre un intervalo de T_0 segundos, como lo muestra la figura. Entonces se puede truncar la serie de funciones de muestreo de la ecuación (5.1), de forma tal que se incluyan sólo N funciones $\varphi(t)$ cuyos picos están dentro del intervalo T_0 de interés. Es decir, la forma de onda puede reconstruirse aproximadamente utilizando N muestras. La ecuación es

$$\omega(t) \approx \sum_{n=n_1}^{n=n_1+N} a_n \varphi_n(t) \quad (ec. 5.6)$$

Donde el conjunto $\{\varphi(t)\}$ está descrito por la ecuación

$$\varphi_n(t) = \frac{\sin\{\pi f_s [t - (n/f_s)]\}}{\pi f_s [t - (n/f_s)]} \quad (ec. 5.7)$$

La figura 2, muestra la forma de onda reconstruida (línea sólida), la cual se obtiene mediante la suma ponderada de formas de onda retrasadas por tiempo de tipo $(\sin x)/x$ ¹² (líneas rayadas), donde las ponderaciones son los valores de muestra $a_n = \omega(n/f_s)$ denotados por los puntos. La forma de onda está limitada en banda a B hertzios con la frecuencia de muestreo de $f_s \geq 2B$. Los valores de muestra pueden almacenarse, por ejemplo, en la memoria de una computadora digital, de tal manera que la forma puede reconstruirse en un futuro o los valores pueden transmitirse sobre un sistema de comunicación en donde se lleva a cabo la reconstrucción de la forma de onda en el lado receptor. En cualquiera de estos casos, la forma de onda puede reconstruirse a partir de los valores de muestra mediante la ecuación (5.6). Es decir, cada valor de muestra se multiplica por la función SA $(\sin x)/x$ adecuada y estas funciones $(\sin x)/x$ ponderadas se suman para generar la forma de onda original. La figura 5.2 ilustra este procedimiento. El mínimo número de valores de muestra necesarios para la reconstrucción de la forma de onda es

¹² Función SA.

$$N = \frac{T_0}{1/f_s} = f_s T_0 \geq 2BT_0 \text{ (ec.5.8)}$$

Y existen N funciones ortogonales en el algoritmo de reconstrucción. Se puede decir que N es el número de dimensiones requeridas para la reconstrucción de la aproximación a T_0 segundos de la forma de onda. (COUCH, 2008)

Desarrollo de la práctica

Ejemplo1.

Transcriba o copie los ejemplos practica5_1.m, IMPULSO.m, SA.m.

```
%% PRACTICA5_1.M
%% Hernando Santillán
%% Tesis Práctica 5-Teorema del Muestreo.
%% Lunes, 23 de agosto 2010 2149Hrs

clc;
clear;

% Genera una función onda cuadrada en función del tiempo w(t)
% que será muestreada.
t = 0:0.1:5;
w = IMPULSO(zeros(length(t),1),t,2,3);

%Genera una onda cuadrada usando el Teorema del Muestreo. Cambie
%el valor de fs (frecuencia de muestro-frequency sample), para notar
%el efecto de escoger una alta o baja frecuencia de muestreo, fs
%siempre debe ser un entero positivo, ejm: fs=8000

fs = input('Entre el valor de la frecuencia de muestreo [fs]: ');
fprintf('\n\nCalculando...por favor espere\n\n');
fs = floor(fs);
if (fs <= 0)
    fs = 1;
end;

n = 1:(5*fs);
ts = n/fs - 1/(2*fs);
a = IMPULSO(zeros(length(ts),1),ts,2,3); %Usa la función PULSO

%wa = Onda generada usando el Teorema de Muestreo (Sampling Theorem).

for (i = 1:length(t))
    for (j = 1:length(n))
        temp(j) = pi*fs*(t(i)-j/fs);
    end;
    wa(i) = a'*SA(temp); %usa la función SA.
end;

subplot(211);
plot(t,w);
```

```
xlabel('t');
ylabel('w(t)');
title('Función onda-cuadrada');

subplot(212)
plot(t,wa);
xlabel('t');
ylabel('w(t)');
title(['Teorema del Muestreo con frecuencia de muestreo [fs]:'
int2str(fs) ]);
subplot(111);
```

Copie del CD o transcriba además la función IMPULSO.m, que nos sirve como una señal de entrada característica de prueba.

```
function w = IMPULSO(w,t,tinicio,tfin)
% ARCHIVO: IMPULSO.m
% LLAMADA: w = IMPULSO(w,t,tinicio,tfin)
% Esta función crea una señal de impulso cuadrado.

% t = vector de entrada que corresponde a los valores de tiempo discreto.
% w = vector de entrada que corresponde a la señal.
% tinicio = define el extremo(esquina) izquierda del pulso.
% tfin     = define el extremo derecho del pulso.

i = length(t);
for (i = 1:length(t))
    if (t(i) >= tinicio & t(i) <= tfin)
        w(i) = 1;
    end;
end;
```

Ahora copie el archive o transcriba la función SA.m

```
function y = SA(x)
% ARCHIVO: SA.M
% LLAMADA: y = SA(x)
% Esta función calcula sen(x)/x.
% Se asume que x es un "escalar" o un vector.
x = x(:);
y = zeros(length(x),1);

for (i = 1:length(x))
    if (x(i) == 0)
        y(i) = 1;
    else
        y(i) = sin(x(i))/x(i);
    end;
end;
```

Al correr el archivo-M **practica5_1.m** se pueden cambiar los valores correspondientes a la frecuencia de muestreo fs, intente con 8000, 4000, 12000.

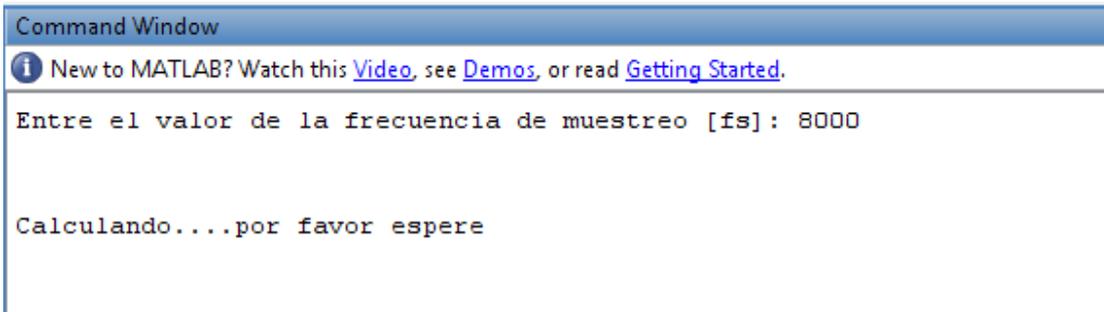


Figura III.69. Ejecución del guión-M.

A continuación se puede apreciar el gráfico resultante

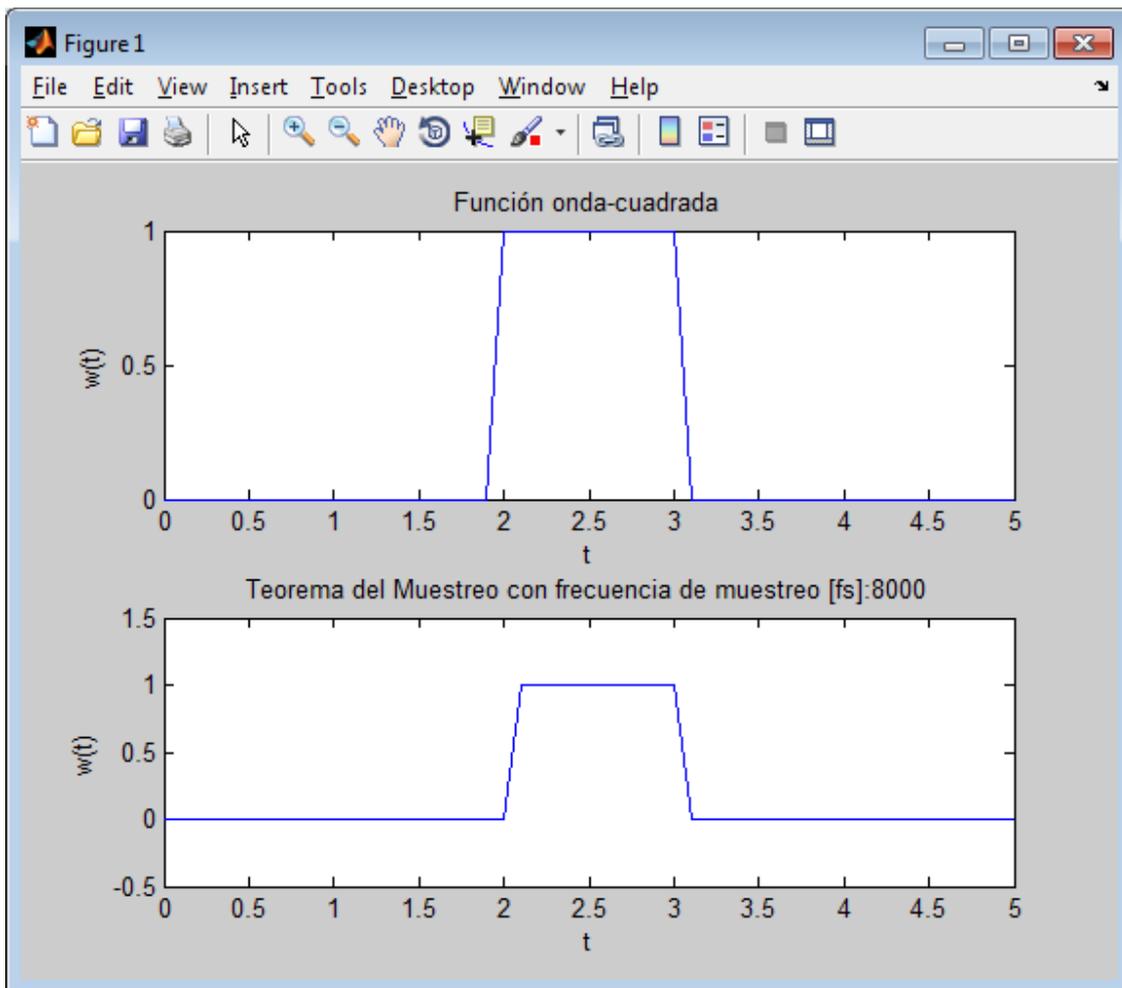


Figura III.70. Salida del guión.

Si cambiamos el valor de fs a 500

```
Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.
Entre el valor de la frecuencia de muestreo [fs]: 500

Calculando....por favor espere

>> |
```

Figura III.71. Nuevo cálculo, frecuencia de muestreo.

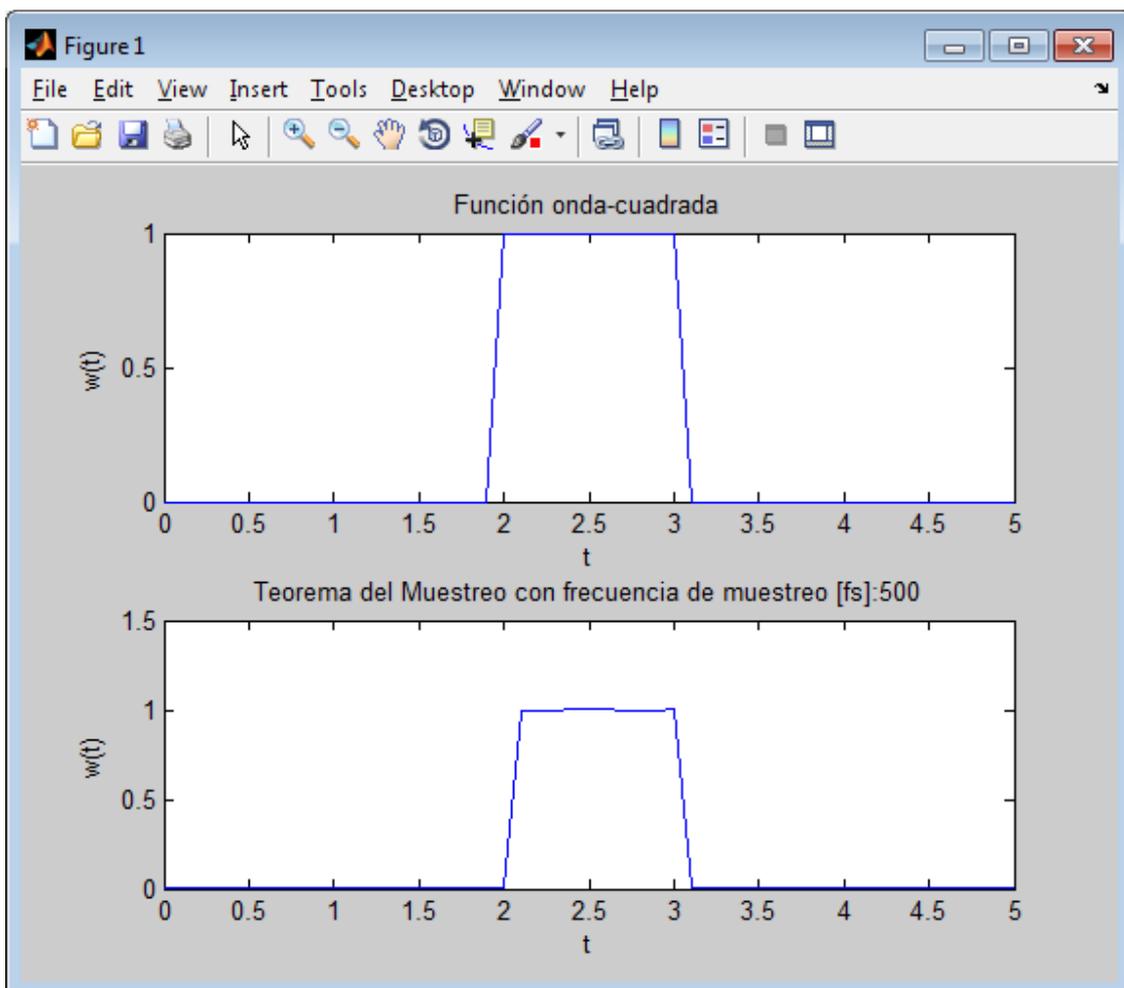


Figura III.72. Nueva salida con fm=500

Se propone hacer cambios a la frecuencia de muestreo f_s y analizar las ondas reconstruidas.

Ejemplo 2.

Copie del CD de prácticas anexo a esta tesis el archivo **practica5_2.m** o si prefiere transcribalo, se propone en cada uno de los guiones poder ver la forma de optimizarlos, y que sean lo más claros y explicativos posibles del marco teórico planteado.

```
%% PRACTICA5_2.M
%% Hernando Santillán
%% Tesis Práctica 5-Teorema del Muestreo.
%% Lunes, 23 de agosto 2010 2218Hrs

clc;
clear;

M = 8;
N = 2^M;
T = 40;
dt = T/N;
n = 0:1:N-1;
tk = n*dt -10;

% Generando una señal de onda PULSO.
w = ESCALON(tk,1) - ESCALON(tk,5);

% Aproximación de la Integral de Fourier usando la función FFT.
W = dt*fft(w);
fn = n/T;
fs = 1/dt;

subplot(211);
plot(tk,w);
title('Onda en función del Tiempo');
xlabel('tk');
ylabel('w(tk)');

% Dibujando la salida de respuesta hasta el 4to armónico.
pos = INDICE(fn,1);
subplot(212);
plot(fn(1:pos),abs(W(1:pos)));
title('MAGNITUD DEL ESPECTRO hasta el 4to. armónico. ');
xlabel('fn');
ylabel('Magnitud de W');
fprintf('\nVea la ventana para el dibujo.\n');
```

Además copie o transcriba las funciones ESCALON.m e INDICE.m

```
function y = ESCALON(t,Tinicio)
% ARCHIVO: ESCALON.M
% LLAMADA: y = ESCALON(t,Tinicio)
% Este archivo crea un escalón unitario (señal de prueba), para un
% vector tiempo "t" que empieza con un tiempo de entrada
% t=Tinicio

n = length(t);
i = 1;
j = 0;

temp = 0;
while(temp == 0)
    if (t(i) >= Tinicio)
        temp = 1;
        j = i;
    else
        i = i+1;
        if (i > n)
            temp = 1;
        end;
    end;
end;

y = zeros(n,1);
if (j > 0)
    for (i = j:1:n)
        y(i) = 1;
    end;
end;
```

Y ahora la función INDICE.m

```
function pos = INDICE(x,valor)
% ARCHIVO: INDICE.M
% LLAMADA: pos = INDICE(x,valor)
% Esta función retorna el índice del vector x que es igual al valor.
% que es, x(pos) = valor

i = 1;
j = -1;
x = x(:);
temp = 0;
while (temp == 0)
    if (x(i) == valor)
        j = 1;
        temp = 1;
    else
        i = i+1;
        if (i > length(x))

```

```
temp = 1;  
end;  
end;  
end;  
  
if (j ~= -1)  
    pos = i;  
else  
    pos = -1;  
end;
```

Al correr el guión o archivo-M practica5_2.m se tiene el siguiente resultado

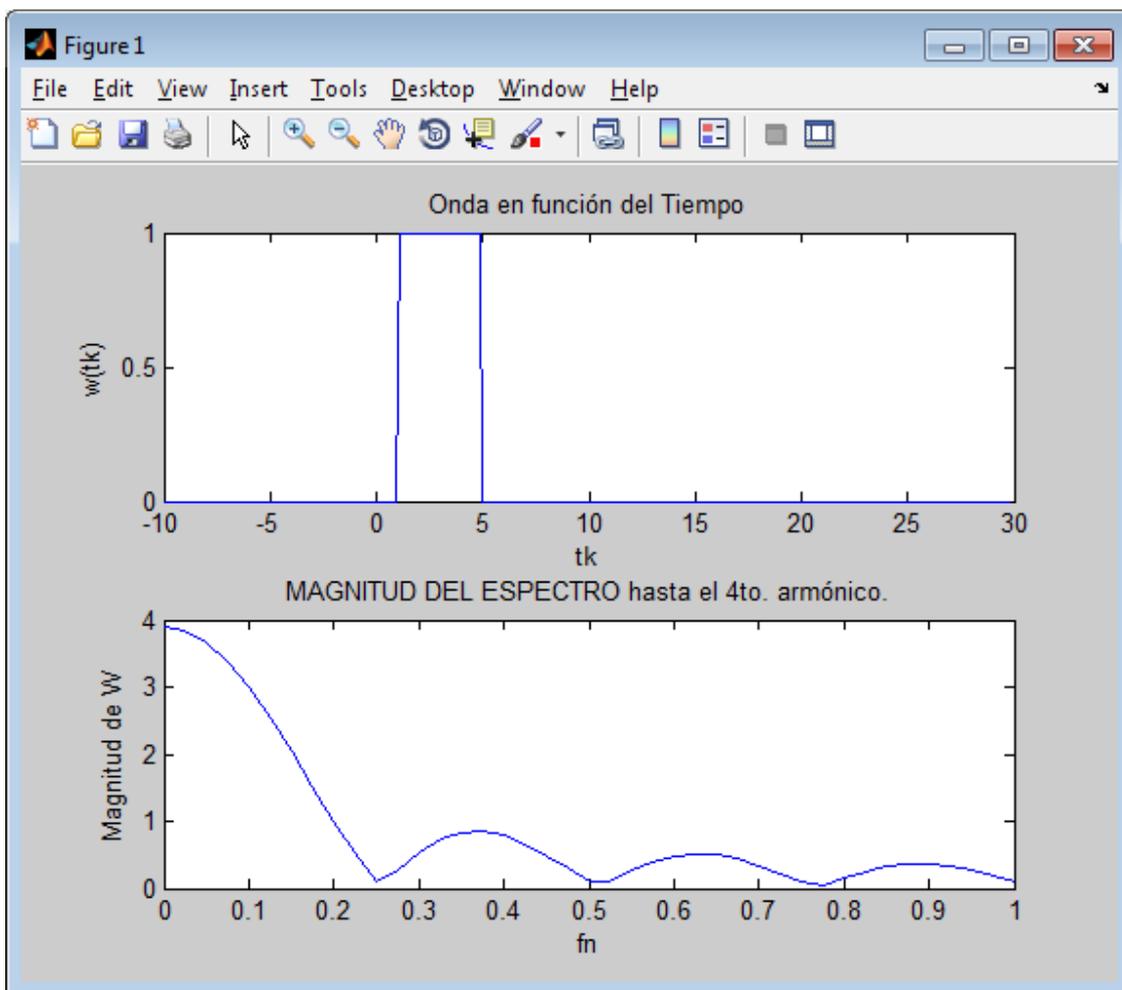


Figura III.73. Respuesta en tiempo y frecuencia.

CONCLUSIONES Y RECOMENDACIONES

Con la siguiente práctica se puede comparar los conocimientos matemáticos revisados en el marco teórico con las señales que han producido los archivos-M mostrados, en lo que se refiere al Teorema de Muestreo.

Se recomienda tratar de optimizar los archivos-M de ser posible, tratando de crear otras funciones características de entrada, por ejemplo una señal RAMPA o señales compuestas.

3.8. PRÁCTICA FILTRO PASABANDA SIMULINK. FILTRO PASABANDA RLC EN EL DOMINIO S REPRESENTADO CON SIMULINK

Objetivo: Realizar la comprobación de los pasos mencionados para crear el modelo Simulink que representa un filtro pasabanda y cambiar los coeficientes de la función de transferencia para luego analizar la gráfica correspondiente.

Se sugiere seguir los pasos indicados en la práctica para construir el modelo Simulink que representa una función de transferencia en el dominio s de la frecuencia correspondiente a un filtro pasabanda o usarlo directamente del CD entregado como anexo a esta tesis **PRACTICA6_1.MDL**. Además se propone hacer cambios en lo que se refiere a la función de transferencia planteada.

Marco teórico.

La transformada de Laplace en el análisis de circuitos

La transformada de Laplace tiene dos características que la convierten en una atractiva herramienta para el análisis de circuitos. En primer lugar transforma el conjunto de ecuaciones diferenciales de coeficientes constantes en un conjunto de ecuaciones polinómicas lineales, que son más fáciles de manipular. En segundo lugar, introduce automáticamente en las ecuaciones polinómicas los valores iniciales de las variables de corriente y tensión. Por tanto, las condiciones iniciales son una parte inherente del proceso de transformación (esto contrasta con el método clásico de resolución de ecuaciones diferenciales, en el que se consideran las condiciones iniciales cuando se evalúan los coeficientes desconocidos).

Elementos de circuito en el dominio de s

El procedimiento para desarrollar un circuito equivalente en el dominio s para cada elemento de circuito es sencillo. En primer lugar, escribimos la ecuación en el dominio del tiempo que relaciona la tensión en los terminales con la corriente que pasa a través de los mismos. A continuación hallamos la transformada de Laplace de la ecuación en el dominio del tiempo. Este paso nos permite obtener una relación algebraica entre la corriente y la tensión en el dominio s . Las

dimensiones de una tensión transformada son voltios-segundo, mientras que las dimensiones de una corriente transformada son amperios-segundo. Un cociente tensión-corriente en el dominio s se mide en ohmios y una admitancia se mide en siemens. Finalmente, construimos un modelo de circuito que satisfaga la relación entre la corriente y la tensión en el dominio de s .

Una resistencia en el dominio de s

En el dominio del tiempo aplicando la ley de Ohm

$$v = Ri \quad (6.1.1)$$

Puesto que R es una constante, la transformada de Laplace de la ecuación 6.1 es

$$V = RI \quad (6.1.2)$$

Donde

$$V = \mathcal{L}\{v\} \quad e \quad I = \mathcal{L}\{i\}$$

La ecuación 6.1.2 indica que el circuito equivalente de una resistencia en el dominio de s es simplemente una resistencia de R ohmios a la que atraviesa una corriente I amperios-segundo y en la que cae una tensión V voltios-segundo.

Una bobina en el dominio de s

La ecuación en el dominio del tiempo que relaciona la tensión en dos terminales con la corriente que atraviesa los mismos es

$$v = L \frac{di}{dt} \quad (6.1.3)$$

La transformada de Laplace de la ecuación 6.3 nos da

$$V = L[sI - i(0^-)] = sLI - LI_0 \quad (6.1.4)$$

Un condensador en el dominio s

Un condensador inicialmente cargado tiene circuitos equivalentes en el dominio s . Un condensador cargado inicialmente a V_0 voltios. La corriente que pasa por los terminales es

$$i = C \frac{dv}{dt} \quad (6.1.5)$$

Transformando esta ecuación a Laplace se tiene

$$I = sCV - CV_0 \quad (6.1.6)$$

Estas ecuaciones se pueden representar por una función de transferencia correspondiente a un circuito RLC pasivo equivalente en el dominio de la frecuencia s .

$$H(s) = \frac{(R/L)s}{s^2 + (R/L)s + (1/LC)} \quad (6.1.7)$$

Con una frecuencia central

$$\omega_0 = \sqrt{\frac{1}{LC}} \quad (6.1.8)$$

Desarrollo de la práctica.

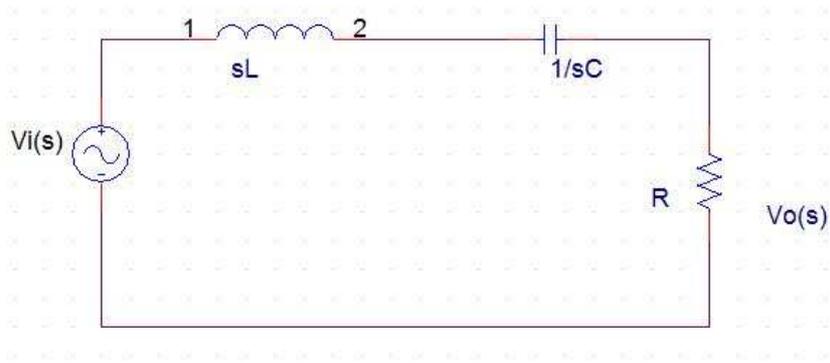


Figura III.74. Circuito RLC representado en el dominio s .

Partiendo de la función de transferencia $H(s) = \frac{(R/L)s}{s^2 + (R/L)s + (1/LC)}$ de un filtro pasabanda pasivo RLC, que se lo obtiene luego de transformar una ecuación integro-diferencial del circuito RLC (figura III.74) en su modelo de frecuencia representado por su transformada de Laplace, esta función de transferencia se la puede representar con Simulink, tomando por ejemplo:

$$H(s) = \frac{10s}{s^2 + 0.1s + 1}$$

Los coeficientes del numerador y denominador deben ingresarse en el bloque de Simulink 'Transfer Fcn' que se lo obtiene de la biblioteca **Continuous**

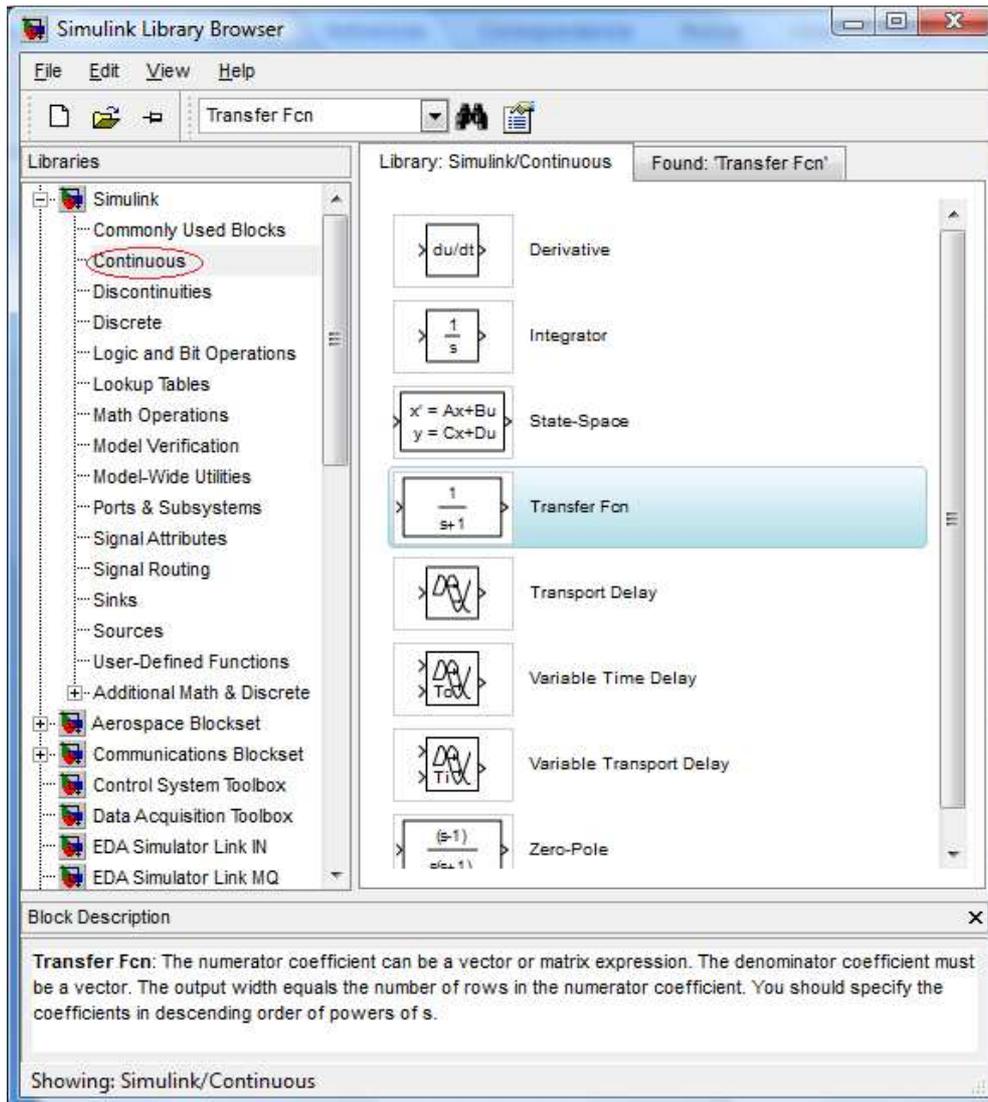


Figura III.75. Bloque Función de Transferencia dominio s.

Al arrastrar el bloque hacia un nuevo modelo, y dar doble clic sobre el mismo la siguiente información debe ser ingresada en relación a la función de transferencia y los valores de los coeficientes del numerador y denominador de la misma.

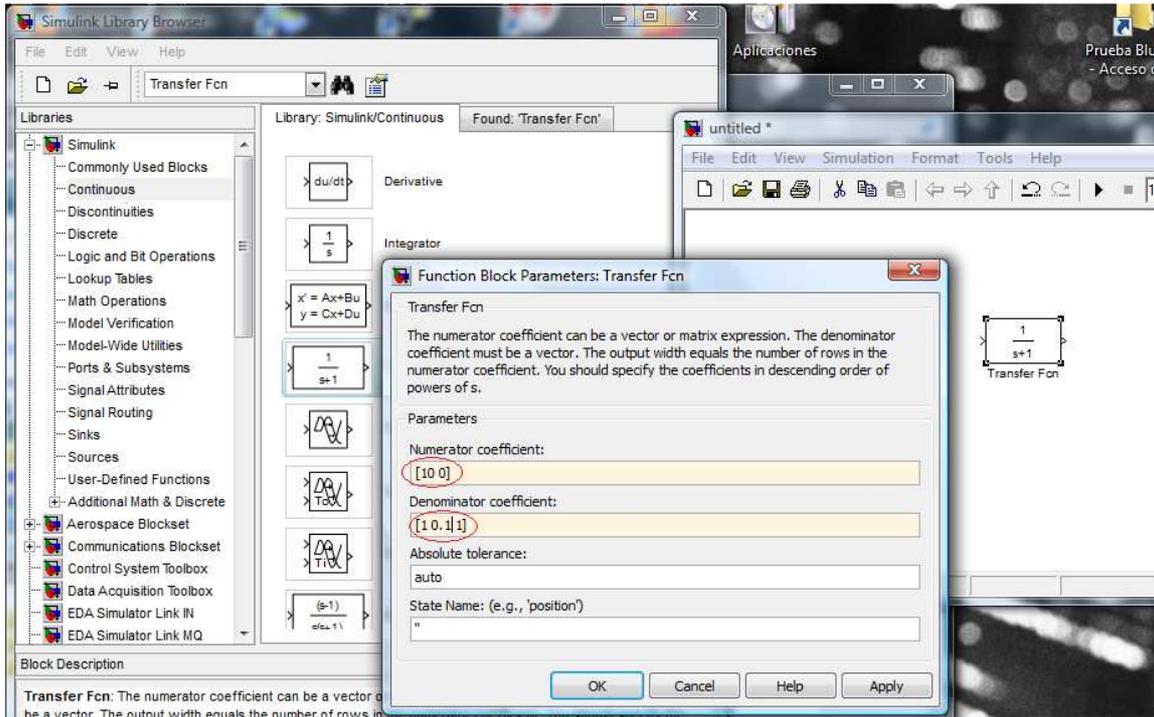


Figura III.76. Entrada de los coeficientes de la función de transferencia plateada.

El numerador $10s$, se representa en la posición de s^1 y 0 como valor de la posición s^0 .

El numerador $s^2+0.1s+1$, se representa por sus coeficientes 1 en la posición de s^2 , 0.1 en la posición de s^1 y finalmente 1 como la unidad de la posición s^0 que siempre es 1 .

Añadimos al modelo el bloque que representa una señal escalón (**STEP**), que se la obtiene de la librería **Sources**

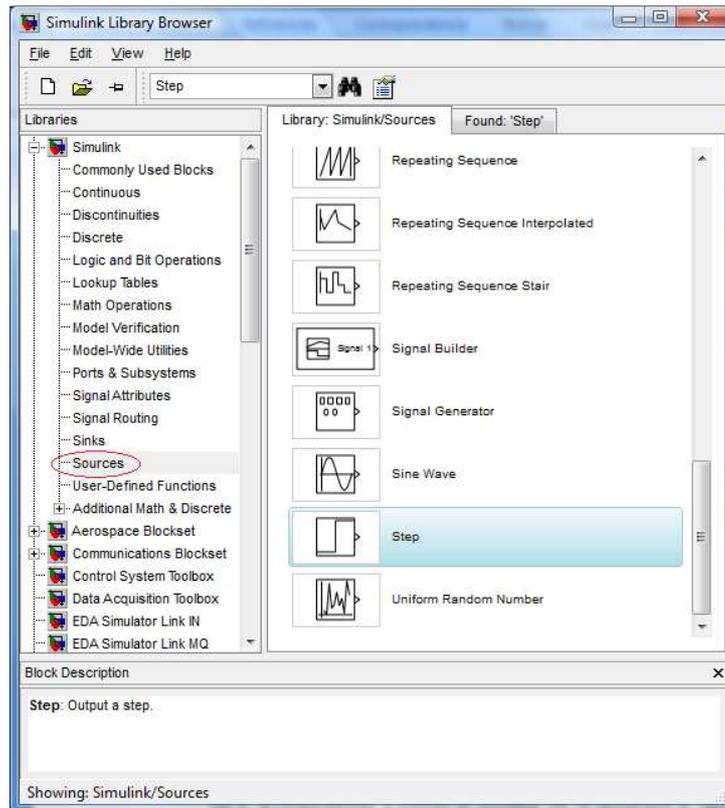


Figura III.77. Bloque escalón del Modelo.

Por lo que el modelo que representa a un filtro pasabanda RLC pasivo quedaría:

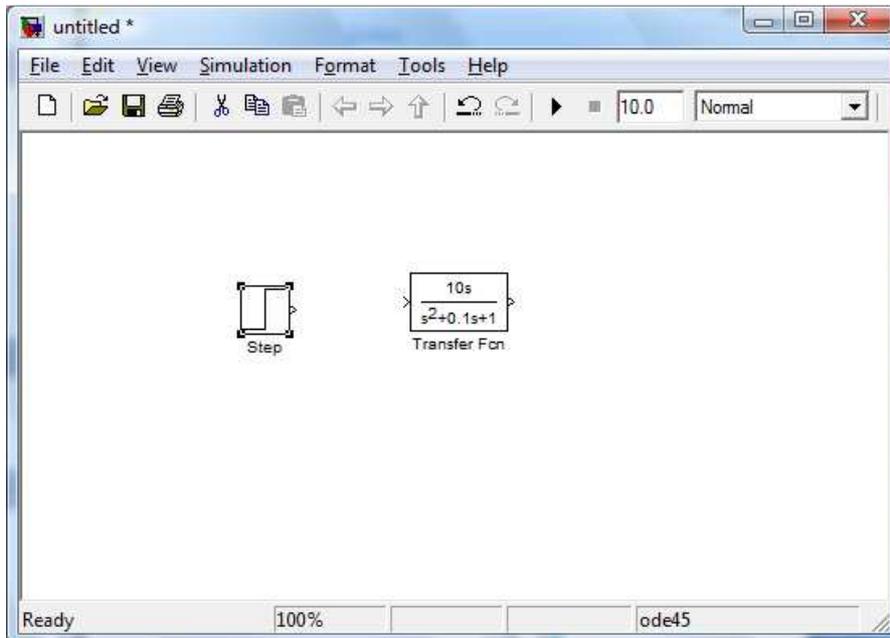


Figura III.78. Modelo en construcción con 2 bloques: Entrada y Función de Transferencia.

Finalmente se añade el bloque Osciloscopio (**Scope**) que se lo encuentra en la biblioteca **Commonly Used Blocks** (bloques comúnmente usados).

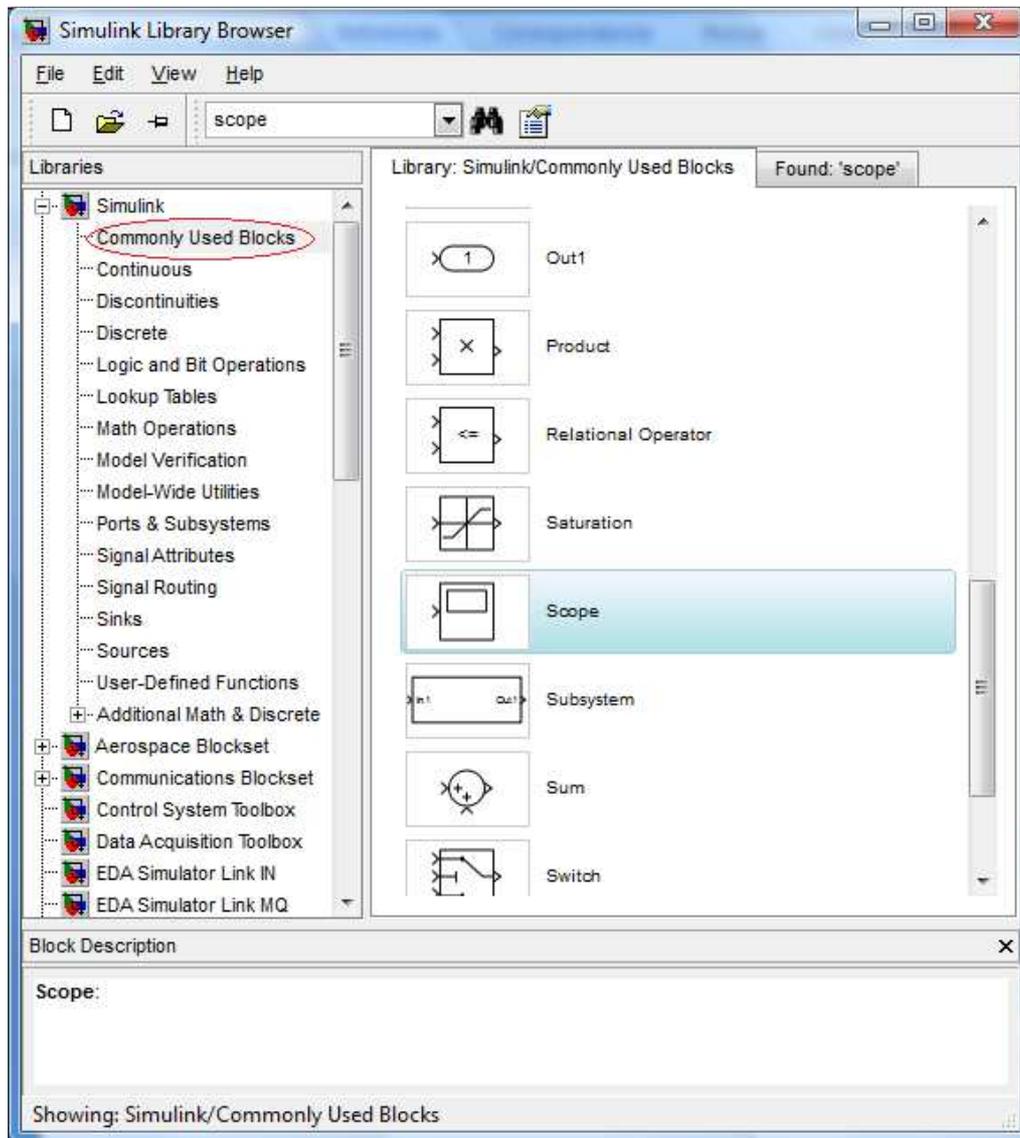


Figura III.79. Modelo Scope (Osciloscopio).

Finalmente debemos dibujar las conexiones de bloques para tener el siguiente modelo ya terminado.

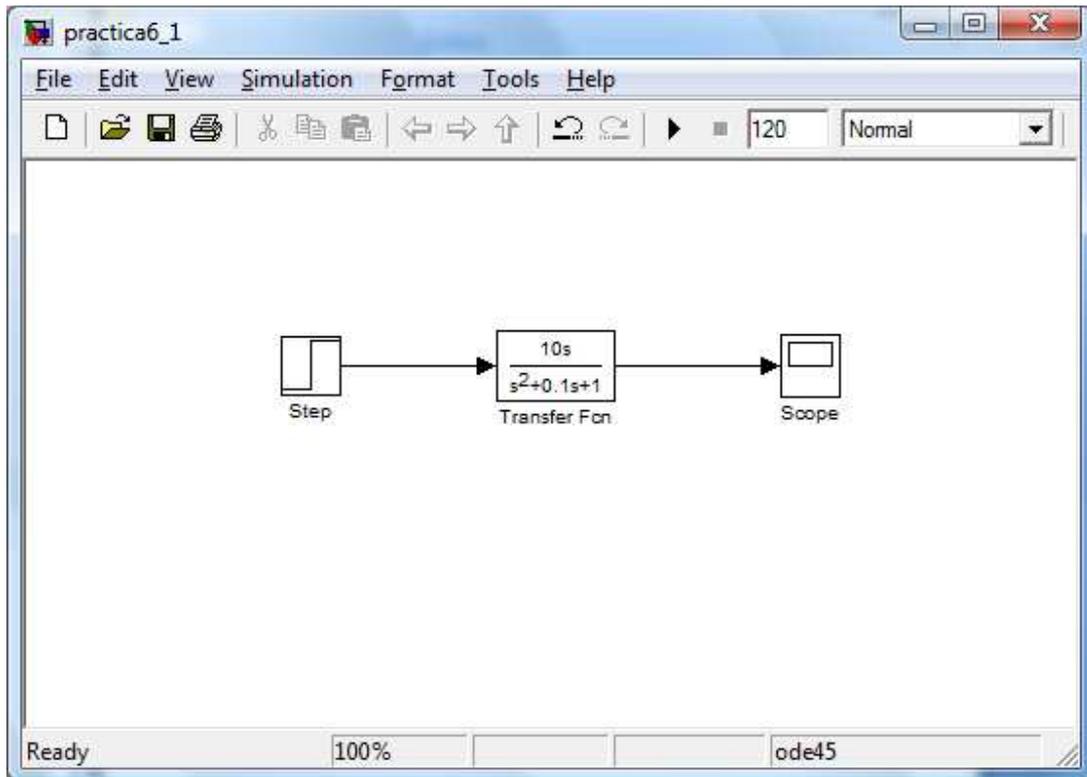


Figura III.80. Modelo Terminado.

Al correr (botón de Reproducir) el mismo modelo se tendrá una señal amortiguada en el bloque Osciloscopio, que es propia de un filtro pasabanda, como se muestra en la siguiente figura.

Se debe dar un clic derecho en Autoscale para tener una mejor apreciación de la gráfica generada por nuestra función de transferencia.

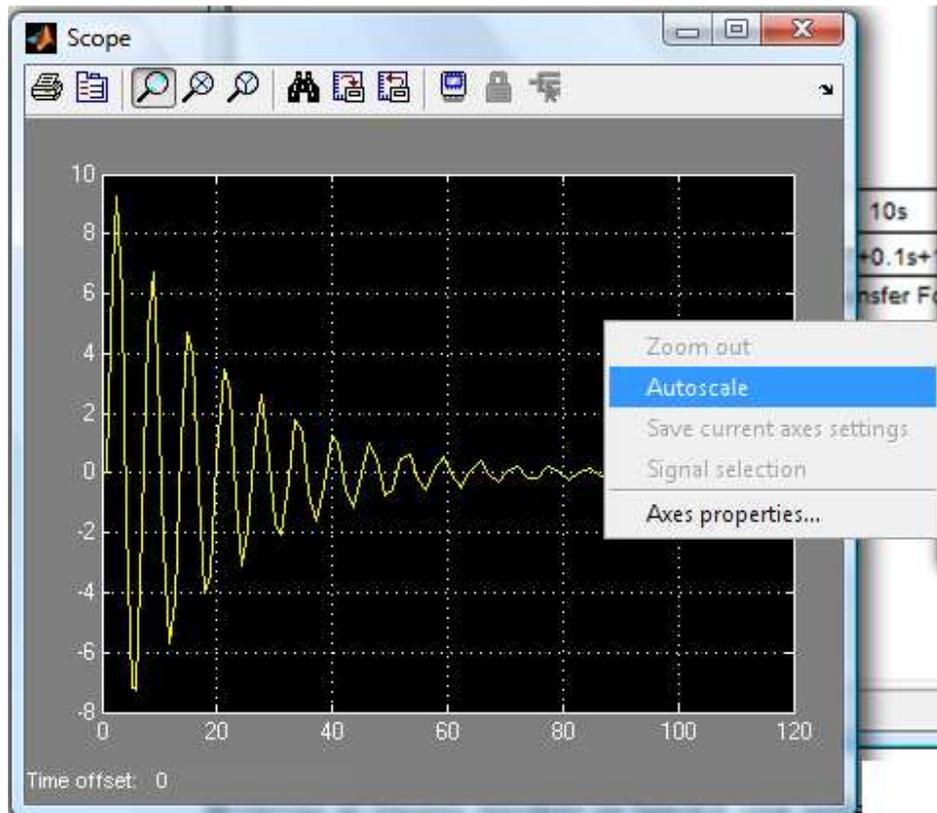


Figura III.81. Respuesta en frecuencia de un filtro RLC pasabajo pasivo.

De esta forma se puede contrastar la parte teórica y esta ayuda pictórica de simulación que nos brinda Simulink, partiendo de un circuito pasabanda RLC. (LÓPEZ, 2006)

Conclusiones y recomendaciones

Se concluye que luego de estudiar las ecuaciones en el dominio del tiempo de un circuito pasivo RLC (resistencia-bobina-condensador) que es un filtro pasabanda y transformarlas en una ecuación polinómica de Laplace, a la cual se le ha aplicado su función de transferencia característica (ecuación 6.1.7), se puede modelar usando Simulink de forma sencilla usando el bloque <<Función de Transferencia>>, al cambiar los valores de los coeficientes se pueden estudiar las señales que se generan y nos brinda un mejor entendimiento para nuestra formación como ingenieros de electrónica, que toman la materia de Análisis de Señales.

Se recomienda aplicar otras funciones de transferencia de filtros pasivos tales como **RC** (filtro pasabajo) ecuación 6.1. 9, o filtro **RL** con una ecuación de función de transferencia 6.1.10 o **CR** (filtro pasa alto) ecuación 6.1.11 con sus respectivas funciones de transferencia:

Filtro pasabajo RC

$$H(s) = \frac{1/RC}{s + 1/RC} \quad (6.1.9)$$

Con una frecuencia de corte de

$$\omega_c = 1/RC$$

Filtro pasabajo RL

$$H(s) = \frac{R/L}{s + R/L} \quad (6.1.10)$$

Con una frecuencia de corte de

$$\omega_c = R/L$$

Filtro pasalto CR

$$H(s) = \frac{s}{s + 1/RC} \quad (6.1.11)$$

Con una frecuencia de corte de

$$\omega_c = 1/RC$$

(NILSSON-RIEDEL, 2005)

Nota. Para reemplazar en el bloque Función de transferencia use sólo las ecuaciones 6.1.9-10-11, sin tomar en cuenta el valor de sus frecuencias de corte, que se han escrito solo por motivos referenciales.

3.9 PRÁCTICA RUIDO BLANCO GAUSSIANO ADITIVO.

RUIDO BLANCO GAUSSIANO ADITIVO REPRESENTADO CON SIMULINK

Objetivo: Probar un modelo Simulink que nos permite analizar e interpretar una canal de una señal generada a la cual se aplica un bloque que contiene ruido blanco gaussiano aditivo o térmico.

Se sugiere seguir los pasos indicados en la práctica para construir el modelo Simulink e intentar usar otras frecuencias de muestreo. Se puede usar el modelo entregado en el CD llamado **RUIDO_GAUSSIANO.MDL**.

Marco teórico.

Análisis de ruido

Se define al ruido eléctrico como cualquier energía indeseable que queda entre la banda de paso de la señal. Por ejemplo, en la grabación de audio se considera como ruido todas las señales eléctricas no deseadas que están dentro de la banda de frecuencias de audio, de 0 a 15 kHz, que interfieren con la música, añadiendo señales aditivas a las sinusoides sin ruido.

Se puede clasificar al ruido en dos categorías: correlacionado y no correlacionado. La correlación implica una relación entre la señal y el ruido. Por consiguiente, el ruido correlacionado sólo existe cuando hay una señal. Por otra parte, el ruido no correlacionado está presente siempre, haya o no una señal.

Ruido térmico

Este ruido se asocia con el movimiento rápido y aleatorio de los electrones dentro de un conductor, producido por la agitación térmica. Este movimiento fue observado por primera vez por Robert Brown, botánico inglés. Observó primero pruebas de la naturaleza de la materia como partículas en movimiento, en los granos de polen. El movimiento aleatorio de los electrones fue reconocido por primera vez en 1927, por J. B. Johnson de los Bell Telephone Laboratories. Los electrones en el interior de un conductor portan una carga negativa unitaria, y la velocidad cuadrática media de uno de ellos es proporcional a su temperatura absoluta. En consecuencia, cada paso de un electrón entre choques con moléculas produce un corto pulso de corriente, que produce un voltaje pequeño a través del componente resistivo del conductor. Como este tipo de movimientos del electrón es totalmente aleatorio y es en todas direcciones, el voltaje promedio

en la sustancia debido a esos movimientos es 0V CD (corriente directa). Sin embargo, ese movimiento aleatorio sí produce una componente de CA (corriente alterna).

La componente de CA debida a la agitación térmica tiene varios nombres, que incluyen al de ruido térmico, porque depende de la temperatura, también movimiento browniano, por su descubridor, ruido de Johnson, en honor de quién relacionó el movimiento browniano de las partículas con el movimiento de los electrones y, **RUIDO BLANCO**, porque el movimiento aleatorio se produce en todas las frecuencias. Por consiguiente, el ruido térmico es el movimiento aleatorio de los electrones libres dentro de un conductor, causado por la agitación térmica.

Johnson demostró que la presencia del ruido térmico es proporcional al producto del ancho de banda por la temperatura. En forma matemática, la potencia del ruido es

$$N = KTB \quad (7.1)$$

N= potencia de ruido (vatios)

B= ancho de banda (hertz)

K= constante de proporcionalidad de Botzmann (1.38e10-23 Joules por grado kelvin).

T=temperatura absoluta, en grados kelvin (la temperatura ambiente = 17° C o 290° K).

Para convertir de °C a grados kelvin sólo se suma 273°. Por consiguiente, T=°C+273°.

La potencia de ruido, expresada en dBm (decibelios referidos a 1 milivatio¹³) es una función logarítmica, igual a

$$N_{(dBm)} = 10 \log \frac{KTB}{0.01} \quad (7.2)$$

Las ecuaciones 7.1 y 7.2 indican que en el cero absoluto (0° K o -273° C) no hay movimiento molecular aleatorio, y que el producto KTB es igual a cero.

Por ejemplo 0.002W equivalen a

$$10 \log \frac{0.002}{0.001} = 3dBm$$

0.0001W

$$10 \log \frac{0.0001}{0.001} = -10dBm$$

10mW

¹³ mW (milliwatt)

$$10\log \frac{10e-3}{0.001} = 10dBm$$

0.001W o 1mW

$$10\log \frac{0.001}{0.001} = 0dBm$$

Se puede ver en estos ejemplos que las potencias mayores que 1mW producen valores positivos de dBm, y las potencias menores que 1mW producen valores negativos de dBm. Una potencia de 1mW equivale a 0 dBm.

Al reordenar la ecuación 7.2 se obtiene

$$N_{(dBm)} = 10\log \frac{KT}{0.001} + 10\log B \quad (7.3)$$

Y para un ancho de banda de 1 Hz a temperatura ambiente,

$$\begin{aligned} N_{(dBm)} &= 10\log \frac{(1.38e-23)(290)}{0.001} + 10\log 1 \\ &= -174 \text{ dBm} \end{aligned}$$

Así, la ecuación 7.2 se puede escribir, para cualquier ancho de banda a la temperatura ambiente, como sigue

$$N_{(dBm)} = -174 \text{ dBm} + 10\log B \quad (7.4)$$

El ruido aleatorio produce una densidad constante de potencia, en función de la frecuencia, y de acuerdo con la ecuación 7.1, la potencia disponible en una fuente de ruido térmico es proporcional al ancho de banda sobre cualquier intervalo de frecuencias. Esto se ha comprobado con frecuencias que van de 0 Hz hasta las máximas frecuencias de microondas que se usan en la actualidad. Así, si el ancho de banda es ilimitado, parece que la potencia disponible en una fuente de ruido térmico también es ilimitada. Eso, naturalmente, no es verdad, porque se puede demostrar que a frecuencias arbitrariamente altas la potencia de ruido térmico llega a bajar a cero.

Cómo el ruido térmico se distribuye por igual en el espectro de frecuencias, a una fuente de ruido térmico se le llama a veces fuente de ruido blanco, por ser análoga a la luz blanca que contiene todas las frecuencias de la luz visible. Por lo anterior, la potencia **rms** (efectiva) medida a cualquier frecuencia en una fuente de ruido blanco es igual a la medida en cualquier otra frecuencia de la misma fuente de ruido. De igual manera, la potencia total rms de ruido medida en cualquier ancho de banda fijo es igual a la potencia total rms de ruido, medida en un ancho de banda igual en cualquier lugar del espectro total de ruido. En otras palabras, la potencia rms de ruido blanco

presente en la banda de 1000 Hz a 2000 Hz es igual a la presente en la banda de 1,001,000 a 1,002,000 Hz.

El ruido térmico es aleatorio y continuo, y se produce en todas las frecuencias. También, es predecible, aditivo, y está presente en todos los dispositivos. Por lo anterior, el ruido térmico es el más importante de todos los ruidos. (TOMASI, 2003)

Desarrollo de la Práctica

Luego de arrancar SIMULINK, buscar en la biblioteca Communications Blockset/Comm Sources/Random Data Sources/Bernoulli Binary Generator, con este bloque se genera un vector de salida con números aleatorios.

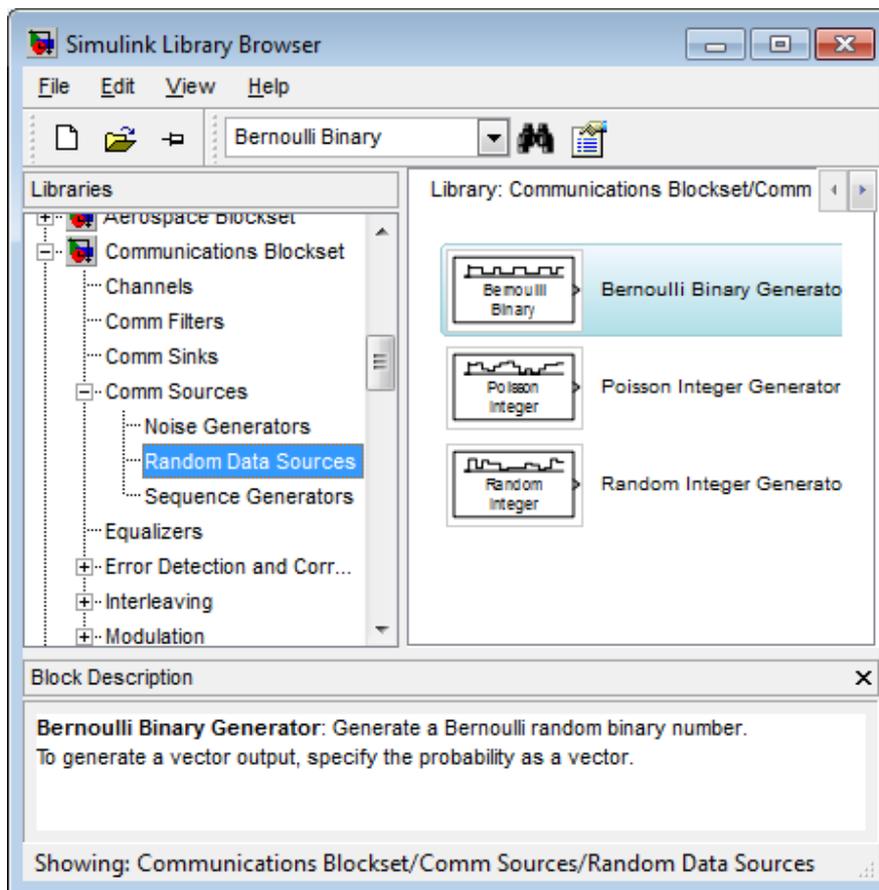


Figura III.82. Bloque Generador Binario de Bernoulli.

Al dar doble clic sobre el bloque generador de números binarios de Bernoulli se tiene la opción de cambiar sus valores en caso de ser necesario.

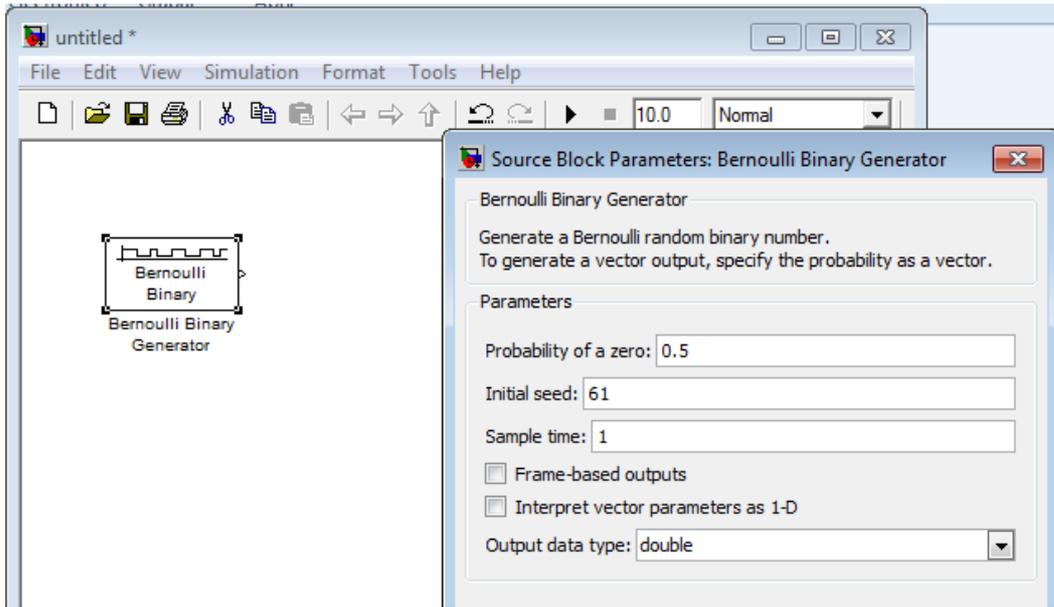


Figura III.83. Configuración Bloque Generador de Señales aleatorias.

En la misma biblioteca **Communications Blockset**, bajo **/Modulation/Digital Baseband Modulation/FM** escogemos el bloque modulador y demodulador **M-FSK** (Baseband Frequency Shift Keying), desplazamiento digital en frecuencia de banda base.

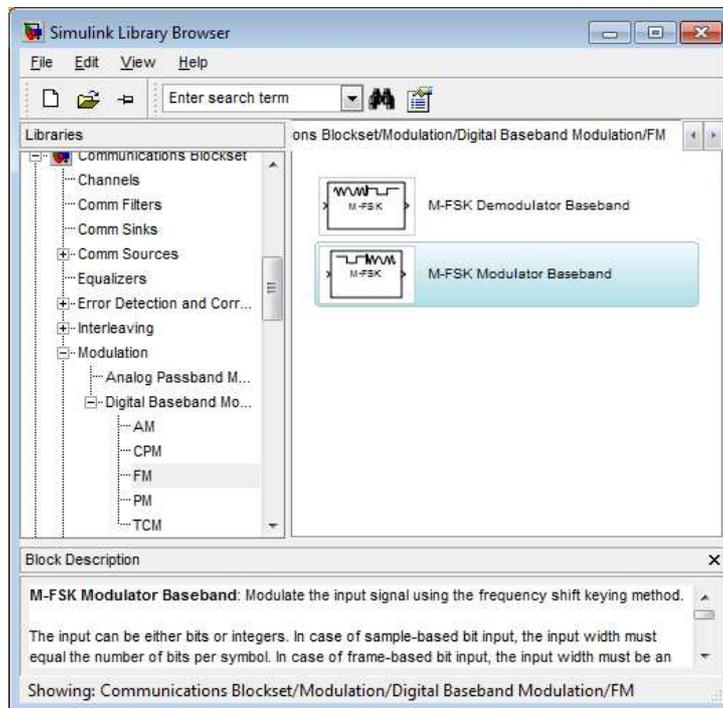


Figura III.84. Bloque M-FSK Modulador FM Digital.

Nuestro modelo hasta el momento sería como se lo muestra en la figura

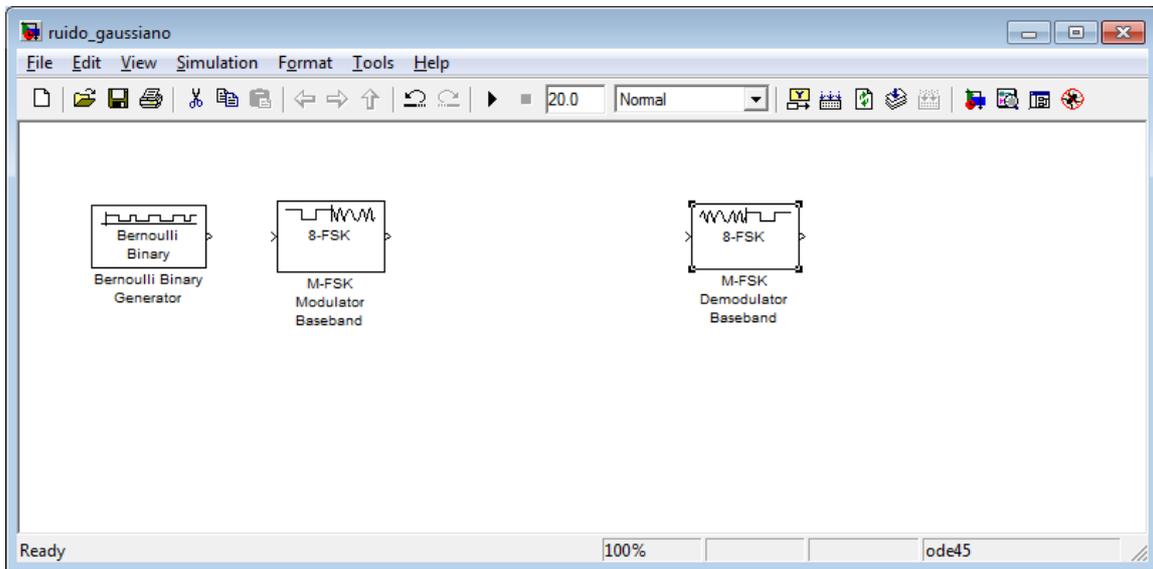


Figura III.85. Modelo actual Modulador/Demodulador FM Digital.

A este medio de comunicación digital debemos añadirle un ruido blanco gaussiano aditivo, para darle a la simulación un carácter más realista. Por lo que se añade el bloque AWGN, que se lo encuentra en la siguiente ruta de Simulink: Communications Blockset/Channels/AWGN¹⁴Channel

¹⁴ AWGN (Aditive White Gaussian Noise), Ruido blanco gaussiano aditivo.

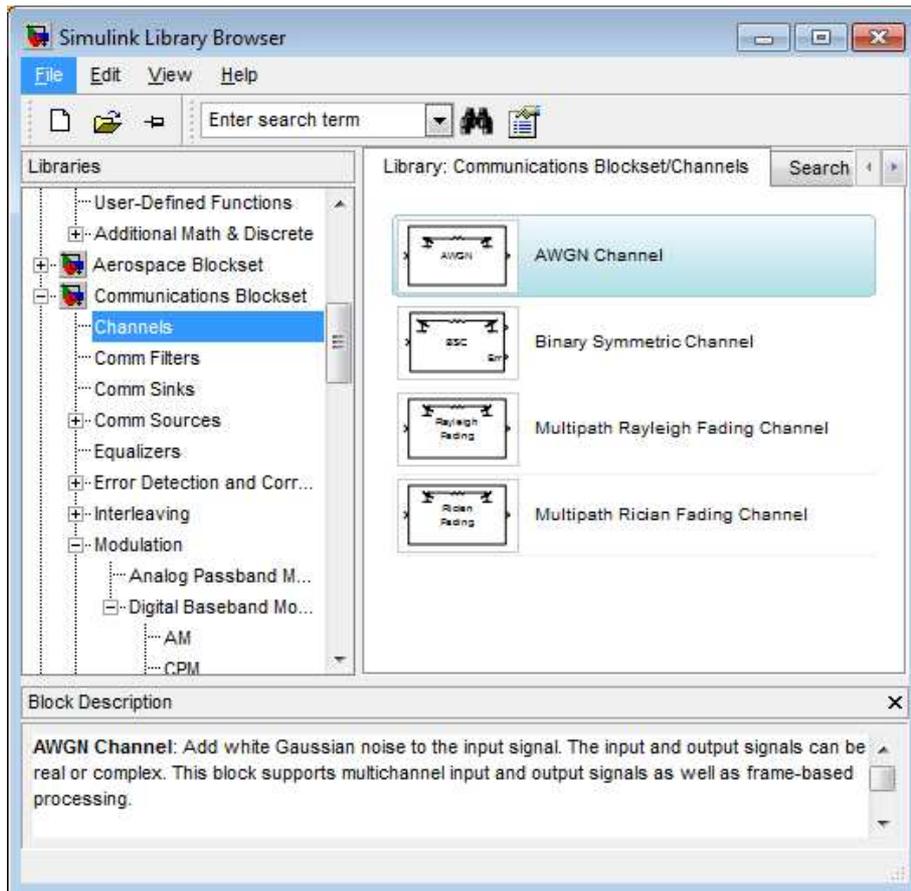


Figura III.86. Bloque AWGN (Ruido Blanco Gaussiano Aditivo o TÉRMICO).

Nuestra simulación la tendríamos de la siguiente forma:

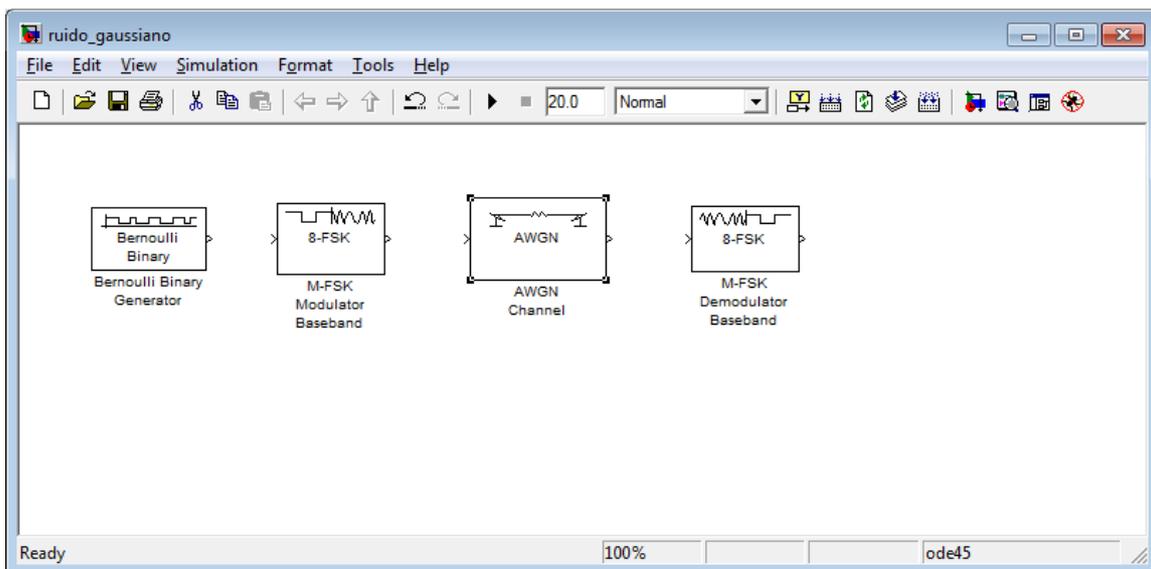


Figura III.87. Bloques de la simulación actuales.

Finalmente añadiríamos el bloque **Scope** (Osciloscopio) para poder diferenciar entre una señal sin ruido y otra a la que se le ha añadido ruido Gaussiano o randómico, que es el motivo principal de esta práctica.

Entonces se añade este bloque que se lo encuentra en la biblioteca Simulink/Commonly Used Blocks/Scope

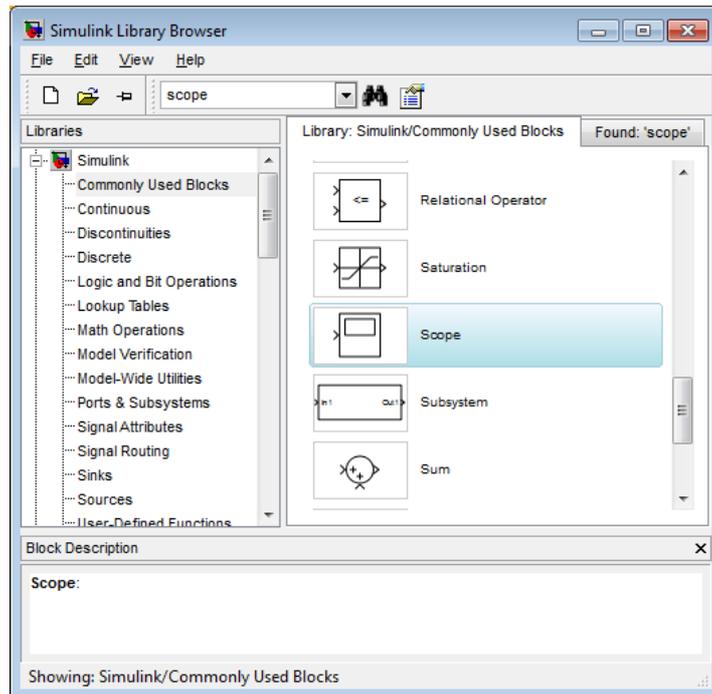


Figura III.88. Bloque Scope (Osciloscopio).

Cambiamos el número de ejes o salidas a 2, como se muestra en la figura

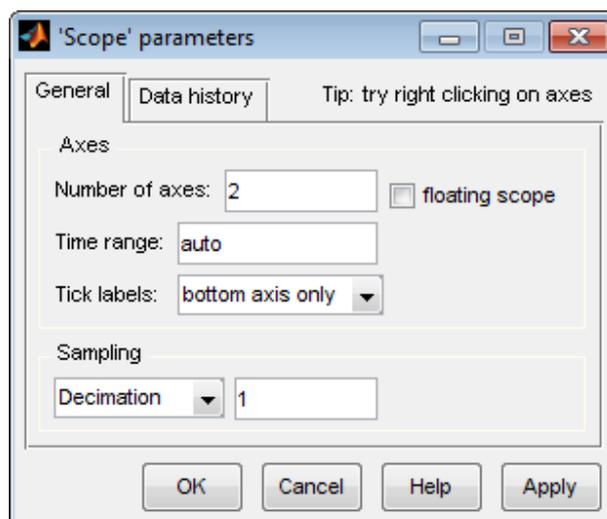


Figura III.89. Dividir Scope en 2 canales.

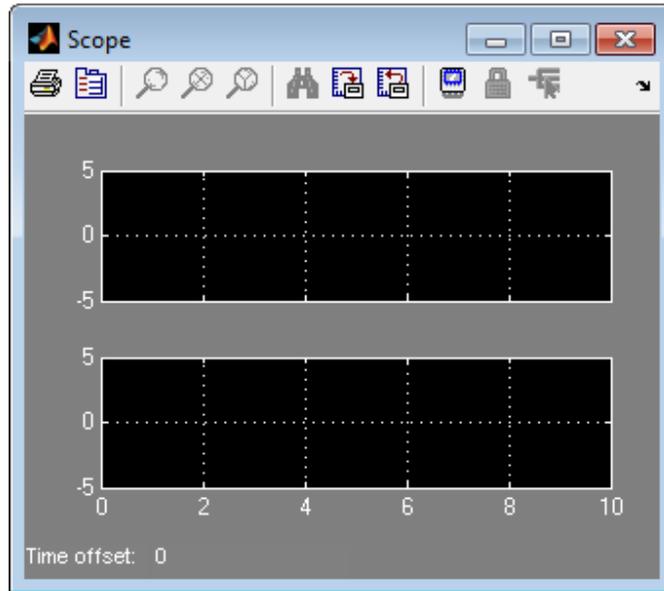


Figura III.90. Osciloscopio listo para recibir las señales.

Nuestro modelo casi terminado lo tendríamos como se muestra en la figura III.91.

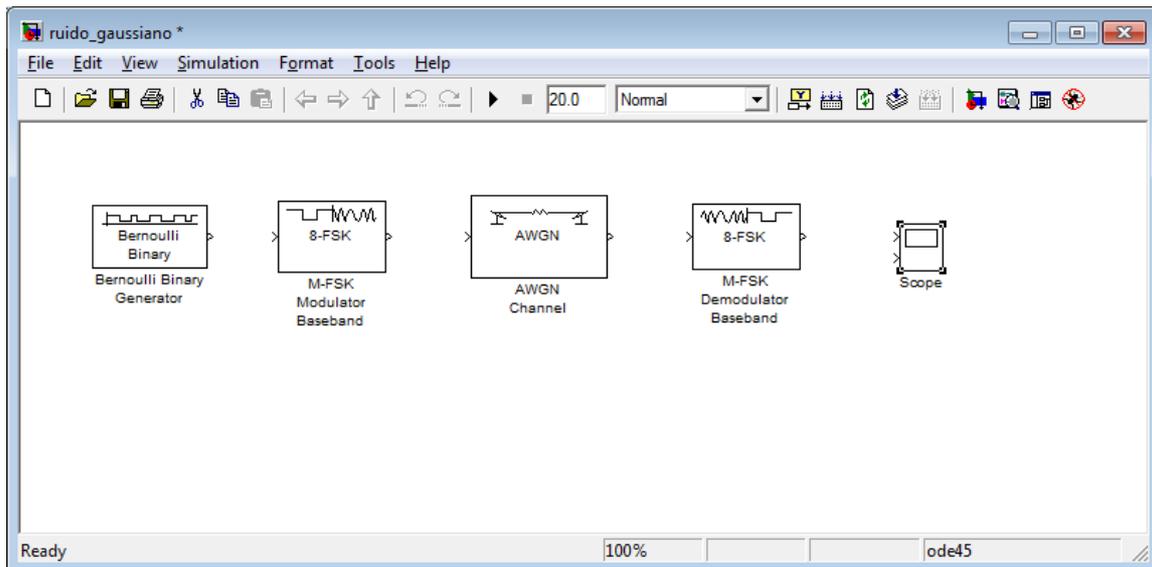


Figura III.91. Modelo actual con todos los bloques, listos a ser unidos sus canales de flujo.

Como ya lo hemos practicado en los modelos anteriores de las prácticas que anteceden a esta, el último paso consiste en unir los bloques y establecer la comunicación entre ellos.

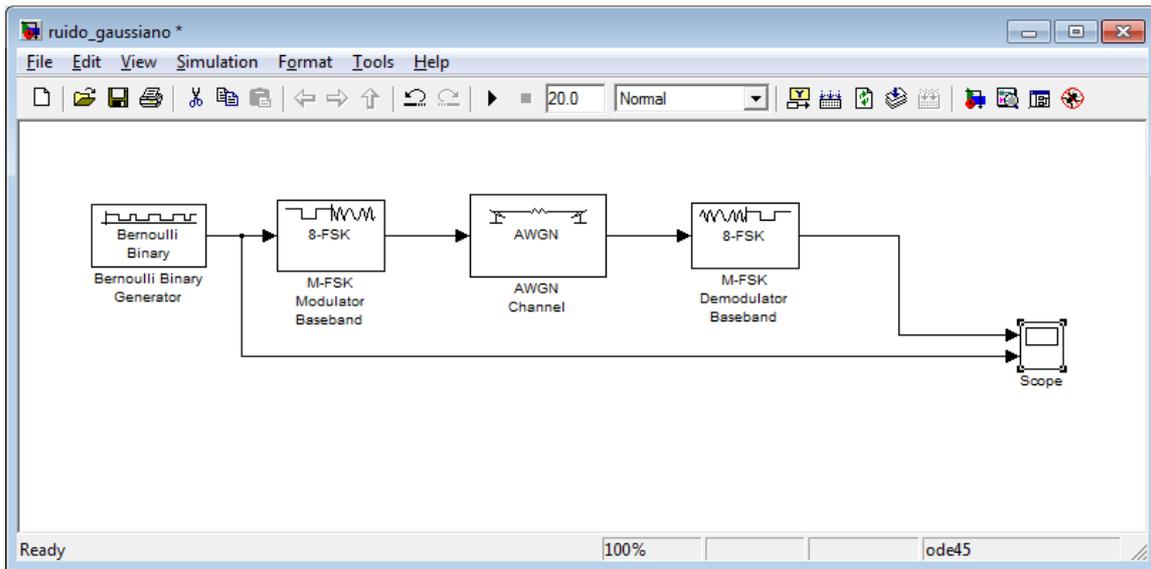


Figura III.92. Modelo Simulink Ruido Gaussiano o Térmico terminado.

El tiempo de simulación lo establecemos a 20 segundos, en el bloque Scope se tiene las siguientes salidas:

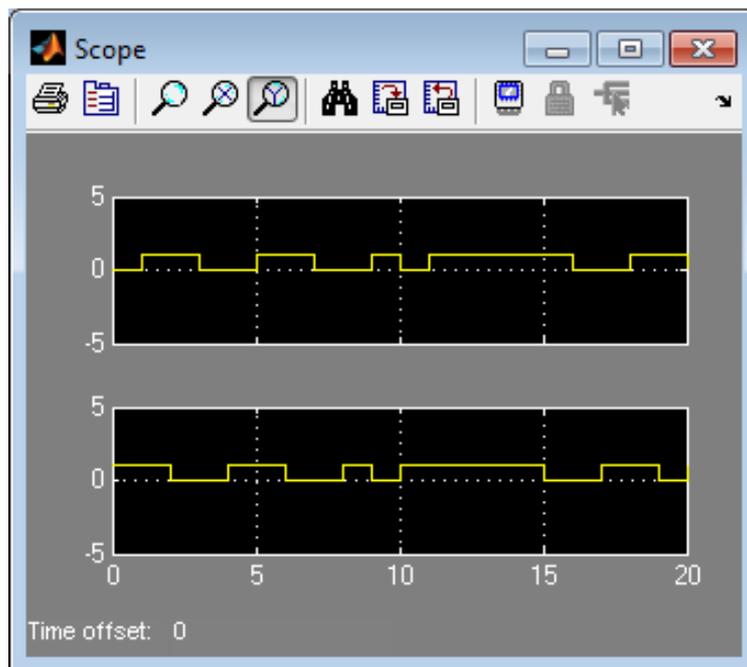


Figura III.93. Análisis de las señales generadas por el modelo.

CONCLUSIONES Y RECOMENDACIONES

Se concluye que existe cambios entre la señal que se envía en forma aleatoria con la del modulador-demodulador y canal de ruido blanco gaussiano aditivo.

Se recomienda buscar otros bloques que puedan generar señales aleatorias de entrada, tal como **Random Integer Generator**, bloque que se ubica en la biblioteca Communications Blockset/Comm Sources/Random Data Sources

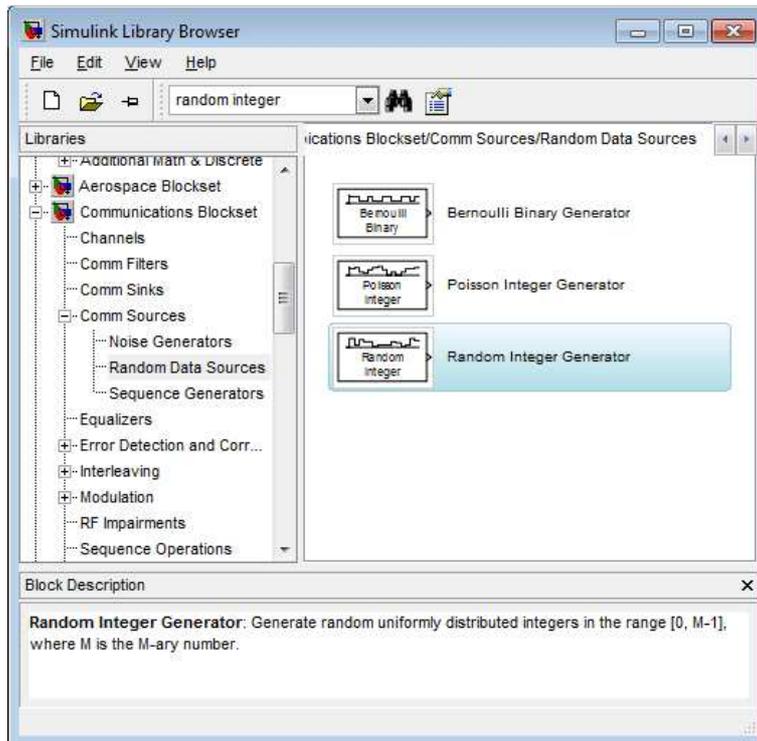


Figura III.94. Bloque Generador Pulsos Enteros Aleatorios.

Se recomienda usar el modelo **RUIDO_QAM.MDL** que usa bloques para la representación de transmisión en cuadratura de amplitud modulada digital, usada en moduladores y demoduladores.

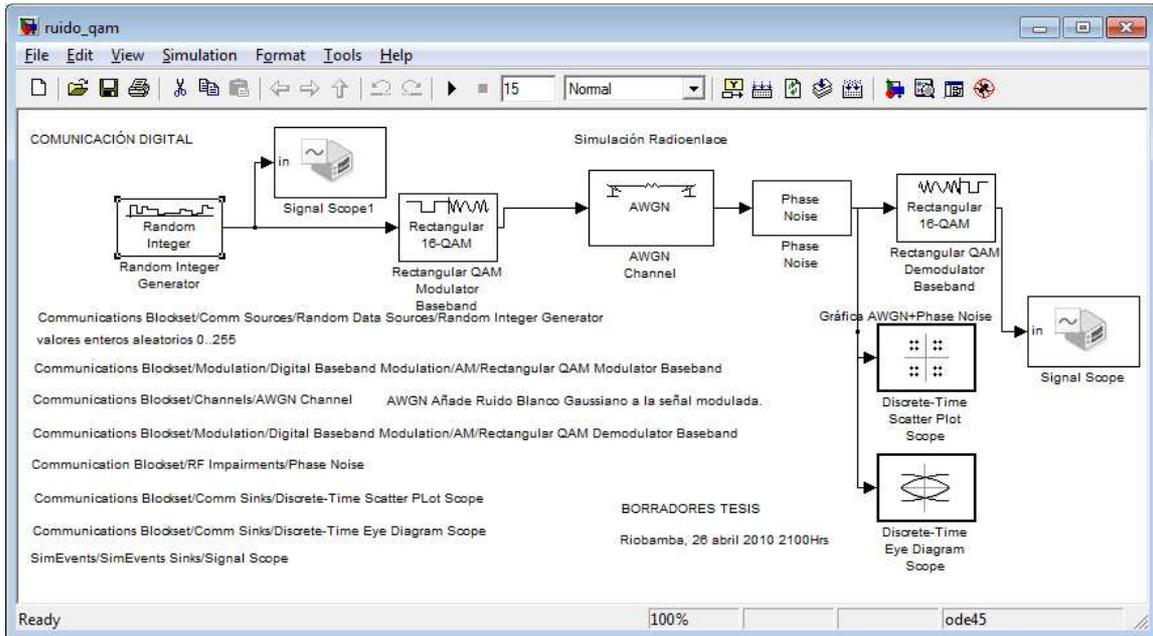


Figura III.95. Simulación QAM usando canal AWGN (Ruido Blanco Aditivo Gaussiano) o Ruido Térmico.

Que genera gráficas como las siguientes:

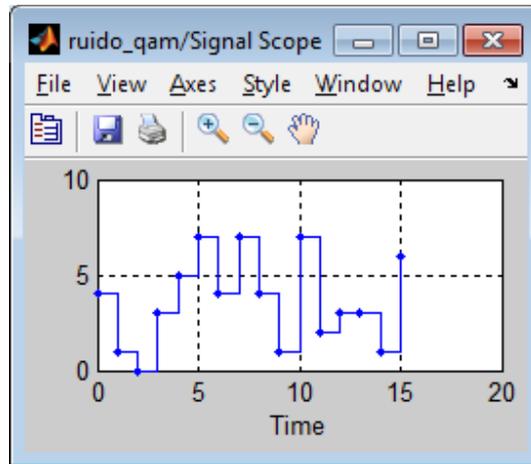


Figura III.96. Señal Digitalizada de entrada.

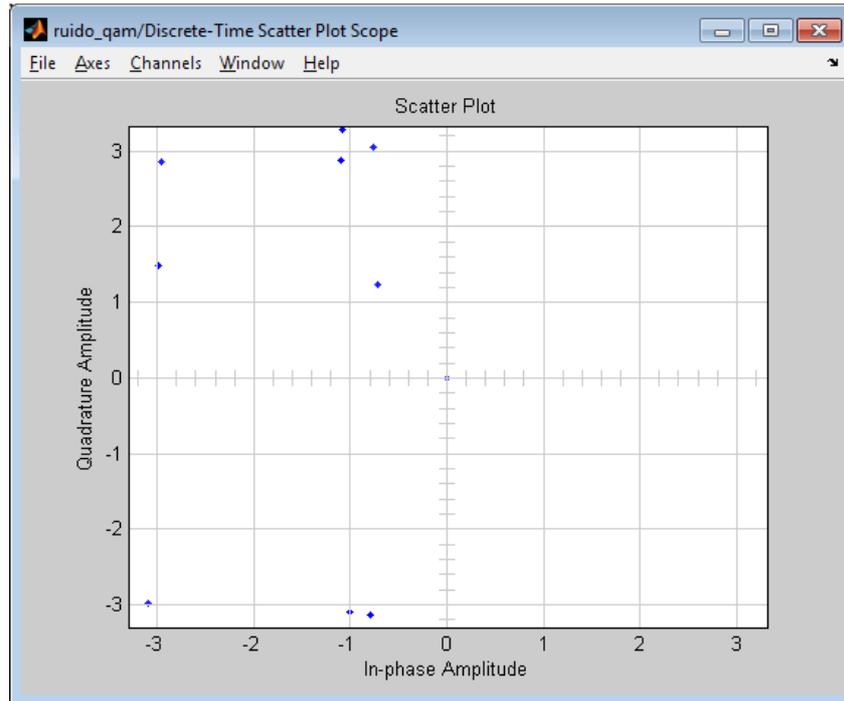


Figura III.97. Gráfica de Fase y Amplitud Q-AM

Se recomienda investigar sobre el significado de este tipo de gráficas en lo que se refiere a QAM (modulación de amplitud en cuadratura), que es una forma de AM en la que dos señales de fuentes separadas de información (es decir, dos canales) modulan a la misma frecuencia de portadora al mismo tiempo, sin interferir entre sí. Las fuentes de información modulan la misma portadora, después de haberse separado en dos señales portadoras desfasadas 90° entre sí, este esquema se llama, a veces, AM en cuadratura (QUAM o QAM).

3.10 PRÁCTICA TRANSFORMADA DISCRETA DE FOURIER.

TRANSFORMADA DISCRETA DE FOURIER USANDO UN ARCHIVO-M

Objetivo: Probar el algoritmo que me permite mostrar el desempeño de una señal y su comportamiento tanto en el dominio del tiempo como de la frecuencia, analizando su frecuencia de muestreo.

Se sugiere usar el archivo-M **PRACTICA8_1.M** incluido en el CD, hacer los respectivos análisis de las señales obtenidas en este guión y realizar cambios en sus parámetros y registrar los cambios en las señales resultantes.

Marco teórico.

Transformada Discreta de Fourier

Muchas formas de onda que se manejan en los sistemas normales de comunicaciones no se pueden definir en forma satisfactoria con ecuaciones matemáticas; sin embargo, es de interés primordial su comportamiento en el dominio de la frecuencia. Con frecuencia hay necesidad de obtener este comportamiento de señales que se captan en el dominio del tiempo, es decir, en tiempo real.

Esta es la razón por la que se desarrolló la transformada discreta de Fourier. En esta transformación se muestrea una señal en el dominio del tiempo, en tiempos discretos. Las muestras se alimentan a una computadora donde un algoritmo calcula la transformación. Sin embargo, el tiempo de computación es proporcional a n^2 , siendo n la cantidad de muestras. Para cualquier cantidad razonable de muestras, el tiempo de computación es excesivo. En consecuencia se desarrolló en 1965, un nuevo algoritmo, llamado transformada rápida de Fourier (FFT, de fast Fourier transform), por Cooley y Tukey. Con la FFT, el tiempo de cómputo es proporcional a $n \log_2 n$, y no a n^2 . Hoy se cuenta con la FFT en forma de subrutina, en muchas bibliotecas de subrutinas científicas en los grandes centros de cómputo. (TOMASI, 2003)

Con la conveniencia que otorgan las computadoras personales y la disponibilidad de circuitos integrados para el procesamiento digital de señales, el espectro de una forma de onda puede fácilmente aproximarse utilizando la transformada discreta de Fourier (DFT). Se puede demostrar cómo la DFT puede emplearse en el cálculo de muestras de una transformada continua de Fourier (CFT) y los valores para los coeficientes complejos de la serie de Fourier.

La transformada de Fourier (FT) de una forma de onda $\omega(t)$ es

$$W(f) = \mathcal{F}[\omega(t)] = \int_{-\infty}^{\infty} [\omega(t)] e^{-j2\pi ft} dt \quad (8.1)$$

Los coeficientes de una serie compleja de Fourier de una forma de onda real están relacionadas con los coeficientes de una serie polar de Fourier a través de

$$C_n = \begin{cases} 0.5D & \leftarrow \varphi_n, & n > 0 \\ D_0 & & n = 0 \\ 0.5D_{-n} & \leftarrow -\varphi_{-n} & n < 0 \end{cases}$$

Definición. La transformada discreta de Fourier (DFT) está definida mediante

$$X(n) = \sum_{k=0}^{k=N-1} x(k) e^{-j(2\pi/N)nk} \quad (8.2)$$

Donde $n=0,1,2,\dots, N-1$, y la transformada discreta inversa de Fourier (IDFT) está definida por

$$x(k) = \frac{1}{N} \sum_{n=0}^{n=N-1} X(n) e^{j(2\pi/N)nk} \quad (8.3)$$

Donde $k=0,1,2,\dots,N-1$.

El tiempo y la frecuencia no aparecen explícitamente debido a que las ecuaciones (8.2 y 8.3) sólo son definiciones implementadas en una computadora digital para calcular N valores para la DFT y la IDFT, respectivamente.

MATLAB utiliza las definiciones para DFT e IDFT dadas por las ecuaciones (8.2 y 8.3), excepto que los elementos del vector están indexados del 1 a N en lugar de 0 a N-1. Por lo tanto, los algoritmos en MATLAB para la FFT están relacionados con las ecuaciones (8.2 y 8.3) a través de

$$X = \mathbf{fft}^{15}(x) \quad (8.4)$$

$$x = \mathbf{ifft}(X) \quad (8.5)$$

Donde \mathbf{x} es un vector de N elementos correspondiente a las muestras de la forma de onda y \mathbf{X} es el vector de N elementos de la DFT. Se escoge una N que sea potencia de 2 (es decir, $N = 2^m$, donde m es un entero positivo). Si se utiliza otro software para la FFT, el usuario debe estar consciente de las definiciones específicas que están implementadas de tal manera que los resultados puedan interpretarse adecuadamente. (COUCH, 2008)

Desarrollo de la práctica

Copie o transcriba el archivo

`%% PRACTICA8_1.M`

¹⁵ DFT (Discrete Fourier Transform)

```
% Transformada Discreta de Fourier.
% Fuente: Sistemas de Comunicación Digitales y Analógicos.
% Leon W. Couch II, 7ma Ed. Pearson Educación.
% ISBN: 978-970-26-1216-2 pp.94
%% TESIS DE GRADO
% Hernando Santillán
% Lunes, 27 Septiembre 2010 1549Hrs

clear;

% Calculate the FFT for a truncated step
% Let tend be the end of the step.
%Calcula la Transformada Rápida de Fourier de una señal truncada escalón.
%la parte final de la función escalón.

M = 8;
N = 2^M;
n = 0:1:N-1;
tend = 1;
T = 10;
dt = T/N;
t = n*dt;

w = zeros(length(t),1);
for (i = 1:1:length(w))
    if (t(i) <= tend)
        w(i) = 1;
    end;
end;

% Calculando FFT (TRF-Transformada Rápida de Fourier.
W = dt*fft(w);
f = n/T;

subplot(211);
plot(t,w);
xlabel('t');
ylabel('w(t)');
title('Onda o Señal en el Tiempo');

subplot(212);
plot(f(1:N/2),abs(W(1:N/2)));
xlabel('f');
title('Magnitud del Espectro a fs/2');
%fs=frecuencia de muestreo (frequency sampling).
subplot(111);
```

Al correr este guión, la gráfica resultante se muestra en la figura III.98, cabe destacar en esta práctica el uso de la función FFT de MATLAB que resuelve el algoritmo que se explico en la parte del fundamento teórico.

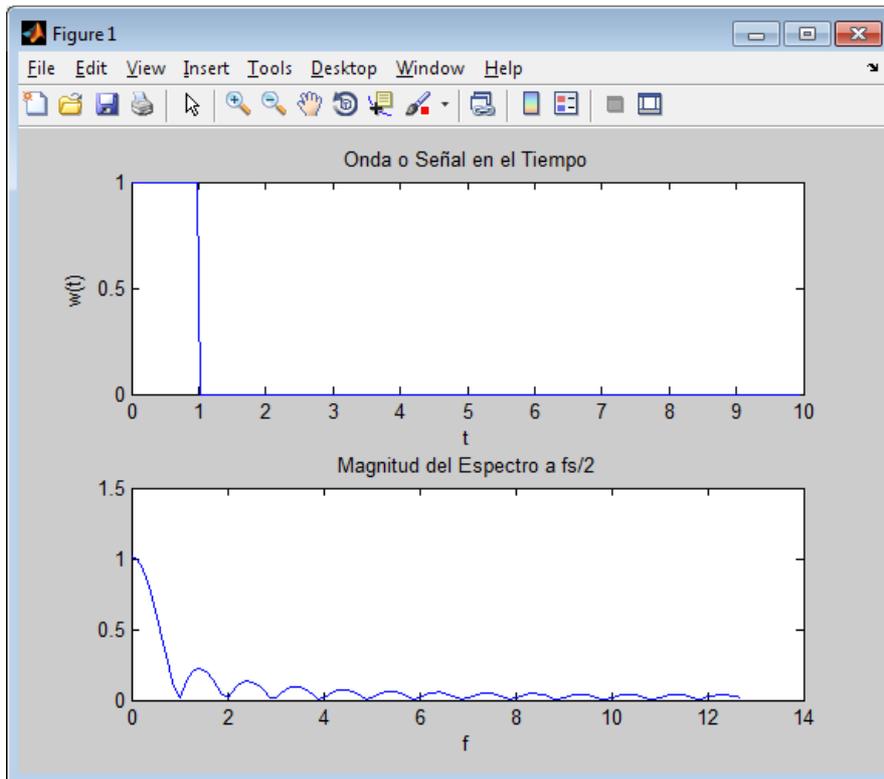


Figura III.98. Resultado de una señal truncada escalón en el dominio del tiempo y frecuencia.

CONCLUSIONES Y RECOMENDACIONES

Se concluye que al usar archivos-M podemos simular y entender el uso de la transformada rápida de Fourier que nos sirve para entender señales de orden complejo, difíciles de analizar usando ecuaciones matemáticas, pero que con la ayuda de un ordenador y software científico como lo es MATLAB nos simplifican esta tarea.

Se recomienda investigar y aplicar la función **IFFT()**, función que usa MATLAB y que menciona en la ecuación 8.5, usando **HELP IFFT**, y luego dar clic en el hipertexto **doc ifft**.

Se recomienda además investigar sobre el fundamento teórico de la transformada Z, que es parte del plan analítico de la materia y probar el modelo **FILTRO_DIGITAL_PASABANDA.MDL**, que nos permite ingresar una función de transferencia y analizar una señal seno en forma continua y discreta, siendo un filtro digital pasabanda de segundo orden.

Nótese que la función de transferencia $N(z) = \frac{1-z^{-1}}{1-0.909z^{-1}+0.809z^{-2}}$ se ingresa en el bloque Discrete Filter (filtro discreto), de la siguiente forma:

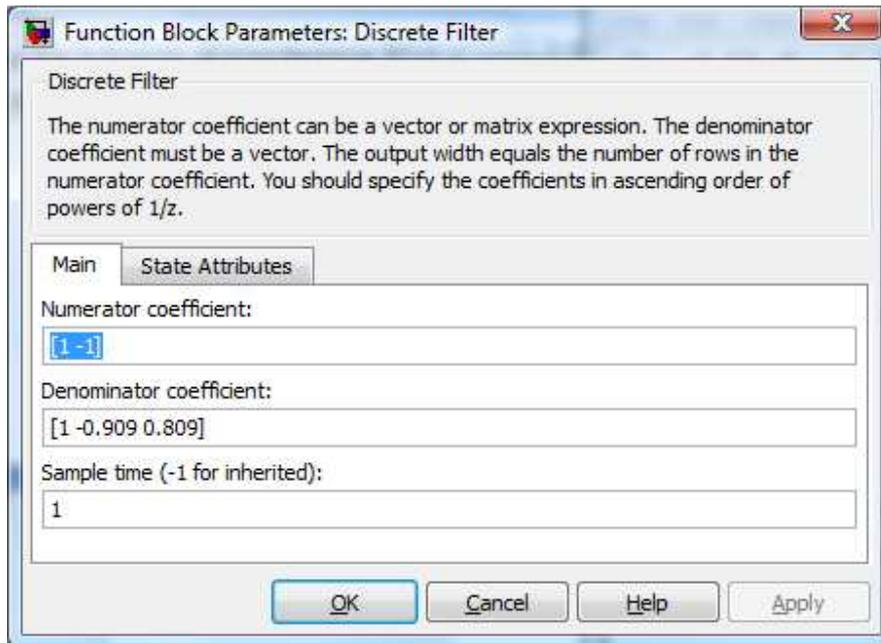


Figura III.99. Ingreso de coeficientes de la función de transferencia recomendada.

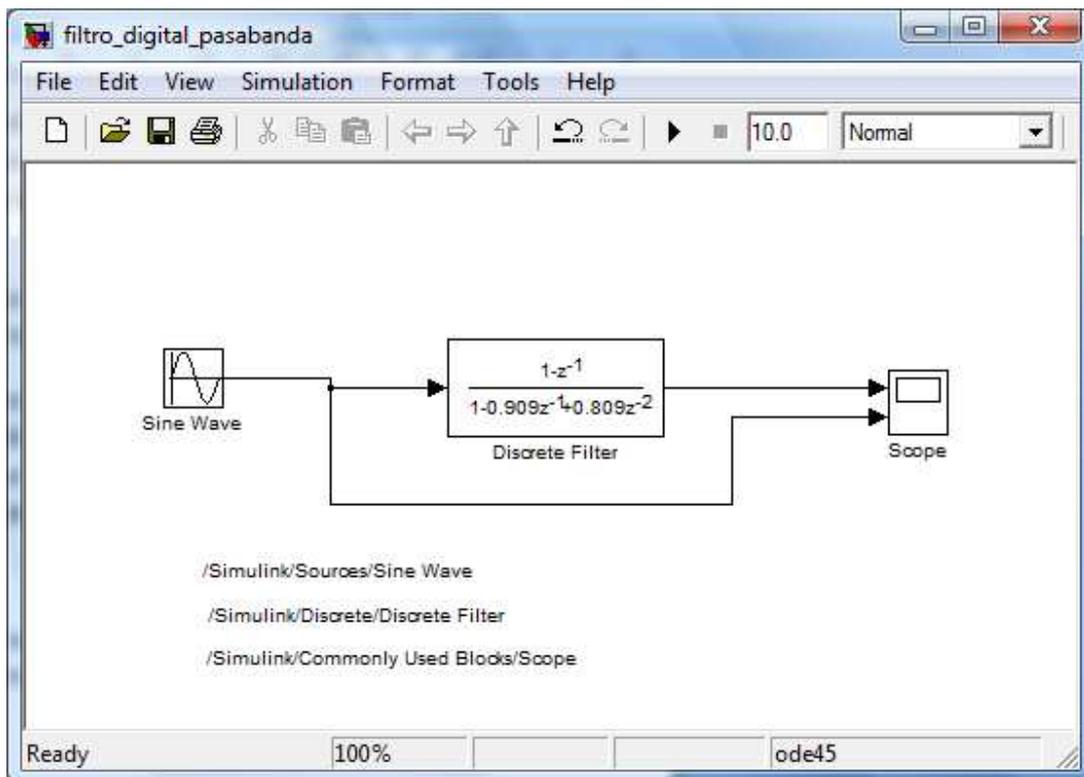


Figura III.100. Modelo de Transformada Z a investigar.

La práctica genera esta gráfica (figura III.100), en donde claramente se aprecia la misma onda seno a la que se le aplica un filtro digital pasabanda y la onda normal.

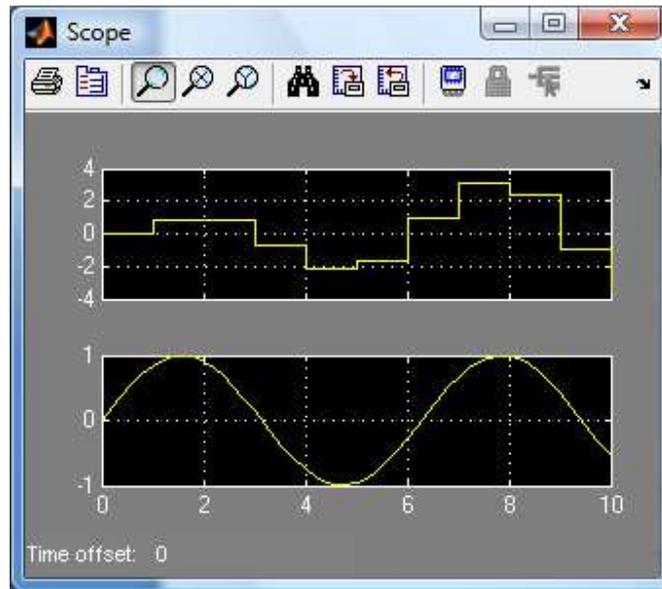


Figura III.101. Salida de la práctica recomendada.

Se recomienda además que luego de haber estudiado cada una de las prácticas planteadas en esta tesis investigativa, proponerse nuevas opciones de simulación y nuevas prácticas orientadas por cada uno de los docentes de la asignatura de ANÁLISIS DE SEÑALES, usando esta importante herramienta EDA como es MATLAB en su distribución 2008, migrar en caso sea necesario a las nuevas versiones del software, tal como la distribución 2010 que ya está en producción.

CAPÍTULO IV

ANÁLISIS DE RESULTADOS

4.1 DESVIACIÓN ESTÁNDAR DE UNA MUESTRA.

Definición. La desviación estándar de un conjunto de valores muestrales, es la medida de variación de los valores con respecto a la media. Es un tipo de desviación promedio de los valores con respecto a la media, que se calcula utilizando las fórmulas 4.1 y 4.2. La fórmula 4.2 es sólo una versión diferente de la 4.1 pero algebraicamente son iguales.

$$s = \sqrt{\frac{\sum(x - \bar{x})^2}{n - 1}} \quad \text{ec 4.1}$$

$$s = \sqrt{\frac{n \sum x^2 - (\sum x)^2}{n(n - 1)}} \quad \text{ec 4.2}$$

Para nuestro estudio se ha empleado la fórmula 4.1, y se han considerado las siguientes características de esta medida de variación estadística. (TRIOLA, 2009)

- ✚ La desviación estándar es una medida de variación de todos los valores con respecto a la media.
- ✚ El valor de la desviación estándar s generalmente es positivo, sólo es igual al cero cuando todos los valores de los datos son el mismo número. (Nunca es negativa). Además, valores grandes de s implican mayores cantidades de variación.
- ✚ El valor de la desviación estándar s puede aumentar de manera drástica con la inclusión de uno o más valores extremos (valores de datos que se encuentran muy lejos de los demás).
- ✚ Las unidades de desviación estándar s (como minutos, pies, libras, notas, etc.), son las mismas de los datos originales.

4.2 CÁLCULO DE LA DESVIACIÓN ESTÁNDAR S A PARTIR DE UNA POBLACIÓN CONOCIDA.

Las prácticas se evaluaron con una población N de 40 estudiantes de las carreras de Ingeniería Electrónica, en las especialidades de Telecomunicaciones y Control, en la materia de Herramientas EDA a cargo del docente Ing. Pablo Lozada. Que fue uno de los principales objetivos de nuestra tesis; el desarrollo de laboratorios de MATLAB para entender los conceptos teóricos de la materia de Análisis de Señales.

Se planteó la siguiente tabla de evaluación de cada una de las prácticas.

Nota	Equivalencia
10-9	Excelente
8	Muy Buena
7	Buena
6-0	Insuficiente

Tabla IV.I. Notas y equivalencias usadas en la muestra estadística.

Aplicado un test sencillo en el cada uno de los estudiantes evaluó el CD que contiene cada una de las prácticas, obteniéndose una nota promedio por cada uno de ellos.

Se obtuvo el siguiente valor con respecto a la media de su apreciación $\bar{x} = 8.575$, con respecto a un universo o población de N=40 estudiantes.

El valor de $\sum(x - \bar{x})^2 = 47.775$, el valor de $n - 1 = 39$ de la ecuación 4.1 para el cálculo de la desviación estándar, entonces:

$$s = \sqrt{\frac{\sum(x - \bar{x})^2}{n - 1}} = \sqrt{\frac{47.775}{39}} = 1.225$$

El valor de la desviación estándar es $s=1.225$.

4.3 DESVIACIÓN ESTÁNDAR DE UNA POBLACIÓN σ

En nuestra definición de la desviación estándar nos referimos a datos muestrales. Para calcular la desviación estándar sigma σ , de una población, se utiliza la fórmula ligeramente diferente, en vez de dividir entre n-1, se divide entre el tamaño N de la población, como en la siguiente expresión:

$$\sigma = \sqrt{\frac{\sum(x - \bar{x})^2}{N}} \text{ ec. 4.3}$$

Por lo que para calcular el valor de la desviación estándar poblacional, tomando en cuenta que el valor de $\sum(x - \bar{x})^2 = 47.775$, tenemos entonces:

$$\sigma = \sqrt{\frac{\sum(x - \bar{x})^2}{N}} = \sqrt{\frac{47.775}{40}} = 1.194$$

4.4 CÁLCULO DEL ESTADÍSTICO DE PRUEBA, DISTRIBUCIÓN CHI-CUADRADA.

El estadístico de prueba es un valor que se utiliza para tomar la decisión sobre la hipótesis nula y se calcula convirtiendo al estadístico muestral (en nuestro caso la desviación estándar muestral s) en una puntuación X^2 , bajo el supuesto de que la hipótesis nula es verdadera.

Por lo que se asume el estadístico de prueba

$$X^2 = \frac{(n - 1)s^2}{\sigma^2} \text{ ec 4.4}$$

Reemplazando los valores ya calculados en nuestro estudio estadístico que tiene como objetivo probar la hipótesis alternativa estadística H_1 .

Teniendo ya calculados los siguientes valores:

s	1.225
σ	1.194
n-1	39

Tabla IV.II. Valores calculados de la desviación estándar, poblacional, y población.

$$X^2 = \frac{(n - 1)s^2}{\sigma^2} = \frac{39(1.225)^2}{(1.194)^2} = \mathbf{41.05}$$

El estadístico de prueba para una media usa una distribución Chi cuadrada, que para nuestro estudio se ha calculado en el valor de 41.05.

4.5 PRUEBA DE HIPÓTESIS

Definición. En estadística una hipótesis es una aseveración o afirmación acerca de una propiedad de una población.

La hipótesis planteada en esta tesis es la siguiente aseveración:

HIPOTESIS: DETERMINAR Y DEMOSTRAR EL SOFTWARE DE MATLAB COMO HERRAMIENTA ÚTIL DE DISEÑO, MODELACIÓN Y CÁLCULO EN LA ASIGNATURA DE ANÁLISIS DE SEÑALES.

Dada esta hipótesis se han cumplido los siguientes objetivos:

Identificar la hipótesis nula y la hipótesis alternativa, expresadas en forma simbólica.

Hipótesis Nula. Denotada por H_0 , es la afirmación de que el valor de un parámetro de población (en nuestro caso la desviación estándar de población σ), es igual a un valor aseverado.

$$H_0: \sigma = 1.194 \quad (\text{hipótesis nula})$$

Hipótesis Alternativa. Denotada por H_1 , es la afirmación de que el parámetro tiene un valor que, de alguna manera difiere de la hipótesis nula.

La forma simbólica de la hipótesis alternativa debe emplear alguno de estos símbolos: $<$, $>$ o \neq (menor que, mayor que o distinto a). Por lo tanto se define como hipótesis alternativa en función del mismo parámetro de población (desviación estándar poblacional sigma σ).

$$H_1: \sigma \neq 1.194 \quad (\text{hipótesis alternativa})$$

Dada ya nuestra aseveración de hipótesis y los datos muestrales obtenidos en el laboratorio de la materia de Herramientas EDA, bajo la supervisión del docente de 5to semestre Ing. Pablo Lozada, profesor de las carreras de Telecomunicaciones y Control, datos con los que ya se ha calculado y obtenido nuestro estadístico de prueba usando la distribución Chi-cuadrada $X^2 = 41.05$.

Región Crítica, Nivel de Significancia, Valor Crítico y Valor P.

La región crítica (o región de rechazo) es el conjunto de todos los valores del estadístico de prueba que pueden provocar que rechacemos la hipótesis nula.

El nivel de significancia (denotado por α) es la probabilidad de que el estadístico de prueba caiga en una región crítica cuando la hipótesis nula es verdadera. Si el estadístico de prueba cae en la región crítica, rechazamos la hipótesis nula, de manera que α es la probabilidad de cometer el error de rechazar la hipótesis nula cuando es verdadera. Las opciones comunes para $\alpha = 0.05$.

Un valor crítico es cualquier valor que separa la región crítica (donde rechazamos la hipótesis nula) de los valores del estadístico de prueba que no conducen al rechazo de la hipótesis nula. Los valores críticos dependen de la naturaleza de la hipótesis nula, de la distribución muestral que se aplique y el nivel de significancia α .

Dos colas, cola izquierda y cola derecha. Las colas en una distribución son las regiones extremas limitadas por los valores críticos. Algunas pruebas de hipótesis incluyen dos colas, que es nuestro caso, otras una cola derecha y otras la cola izquierda.

Prueba de dos colas. La región crítica se encuentra en las dos regiones extremas (colas) bajo la curva generada por la distribución Chi-cuadrada, que es nuestro estadístico de prueba. Siendo el **valor de significancia alfa $\alpha=0.05$** se divide en dos de 0.025,

Por lo que usando la tabla A-4 de la distribución Chi-cuadrada, identificamos primero la columna de grados de libertad, en nuestro caso población $N=40$, $n-1=39$ grados de libertad, la fila de grados

de libertad más cercana es **30**. Entonces en la cola de la izquierda hacemos $1-0.025=0.975$, y la cola de la derecha 0.025 .

Los valores en la tabla A-4 son los siguientes:

Grados de libertad	$1-0.025=0.975$	0.025
30	16.791	46.979

Tabla IV.III. Datos de la tabla A-4 (Anexa) de la Distribución Chi-Cuadrada.

Planteo de la prueba de Hipótesis

Luego de analizar los valores conseguidos de la tabla A-4 anexa a este capítulo, se tienen los dos valores críticos de nuestra distribución Chi-cuadrada de dos colas. El valor de nuestro estadístico de prueba es de $X^2 = 41.05$, que NO está dentro de los valores críticos de la tabla; por lo que se concluye:

No se rechaza la Hipótesis Nula $H_0: \sigma = 1.194$

Y se comprueba de que la **Hipótesis Alternativa** $H_1: \sigma \neq 1.194$ es VERDADERA.

De esta manera se ha podido comprobar que la aseveración estadística planteada en esta tesis es congruente con la hipótesis planteada y los objetivos trazados. A continuación se aprecia la gráfica que demuestra esta ponencia.

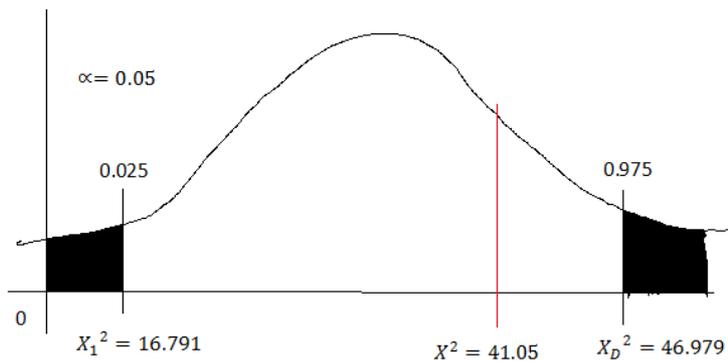


Figura IV.102. Ubicación del Estadístico de Prueba en la Distribución Chi-cuadrada para la prueba de hipótesis H1.

De esta manera se concluye, que nuestra tesis ha sido validada por medio de la aceptación de los estudiantes del Quinto Semestre de la Escuela de Ingeniería Electrónica, en sus dos especialidades, que fue uno de los objetivos principales planteados en la misma.

EVALUACIÓN DEL SISTEMA BASADO EN UN CD DE PRÁCTICAS

La estrecha relación de los temas impartidos en clases de la materia de Análisis de Señales con las prácticas desarrolladas en esta tesis y salvadas en un medio magnético (CD), hizo que al momento de usar los modelos Simulink, archivos-M y compaginar el marco teórico escrito para cada una de las prácticas, los estudiantes de las carreras de Control y Telecomunicaciones se sintieran rápidamente familiarizados, por su perfil de materias de programación impartidas dentro de la formación básica en la escuela de ingeniería electrónica de la ESPOCH.

Tomando en cuenta la media en lo que se refiere a la calificación estimada por la población estudiantil, se infiere que la herramienta tuvo una aceptación que se espera sea mejorada, debido a que el estudiante puede revisar, depurar y optimizar algoritmos en lo que se refiere a archivos-M, o probar muchos de los bloques que posee MATLAB dentro de las áreas relacionadas con nuestras carreras de formación de electrónica.

CONCLUSIONES

Se concluye que al usar archivos-M podemos simular y entender el uso de diferentes procesos matemáticos tal como la transformada rápida de Fourier que nos sirve para entender señales de orden complejo, difíciles de analizar usando ecuaciones, pero que con la ayuda de un ordenador y software científico como lo es MATLAB nos simplifican esta tarea.

Se concluyó en forma estadística que la hipótesis planteada fue aceptada o tomada, y que el objetivo principal y específicos llegaron a feliz término.

Se concluye que se ha podido distinguir la diferencia entre muchos tipos de ondas generadas por los diferentes bloques en caso del uso de Simulink o de ondas escritas bajo los archivos-M, que nos sirven para su mejor entendimiento, con las cuales se pueden tratar en la etapa de modelaje y diseño, teniéndose un claro margen de ahorro de tiempo y dinero en vez de probar con sistemas físicos reales.

Se concluye que luego de estudiar, investigar, desarrollar y evaluar la herramienta científica MATLAB, se ha obtenido un material de apoyo para la materia de "ANÁLISIS DE SEÑALES" en un CD listo para ser utilizado en las prácticas de laboratorio de la asignatura común en las dos carreras profesionales de Telecomunicaciones y Control en el quinto semestre de la actual malla curricular.

Se concluye además que se ha cumplido con el objetivo específico de conseguir redactar documentos técnico-científicos a través de procesos para crear modelos Simulink y archivos-M de MATLAB, documentos estrechamente relacionados con los temas del plan analítico de la materia de Análisis de Señales.

Finalmente se ha concluido de que luego de las pruebas de éstas prácticas de laboratorio con una población estudiantil de cuarenta alumnos, se probó su real uso; recibiendo una interesante retroalimentación por parte del sector docente y docente que participó en la evaluación de las mismas, ayudando a mejorar las mismas y que estas sean el punto de inicio para seguir aprovechando estas herramientas EDA.

RECOMENDACIONES

Se recomienda buscar otros bloques de Simulink de los que se han mostrado en este trabajo investigativo y que puedan generar señales aleatorias de entrada y salida, probar además la compatibilidad de datos entre los mismos.

Se recomienda además que luego de haber estudiado cada una de las prácticas planteadas en esta tesis investigativa, proponerse nuevas opciones de simulación y nuevas prácticas orientadas por cada uno de los docentes de la asignatura de ANÁLISIS DE SEÑALES, usando esta importante herramienta EDA como es MATLAB en su distribución 2008, migrar en caso sea necesario a las nuevas versiones del software, tal como la distribución 2010 que ya está en producción por parte de MATHWORKS.

Por tanto se recomienda que las prácticas sean un complemento curricular tanto para las asignaturas de Análisis de señales y Herramientas EDA, y que el CD adjunto sea utilizado y mejorado por futuros estudiantes de la escuela de ingeniería electrónica.

Se recomienda continuar con este tipo de trabajos de investigación científica en miras de fortalecer la formación académica del futuro profesional, que puede escoger el perfil de investigación y de esta forma contribuir a la actual tendencia del país de crear más investigadores científicos y ayudar al bienestar de la sociedad ecuatoriana toda.

RESUMEN

Se desarrolló y se evaluó un conjunto de prácticas de laboratorio para la materia de ANÁLISIS DE SEÑALES, asignatura de la malla curricular de estudios de la Escuela de Ingeniería Electrónica en sus dos carreras profesionales de Telecomunicaciones y Redes, y Control y Redes Industriales de la Facultad de Informática y Electrónica de la Escuela Superior Politécnica de Chimborazo.

Para el desarrollo e implementación de las prácticas se aplicaron varios test de prueba, con la respectiva retroalimentación del personal docente y docente de las asignaturas mencionadas. Para desarrollarlas se usó la herramienta para diseño MATLAB, software científico de MATHWORKS disponible en los laboratorios de la Escuela de Ingeniería Electrónica.

Como resultado del estudio, investigación y desarrollo se obtuvo un documento de práctica de laboratorio por cada uno de los capítulos de la asignatura de Análisis de Señales, junto con archivos-M y modelos Simulink, necesarios para la simulación y modelación de los enunciados teóricos planteados en la materia, motivo de nuestro estudio.

Se concluyó que las prácticas cumplieron con el objetivo principal y objetivos específicos planteados al momento de ser evaluadas junto a los educandos del quinto semestre de la escuela de ingeniería electrónica.

Por tanto se recomienda que las prácticas sean un complemento curricular tanto para las asignaturas de Análisis de señales y Herramientas EDA, y que el CD adjunto sea utilizado y mejorado por futuros estudiantes de la escuela de ingeniería electrónica.

Se recomienda continuar con este tipo de trabajos de investigación científica en miras a fortalecer la formación académica del futuro profesional, que puede escoger el perfil de investigación y de esta forma contribuir a la actual tendencia del país de crear más investigadores científicos y ayudar al bienestar de la sociedad ecuatoriana toda.

Se aclara que esta investigación sobre el uso de una herramienta EDA basada en los capítulos de la asignatura de Análisis de Señales, contempla un nivel básico hasta intermedio, dejando opción a continuar con este trabajo a futuro, en niveles más avanzados.

SUMMARY

A conjunct of practices was developed and evaluated for the SIGNAL ANALISYS subject matter, a subject of the pensum studiorum of the Electronics Engineering School and in its two professional careers; Telecommunications and Networks and Control and Industrial Networks of the Informatics and Electronics Faculty of the Chimborazo Higher Education Polytechnic School.

For the development and implementation of the practices various proof tests were applied with the respective feedback of teachers and students of the above subject matters. To develop them the design and scientific MATLAB's tool developed by MATHWORKS software is currently available in the Electronics Engineering School were used. As a result of the study, research and development a document of each practice was obtained for each subject chapter of Signal Analysis, together with M-files and Simulink models, necessary for the simulation and modeling of the theoretical statements stated on the subjects, objective of our study.

It is concluded that the practices accomplished the main objective and the specific objectives stated at the moment of being evaluated together with the students from fifth semester of the Electronics Engineering School. Therefore, it is recommended to use the practices as a curricular complement for both the Signal Analysis and EDA Tools subjects, and that the CD be used and improved by future students of the Electronics Engineering School.

It is recommended to continue this type of scientific investigation or research works to strength academic formation of the future professional who could choose the research profile so as to contribute to the actual tendency of the country of creating more scientific investigators and help higher levels with this work.

It should be pointed out that this investigation of the use of the EDA tool based on the chapters of the Signal Analysis Subject, considers a basic levels up to intermediate one, with the possibility of continuing this work at more advanced levels.

ANEXOS

CÁLCULO DE VALORES **Fórmulas y tablas**

para Estadística, décima edición, de Mario Triola
D.R. © 2006 Pearson Educación de México S.A. de C.V.

TABLA A-4 Distribución chi cuadrada (χ^2)

Área a la derecha del valor crítico

Grados de libertad	Área a la derecha del valor crítico									
	0.995	0.99	0.975	0.95	0.90	0.10	0.05	0.025	0.01	0.005
1	—	—	0.001	0.004	0.016	2.706	3.841	5.024	6.635	7.879
2	0.010	0.020	0.051	0.103	0.211	4.605	5.991	7.378	9.210	10.597
3	0.072	0.115	0.216	0.352	0.584	6.251	7.815	9.348	11.345	12.838
4	0.207	0.297	0.484	0.711	1.064	7.779	9.488	11.143	13.277	14.860
5	0.412	0.554	0.831	1.145	1.610	9.236	11.071	12.833	15.086	16.750
6	0.676	0.872	1.237	1.635	2.204	10.645	12.592	14.449	16.812	18.548
7	0.989	1.239	1.690	2.167	2.833	12.017	14.067	16.013	18.475	20.278
8	1.344	1.646	2.180	2.733	3.490	13.362	15.507	17.535	20.090	21.955
9	1.735	2.088	2.700	3.325	4.168	14.684	16.919	19.023	21.666	23.589
10	2.156	2.558	3.247	3.940	4.865	15.987	18.307	20.483	23.209	25.188
11	2.603	3.053	3.816	4.575	5.578	17.275	19.675	21.920	24.752	26.757
12	3.074	3.571	4.404	5.226	6.304	18.549	21.026	23.337	26.217	28.299
13	3.565	4.107	5.009	5.892	7.042	19.812	22.362	24.736	27.688	29.819
14	4.075	4.660	5.629	6.571	7.790	21.064	23.685	26.119	29.141	31.319
15	4.601	5.229	6.262	7.261	8.547	22.307	24.996	27.488	30.578	32.801
16	5.142	5.812	6.908	7.962	9.312	23.542	26.296	28.845	32.000	34.267
17	5.697	6.408	7.564	8.672	10.085	24.769	27.587	30.191	33.409	35.718
18	6.265	7.015	8.231	9.390	10.865	25.989	28.869	31.526	34.805	37.156
19	6.844	7.633	8.907	10.117	11.651	27.204	30.144	32.852	36.191	38.582
20	7.434	8.260	9.591	10.851	12.443	28.412	31.410	34.170	37.566	39.997
21	8.034	8.897	10.283	11.591	13.240	29.615	32.671	35.479	38.932	41.401
22	8.643	9.542	10.982	12.338	14.042	30.813	33.924	36.781	40.289	42.796
23	9.260	10.196	11.689	13.091	14.848	32.007	35.172	38.076	41.638	44.181
24	9.886	10.856	12.401	13.848	15.659	33.196	36.415	39.364	42.980	45.559
25	10.520	11.524	13.120	14.611	16.473	34.382	37.652	40.646	44.314	46.928
26	11.160	12.198	13.844	15.379	17.292	35.563	38.885	41.923	45.642	48.290
27	11.808	12.879	14.573	16.151	18.114	36.741	40.113	43.194	46.963	49.645
28	12.461	13.565	15.308	16.928	18.939	37.916	41.337	44.461	48.278	50.993
29	13.121	14.257	16.047	17.708	19.768	39.087	42.557	45.722	49.588	52.336
30	13.787	14.954	16.791	18.493	20.599	40.256	43.773	46.979	50.892	53.672
40	20.707	22.164	24.433	26.509	29.051	51.805	55.758	59.342	63.691	66.766
50	27.991	29.707	32.357	34.764	37.689	63.167	67.505	71.420	76.154	79.490
60	35.534	37.485	40.482	43.188	46.459	74.397	79.082	83.298	88.379	91.952
70	43.275	45.442	48.758	51.739	55.329	85.527	90.531	95.023	100.425	104.215
80	51.172	53.540	57.153	60.391	64.278	96.578	101.879	106.629	112.329	116.321
90	59.196	61.754	65.647	69.126	73.291	107.565	113.145	118.136	124.116	128.299
100	67.328	70.065	74.222	77.929	82.358	118.498	124.342	129.561	135.807	140.169

De Donald B. Owen, *Handbook of Statistical Tables*, © 1962 Addison-Wesley Publishing Co., Reading, MA. Reproducido con permiso del editor.

BIBLIOGRAFÍA

MALVINO, Albert. Principios de Electrónica. 7. ed. España, McGraw-Hill, 2007. pp. 82-96.

BLAKE, Roy. Sistemas Electrónicos de Comunicaciones. 2. ed. México D.F., Thomson, 2004. pp. 1-13, pp. 36-38.

CARLSON, Crilly. Sistemas de Comunicación. México, Pearson-Prentice Hall, 2007. pp. 50-65.

COUCH, Leon II. Sistemas de Comunicación Digitales y Analógicos. 7. ed. México, Pearson-Prentice Hall, 2008. pp. 33-115.

LÓPEZ, David. Matlab con Aplicaciones a la Ingeniería, Física y Finanzas. Colombia, Alfaomega, 2006. pp. 380-402.

MOORE, Holly. Matlab para Ingenieros. México, Pearson-Prentice Hall, 2007. pp. 1-50.

NILSSON, James. Circuitos Eléctricos. 7. ed. Madrid, Pearson-Prentice Hall, 2005. pp. 631-706.

O'NEIL, Peter. Matemáticas Avanzadas para Ingeniería. 5. ed. México D. F., Thomson, 2004. pp. 3-51, pp. 587-590, pp. 601, pp. 603-604.

TOMASI, Wayne. Sistemas de Comunicaciones Electrónicas. 4. ed. México, Prentice Hall, 2003. pp. 26.

TRIOLA, Mario. Estadística. 10. ed. México, Pearson-Addison Wesley, 2009. pp. 363-366, pp. 383-397, pp. 439

MATHWORKS. www.mathworks.com