

Informe Técnico

Sistema de Simulación RCLZ didáctico implementado en Simulink con MatLab

Asur Guadarrama Santana

15 de febrero de 2022

Desarrollo Tecnológico

Resumen

Este trabajo describe la implementación de un sistema didáctico de simulación de un medidor de circuitos resistivos-reactivos capacitivos e inductivos en serie y en paralelo basado en la técnica de detección sensible a la fase de señales eléctricas alternas complejas. Las componentes de la impedancia se obtienen de forma rectangular y en forma polar en función de los voltajes de salida en fase y en cuadratura de una etapa de salida de detección sensible a la fase de dos canales. El sistema esta conformado de forma modular con el fin de facilitar la monitorización de las señales de entrada y salida de cada etapa y mostrar de forma didáctica la función que desempeña cada módulo. El sistema se implementó en Simulink con MatLab utilizando las herramientas con que cuenta en sus librerías. Las ecuaciones que describen el comportamiento de los circuitos resistivos-reactivos simulados se obtuvieron realizando un análisis de función de transferencia de cada módulo para cada caso basado en los resultados de trabajos publicados anteriormente por el autor. El algoritmo de las ecuaciones mencionadas anteriormente fue programado en un módulo de procesamiento aritmético con el fin de obtener los valores de cada variable que interviene en la impedancia en estudio. Los resultados obtenidos se compararon con resultados teóricos basados en la teoría de circuitos eléctricos. El sistema se basa en la simulación de circuitos eléctricos pasivos y electrónicos analógicos lo que resulta en una herramienta didáctica para profesores y alumnos de la carrera de Ingeniería eléctrica y electrónica en temas relacionados con instrumentación y medición a nivel licenciatura y maestría.

Nomenclatura

RCLZ - Resistencia, Capacitancia, Inductancia, Impedancia
SSRCLZ - Sistema de Simulación RCLZ
MZ - Módulo de Impedancia
MA - Módulo de Amplificación
MDSF - Módulo de Detección Sensible a la Fase
MPA - Módulo de Procesamiento Aritmético
DMS - Demodulación Síncrona
DMSC - Demodulación Síncrona Conmutada
DMSCf - Demodulación Síncrona Conmutada en Fase
DMSCc - Demodulación Síncrona Conmutada en Cuadratura
DR - Despliegue de Resultados
RC - Resistencia Capacitancia
RL - Resistencia Inductancia
FPB – Filtro Paso Bajas
 R – Resistencia
 X_C – Reactancia Capacitiva
 X_L – Reactancia Inductiva
 C – Capacitancia
 L – Inductancia
 θ – ángulo de desfaseamiento
 Z – Impedancia
 Y – Admitancia
 G – Conductancia
 B - Susceptancia
 $Z(R, jX)$ - Impedancia en función de la Resistencia y la Reactancia Compleja
 $G_a(Z)$ - Ganancia en función de la Impedancia
 R_f - Resistencia de ganancia
 R_{fL} - Resistencia de ganancia para circuitos RL
 R_{fS} - Resistencia de ganancia para circuitos RC serie
 R_{fP} - Resistencia de ganancia para circuitos RC paralelo
 f - frecuencia
 A_{PP} - Amplitud pico a pico
 A_P – Amplitud pico
 $R - jX_C$ - Resistencia - Reactancia Capacitiva
 $R + jX_L$ – Resistencia + Reactancia Inductiva
 V_{ac} - Voltaje alterno senoidal
 V_{Ze} - Voltaje de entrada al MI
 V_{Zs} - Voltaje de salida al MI
 V_{SA} – Voltaje de salida del MA
 V_{sf} – Señal de voltaje de sincronía de conmutación en fase
 V_{sc} – Señal de voltaje de sincronía de conmutación en cuadratura
 V_{Sdf} – Componente en fase de voltaje directo
 V_{SDc} - Componente en cuadratura de voltaje directo
 V_{Sdf} – Señal de voltaje de salida de la etapa de DMSCf
 V_{Sdc} - Señal de voltaje de salida de la etapa de DMSCc

Índice

Introducción.....	5
1. Diagrama de bloques del Sistema de Simulación RCLZ.....	5
2. Diagrama esquemático general del Sistema de Simulación RCLZ.....	6
3. Descripción modular del Sistema de Simulación RCLZ.....	6
3.1 Módulo de Impedancia MZ (R, jX).....	6
3.2 Módulo de Amplificación MA.....	7
3.3 Módulo de Detección Sensible a la Fase MDSF de dos canales.....	8
3.4 Comprobación de la función de demodulación síncrona para cada canal.....	11
3.5 Módulo de Procesamiento Aritmético MPA.....	12
4. Simulaciones de circuitos eléctricos resistivos-reactivos.....	14
5. Resultados obtenidos con el Sistema de Simulación RCLZ.....	15
6. Conclusiones.....	16
Anexo.....	17
Introducción a Simulink	
Referencias Bibliográficas.....	23

Introducción

Dentro de la carrera de Ingeniería Eléctrica los cursos teóricos relacionados con temas de circuitos eléctricos y electrónicos por lo general se complementan con actividades prácticas en laboratorio con el uso de equipos de instrumentación electrónica. Sin embargo y de forma alternativa, actualmente se cuenta con una variedad de software educativo que permiten implementar instrumentos virtuales, con los cuales es posible realizar simulaciones en computadora del comportamiento de algún circuito eléctrico o electrónico en estudio ya sea en clase presencial o por medio de internet en forma remota. Este trabajo describe la implementación de un sistema didáctico de simulación de un medidor de circuitos resistivos-reactivos capacitivos e inductivos en serie y en paralelo basado en la técnica de detección sensible a la fase de señales eléctricas alternas complejas. Las componentes de la impedancia se obtienen de forma rectangular y en forma polar en función de los voltajes de salida en fase y en cuadratura de una etapa de detección sensible a la fase de dos canales. El sistema esta conformado de forma modular con el fin de facilitar la monitorización de las señales de entrada y salida de cada etapa y mostrar de forma didáctica la función que desempeña cada módulo. El sistema se implementó en Simulink con MatLab utilizando las herramientas con que cuenta en sus librerías (ver anexo). Las ecuaciones que describen el comportamiento de los circuitos resistivos-reactivos simulados se obtuvieron realizando un análisis de función de transferencia de cada módulo para cada caso basado en los resultados de trabajos publicados anteriormente por el autor. El algoritmo de las ecuaciones mencionadas anteriormente fue programado en un módulo de procesamiento aritmético con el fin de obtener los valores de cada variable que interviene en la impedancia en estudio. Los resultados obtenidos se compararon con resultados teóricos basados en la teoría de circuitos eléctricos. El sistema se basa en la simulación de circuitos eléctricos pasivos y electrónicos analógicos lo que resulta en una herramienta didáctica para profesores y alumnos de la carrera de Ingeniería eléctrica y electrónica en temas relacionados con instrumentación y medición a nivel licenciatura y maestría.

1. Diagrama de bloques del Sistema de Simulación RCLZ

El Sistema de Simulación RCLZ (SSRCLZ) propuesto está conformado principalmente por una fuente de voltaje alterna senoidal V_{ac} , un módulo de impedancias **MZ** con una $Z(R, jX)$ formada por circuitos resistivos-reactivos conectados en serie y en paralelo, un módulo de amplificación **MA** con ganancia $G_a(Z)$, un módulo de detección sensible a la fase **MDSF** de dos canales conformado por dos etapas de demodulación síncrona **DMS** (fase y cuadratura), un módulo de procesamiento aritmético **MPA** y el despliegue de los resultados **DR**, como se muestra en la figura 1.

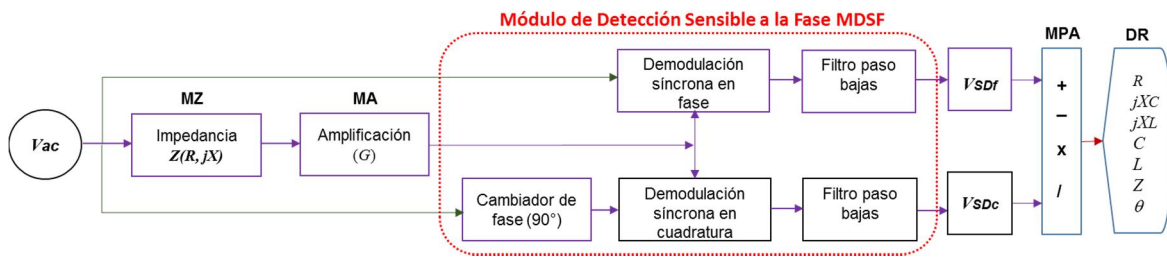


Figura 1. Diagrama de bloques del Sistema de Simulación RCLZ propuesto.

Cada componente y etapa se implementó y probó en Simulink con MatLab utilizando las librerías con que cuenta el software (ver anexo). Los módulos implementados se interconectaron para formar el Sistema de Simulación RCLZ basado en el diagrama de bloques de la figura 1. Las ecuaciones

utilizadas para calcular las variables eléctricas que describen la impedancia Z en estudio se obtuvieron de un análisis de función de transferencia del sistema propuesto por Asur *et al.* (2021).

2. Descripción de los módulos del Sistema de Simulación RCLZ

El Sistema de Medición RCLZ propuesto se implementó en Simulink con MatLab utilizando las librerías correspondientes con que cuenta el software. Las librerías cuentan con diversas herramientas que permiten desarrollar una programación de algoritmos en forma gráfica. De esta forma cada componente utilizado se va uniendo uno con otro para obtener la función requerida en cada módulo conectando las entradas y salidas de estos hasta formar el sistema propuesto, como se muestra en la figura 2.

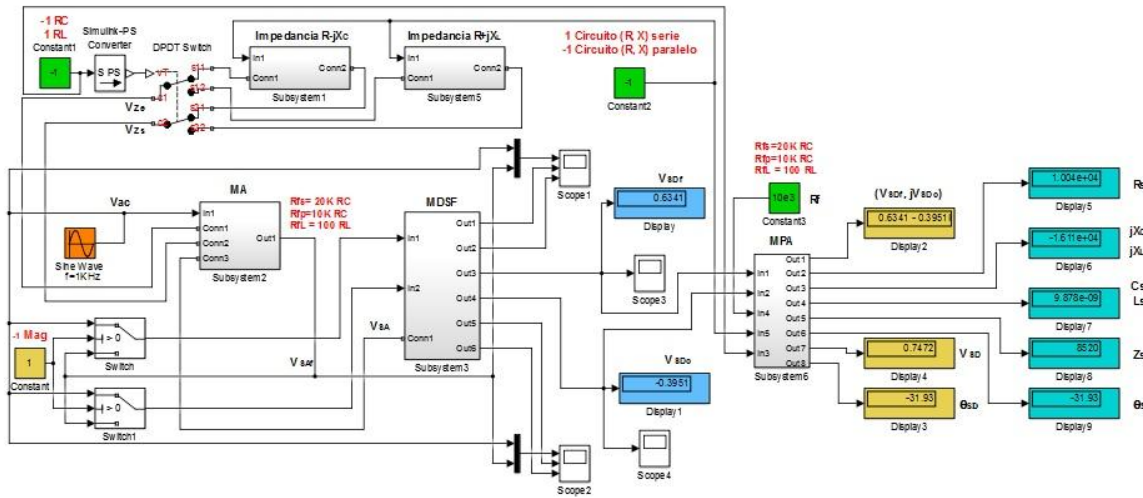


Figura 2. Esquema modular completo del Sistema de Simulación implementado en Simulink, Asur *et al.* 2021.

Cada módulo está compuesto por sub-módulos en donde se encuentran diferentes configuraciones de componentes eléctricos, electrónicos y funciones matemáticas para obtener en conjunto una simulación de señales eléctricas. Estas se evalúan como variables de voltaje que se procesan matemáticamente en las ecuaciones que describen el comportamiento de la impedancia eléctrica y así obtener los valores de las variables relacionadas por medio del **MPA**.

3. Descripción modular del Sistema de Simulación RCLZ propuesto

3.1 Módulo de Impedancia MI (R, jX)

Este módulo está formado por dos sub-módulos (**Subsystem 1** y **5**) para circuitos RC y RL, respectivamente, alimentados por una fuente de voltaje alterno V_{ac} (Sine Wave, $f = 1\text{kHz}$) con amplitud pico a pico $A_{pp} = 2\text{V}$ que se puede monitorizar en **Scope1** y **Scope2**, figura 2. La alimentación para un circuito u otro se realiza por medio de un interruptor externo (**DPDT Switch**) y una variable de condición (**Constant1**) con valores de -1 o 1 para circuitos RC y RL, respectivamente. Las conexiones en paralelo o en serie son seleccionadas por medio de un interruptor interno (**DPDT Switch**) para cada sub-módulo que se selecciona por medio de una variable de condición (**Constant2**) con valores de -1 ó 1 para la conexión del circuito en paralelo o en serie, respectivamente. Realizando doble clic sobre **Subsystem 1** y **5** se muestran los respectivos circuitos RC y RL, como se muestra en la figura 3.

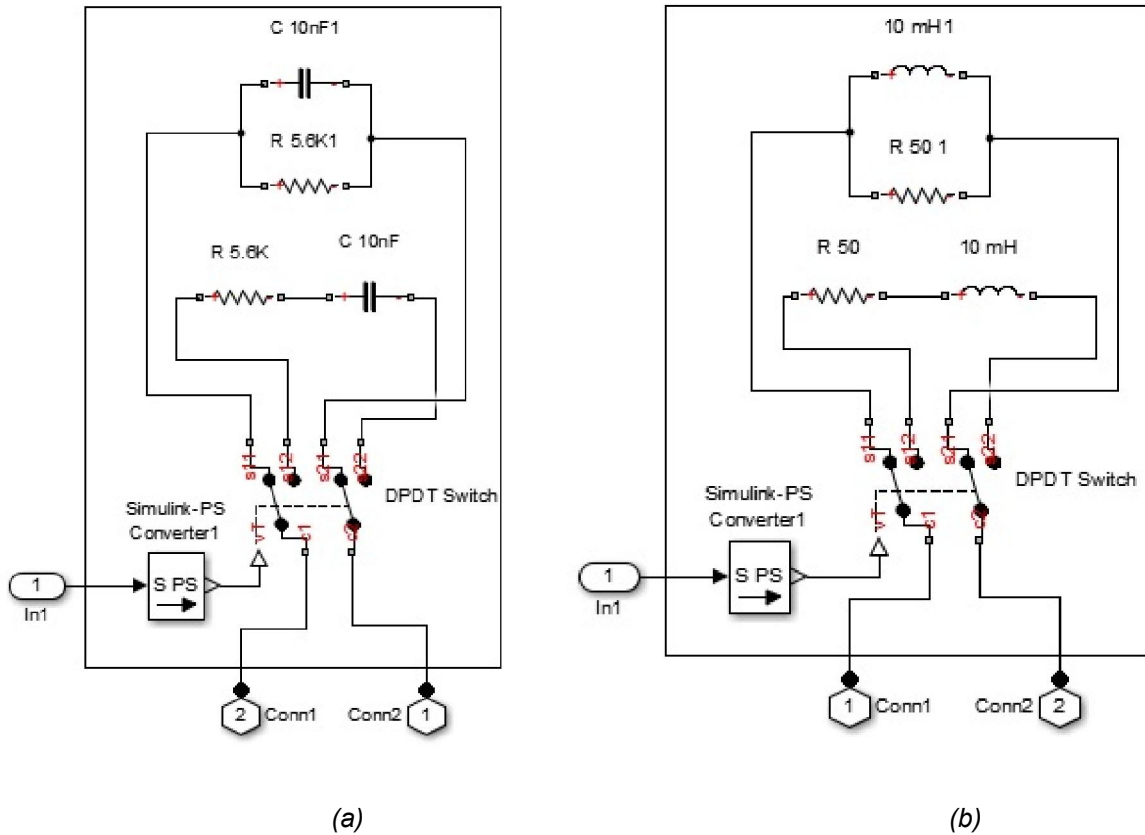


Figura 3. Módulo de Impedancia formado por los sub-módulos: a) Impedancia $R-jX_C$ (**Subsystem1**) y b) Impedancia $R+jX_L$ (**Subsystem5**).

Las conexiones de forma física (**Conn 1 y 2**) de cada sub-módulo son las que alimentan a cada circuito dependiendo del valor de las condiciones aplicadas a cada interruptor (**DPDT Switch**).

3.2 Módulo de Amplificación MA

La función del **MA (Subsystem2)** es amplificar la señal de voltaje que sale del **MI** manteniendo la fase de la impedancia seleccionada. En donde V_{Ze} es el voltaje que entra al **MI** y V_{Zs} es el voltaje de salida del **MI**, que corresponden a **c1** y **c2** del **DPDT Switch** externo, como se muestra en la figura 1. La entrada **In 1** del **MA** corresponde a la fuente V_{ca} , **Conn 1** y **Conn 2** corresponden a V_{Ze} y V_{Zs} . **Conn 3** corresponde al voltaje de salida física V_{SA} del MA y la salida Out1 corresponde también a la salida del MA pero en forma simulada, la cual se utiliza para monitorizarla en **Scope1** y **Scope2**. Además por medio de dos interruptores (**Switch** y **Switch1**) y de una constante de condición (**Constant**) con valores de -1 y 1 es posible obtener la magnitud de la señal V_{SA} directamente o la misma señal desfasada, correspondientemente. Realizando doble clic sobre el **MA (Subsystem2)** se muestra el circuito de amplificación, como se muestra en la figura 4.

Como se puede observar el **MA** presenta dos etapas de amplificación en donde la ganancia de la primera etapa (**Op-Amp**) es función de la impedancia y de R_f la cual también es función de los componentes utilizados y de la conexión (serie o paralelo) del circuito eléctrico que forma la impedancia de entrada.

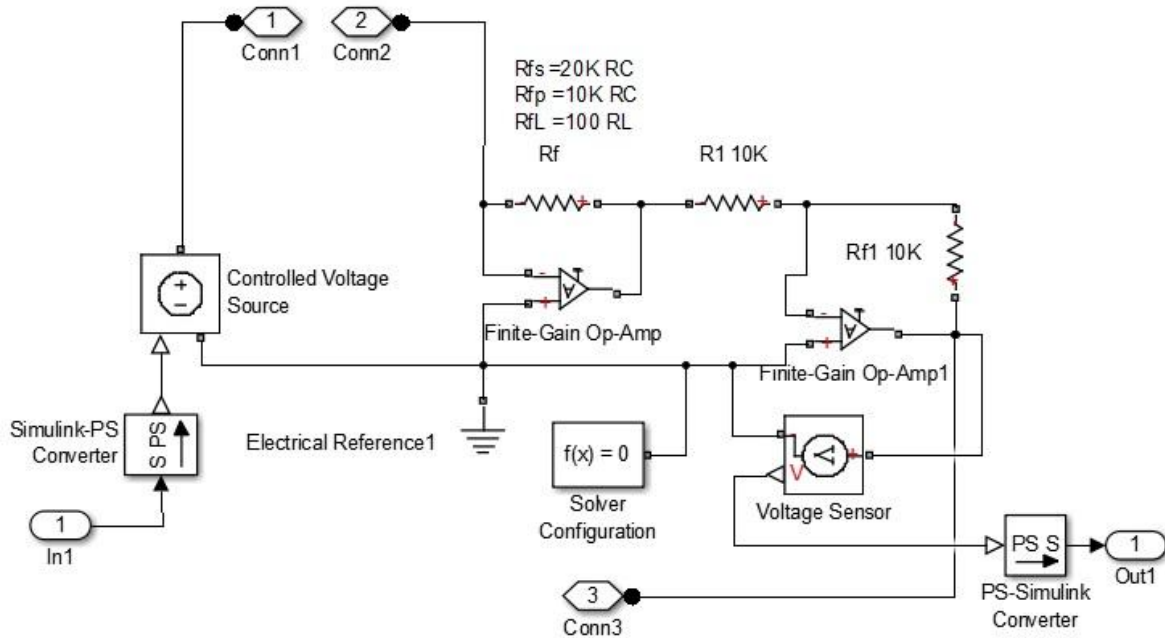


Figura 4. Módulo de Amplificación constituido por dos etapas inversoras para mantener la fase de la señal de entrada.

Los valores de Resistencia, Capacitancia, Inductancia y de R_f establecidos permiten que la salida V_{SA} del **MA** no sobrepase el rango dinámico de los amplificadores y no se presente saturación de la señal en la salida. El valor de la resistencia de ganancia R_f (R_{fs} , R_{fp} , R_{fL}) se selecciona dependiendo del tipo de circuito eléctrico que se analizará. La ganancia de la segunda etapa de amplificación (**Op-Amp1**) es unitaria ($R_i = R_{f1}$) y compensa el desfase de 180° de la primera etapa.

3.3 Módulo de Detección Sensible a la Fase MDSF de dos canales

Realizando doble clic sobre el **MDSF (Sbsystem3)** se puede ver como se encuentra conformado por dos etapas de demodulación síncrona conmutada **DMSC** una en fase (**Subsystem1**) y otra en cuadratura (**Subsystem2**), cada una de sus salidas entra a una etapa de Filtrado Paso Bajas **FPB (Analog Filter Desig 1 y 2)**, como se muestra en la figura 5. La salida de cada canal corresponden a las componentes de voltaje directo, en fase y en cuadratura están en función de la amplitud, la fase y la frecuencia del voltaje alterno de salida V_{SA} del **MA** y que entran al **MDSF**.

Realizando doble clic sobre el demodulador síncrono conmutado en fase **DMSCf (DemSincf-Subsystem1)** se muestra el circuito electrónico, figura 6. La entrada In1 corresponde al voltaje V_{ac} el cual se utiliza para generar una señal de sincronía de conmutación V_{sf} (**Out1**) por medio de una etapa de comparación y un interruptor de sincronía **SPDT Switch**. Para el **DMSCf** la frecuencia de V_{ac} es igual a la de la señal de sincronía en fase V_{sf} . El interruptor de sincronía es parte de la etapa de amplificación (**Op-Amp2**) con ganancia programable que amplifica la señal de entrada (**Conn1**) que proviene de la salida física V_{SA} del **MA** (**Conn3**). La salida simulada **Out1** permite visualizar una señal cuadrada de sincronía V_{sf} en el visualizador **Scope1**. La salida simulada **Out2** corresponde a la señal de salida en fase V_{sdf} del **DMSCf** que también se puede visualizar en **Scope1**, además entra a la etapa de **FPB** tipo Butterworth para obtener la componente en fase de voltaje directo V_{SDf} que corresponde a la salida **Out3** y cuyo valor se visualiza en **Display (azul)** y **Scope3**.

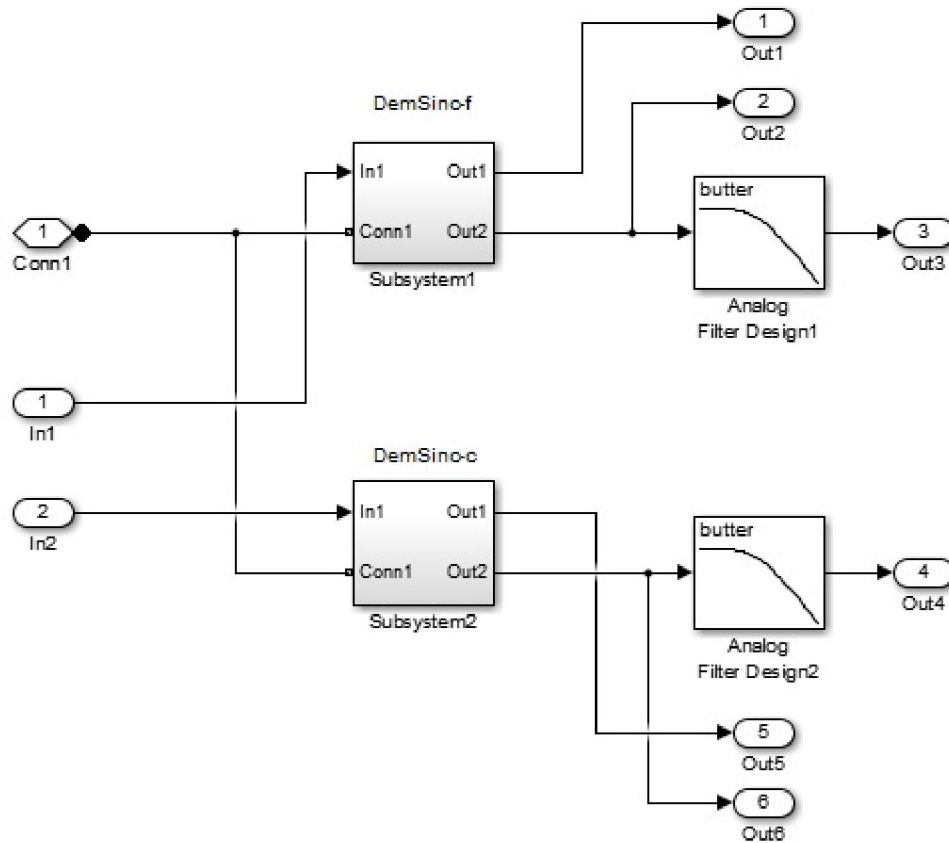


Figura 5. Módulo de Detección Sensible a la Fase de dos canales conformado por dos etapas de demodulación síncrona conmutada (**DemSincf-Subsystem1** y **DemSincc-Subsystem2**) y dos etapas de filtrado paso bajas (**Analog Filter Design 1 y 2**).

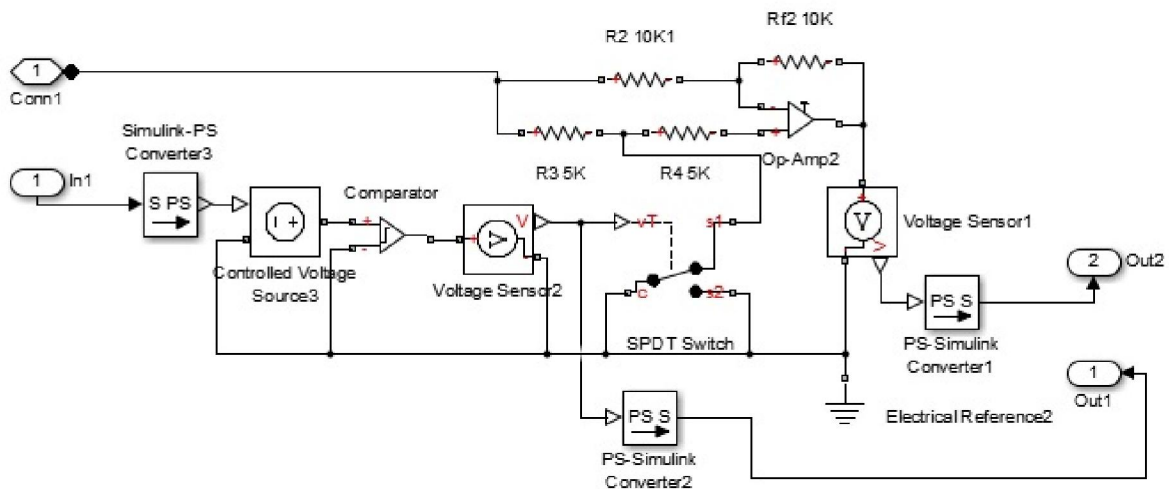


Figura 6. Demodulador Síncrono Conmutado en Fase (**DemSincf-Subsystem1**) conformado por una etapa de comparación y un amplificador de ganancia programable.

Los parámetros funcionales del **FPB** se pueden configurar realizando doble clic sobre el ícono de **Analog Filter Design1** para abrir la ventana que se muestra en la figura 7.

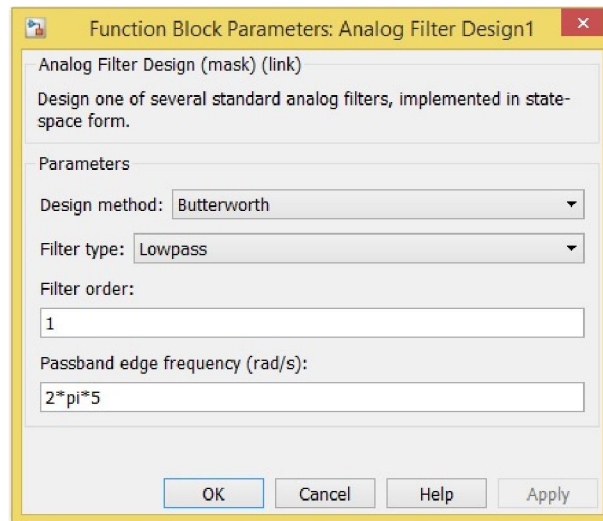


Figura 7. Ventana que muestra los parámetros funcionales del FPB (*Analog Filter Design1*).

Analizando el demodulador síncrono conmutado en cuadratura **DMSCc** (**DemSincf-Subsystem2**), realizando doble clic sobre su bloque, se puede observar que la entrada **In1** que corresponde al voltaje V_{ac} entra a una etapa de retraso (**Transport Delay1**), figura 8. El retraso de la señal V_{ac} tiene la función de provocar un desfase de 90° en la señal de sincronía en cuadratura V_{sc} (**Out1**) que puede ser visualizada por medio de la salida **Out5** en **Scope2**. Este desfase es necesario para obtener la componente de voltaje en cuadratura a la salida del **DMSCc**. La salida simulada **Out2** corresponde a la señal de salida en cuadratura V_{Sdc} del **DMSCc** que también se puede visualizar en Scope2 por medio de Out6, además entra a la etapa de **FPB** tipo Butterworth para obtener la componente en cuadratura de voltaje directo V_{Sdc} que corresponde a la salida **Out4** y cuyo valor se visualiza en **Display1** (azul) y **Scope4**.

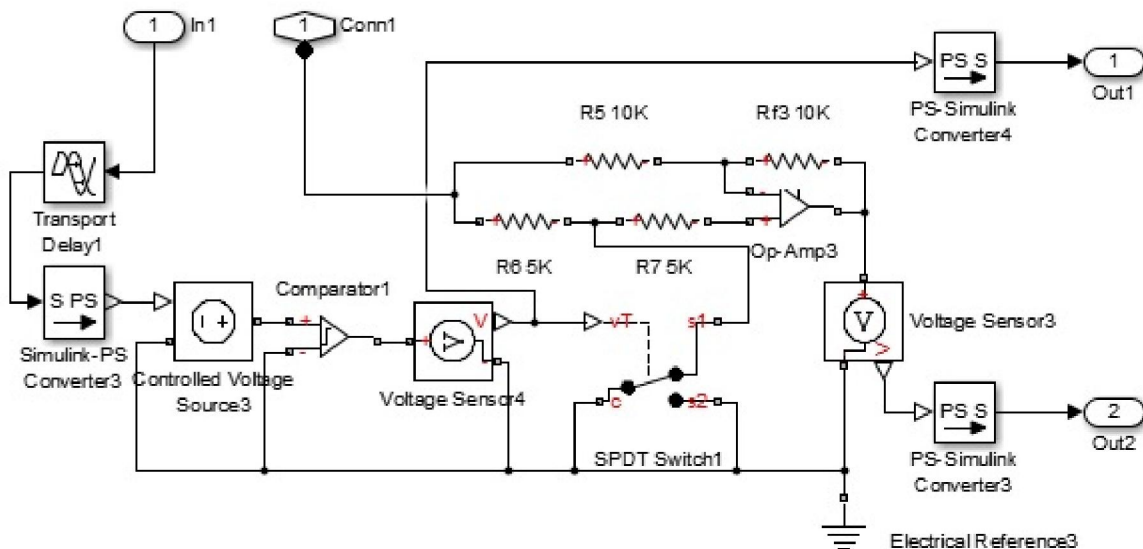


Figura 8. Demodulador Síncrono Conmutado en Cuadratura (**DemSincf-Subsystem2**) conformado por una etapa de desfase de 90° , de comparación y un amplificador de ganancia programable.

Los parámetros funcionales del **FPB** del canal en cuadratura son los mismos que los mostrados para el **FPB** del canal en fase y también se pueden configurar realizando doble clic sobre el ícono de **Analog Filter Design2**, como se muestra en la figura 7.

3.4 Comprobación de la función de demodulación síncrona para cada canal

La comprobación de la función de demodulación síncrona en fase y en cuadratura se realiza con la señal de la salida V_{SA} del **MA** en la entrada **Conn1** y con V_{ac} en las entradas **In1** e **In2** del **MDSF**. Las salidas **Out1** (V_{SA}) del **MA**, **Out1** (V_{Sf}) y **Out2** (V_{Sdf}) del **MDSF** así como V_{ac} se monitorizan en **Scope1**, mientras que las salidas **Out1** (V_{SA}) del **MA**, **Out5** (V_{Sc}) y **Out6** (V_{Sdc}) del **MDSF** así como V_{ac} se monitorizan en **Scope2**, como se muestra en la figura 9.

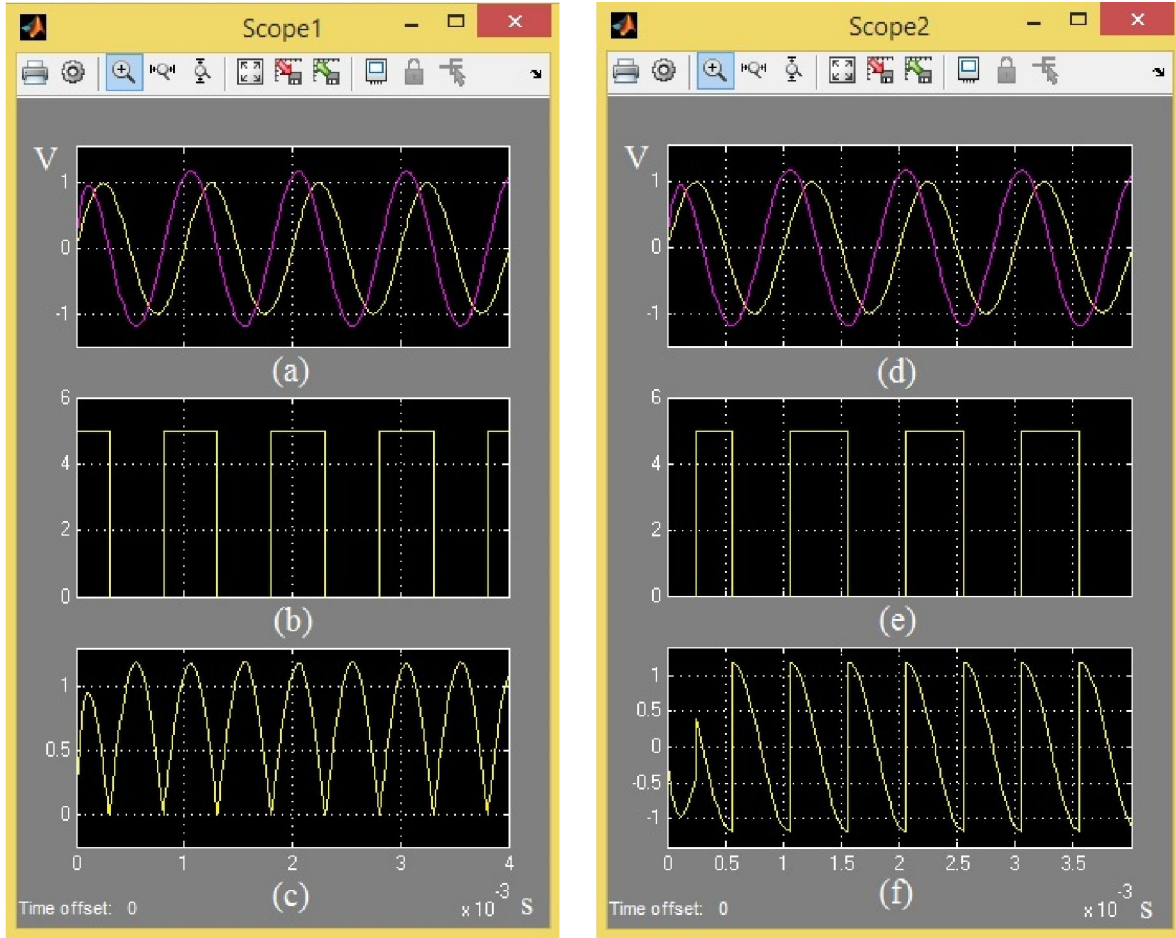


Figura 9. Señales eléctricas de las etapas de demodulación síncrona en fase y en cuadratura. (a) Señal V_{ac} (amarillo) y V_{SA} (violeta), (b) Señal de sincronía en fase V_{Sf} , (c) Señal de salida V_{Sdf} del demodulador síncrono en fase, (d) Señal V_{ac} (amarillo) y V_{SA} (violeta), (e) Señal de sincronía en cuadratura V_{Sc} , (f) Señal de salida V_{Sdc} del demodulador síncrono en cuadratura, Asur et al. 2021.

Las figuras 9(c) y 9(f) representan las salidas V_{Sdf} y V_{Sdc} de las etapas de **DMScf** y del **DMScq**, respectivamente. En el caso de la figura 9(c), se muestra una señal rectificada V_{Sdf} cuando las señales V_{SA} y V_{Sf} presentan la misma fase y frecuencia. Para el caso de la figura 9(f), se muestra una señal V_{Sdc} simétrica cuyo valor medio es cero cuando las señales V_{SA} y V_{Sc} presentan la misma frecuencia

pero se encuentran desfasadas 90°. Estas formas de onda son características de la función de demodulación síncrona con las condiciones mencionadas.

3.5 Módulo de Procesamiento Aritmético MPA

Las componentes de voltaje en fase y en cuadratura V_{SDf} y V_{SDc} obtenidas a la salida del **MDSF** son función de la impedancia $Z(R, jX)$ a la entrada del **SSRCLZ** propuesto. Para obtener los valores de las componentes de R, X_C, X_L, C, L, Z y θ , se realizó un análisis de impedancias, admitancias y función de transferencia del sistema propuesto; esto con el fin de obtener las ecuaciones que nos ayudan a evaluar estas variables en función de los valores de las componentes de voltaje en fase y cuadratura obtenidas con las simulaciones, *Asur et al. (2021)*.

Para circuitos en serie en función de $Z(R+jX)$,

$$R_s = \left(\frac{2A_p}{\pi} \right) \left(\frac{R_f}{\sqrt{V_{SDf}^2 + V_{SDc}^2}} \right) \cos \left[\tan^{-1} \left(\frac{V_{SDc}}{V_{SDf}} \right) \right] \quad (1)$$

$$X_s = \left(\frac{2A_p}{\pi} \right) \left(\frac{R_f}{\sqrt{V_{SDf}^2 + V_{SDc}^2}} \right) \sen \left[\tan^{-1} \left(\frac{V_{SDc}}{V_{SDf}} \right) \right] \quad (2)$$

$$C_s = \left\{ \left(\frac{2\omega A_p}{\pi} \right) \left(\frac{R_f}{\sqrt{V_{SDf}^2 + V_{SDc}^2}} \right) \sen \left[-\tan^{-1} \left(\frac{V_{SDc}}{V_{SDf}} \right) \right] \right\}^{-1} \quad (3)$$

$$L_s = \left(\frac{2A_p}{\omega\pi} \right) \left(\frac{R_f}{\sqrt{V_{SDf}^2 + V_{SDc}^2}} \right) \sen \left[\tan^{-1} \left(\frac{V_{SDc}}{V_{SDf}} \right) \right] \quad (4)$$

Para circuitos en paralelo en función de $Y(G+jB)$,

$$R_p = \left\{ \left[\left(\frac{2A_p}{\pi} \right) \left(\frac{R_f}{\sqrt{V_{SDf}^2 + V_{SDc}^2}} \right) \right]^{-1} \cos \left[\tan^{-1} \left(\frac{V_{SDc}}{V_{SDf}} \right) \right] \right\}^{-1} \quad (5)$$

$$X_p = \left\{ \left[\left(\frac{2A_p}{\pi} \right) \left(\frac{R_f}{\sqrt{V_{SDf}^2 + V_{SDc}^2}} \right) \right]^{-1} \sen \left[\tan^{-1} \left(\frac{V_{SDc}}{V_{SDf}} \right) \right] \right\}^{-1} \quad (6)$$

$$C_p = \left\{ \left[\left[\left(\frac{2A_p}{\pi} \right) \left(\frac{R_f}{\sqrt{V_{SDf}^2 + V_{SDc}^2}} \right) \right]^{-1} \sen \left[-\tan^{-1} \left(\frac{V_{SDc}}{V_{SDf}} \right) \right] \right]^{-1} (\omega) \right\}^{-1} \quad (7)$$

$$L_p = \left\{ \left[\left(\frac{2A_p}{\pi} \right) \left(\frac{R_f}{\sqrt{V_{SDf}^2 + V_{SDc}^2}} \right) \right]^{-1} \sin \left[\tan^{-1} \left(\frac{V_{SDc}}{V_{SDf}} \right) \right] \right\}^{-1} \left(\frac{1}{\omega} \right) \quad (8)$$

Para ambos circuitos el ángulo de desfase θ es el mismo,

$$\theta = \tan^{-1} \left(\frac{V_{SDc}}{V_{SDf}} \right) \quad (9)$$

Las ecuaciones (1) a (9) representan los valores de las variables relacionadas con la impedancia eléctrica de circuitos resistivos-reactivos conectados en serie y en paralelo, el algoritmo de las ecuaciones se implementó en el **MPA** para evaluar las variables mencionadas en función de los voltajes V_{SDf} y V_{SDc} , como se muestra en la figura 10.

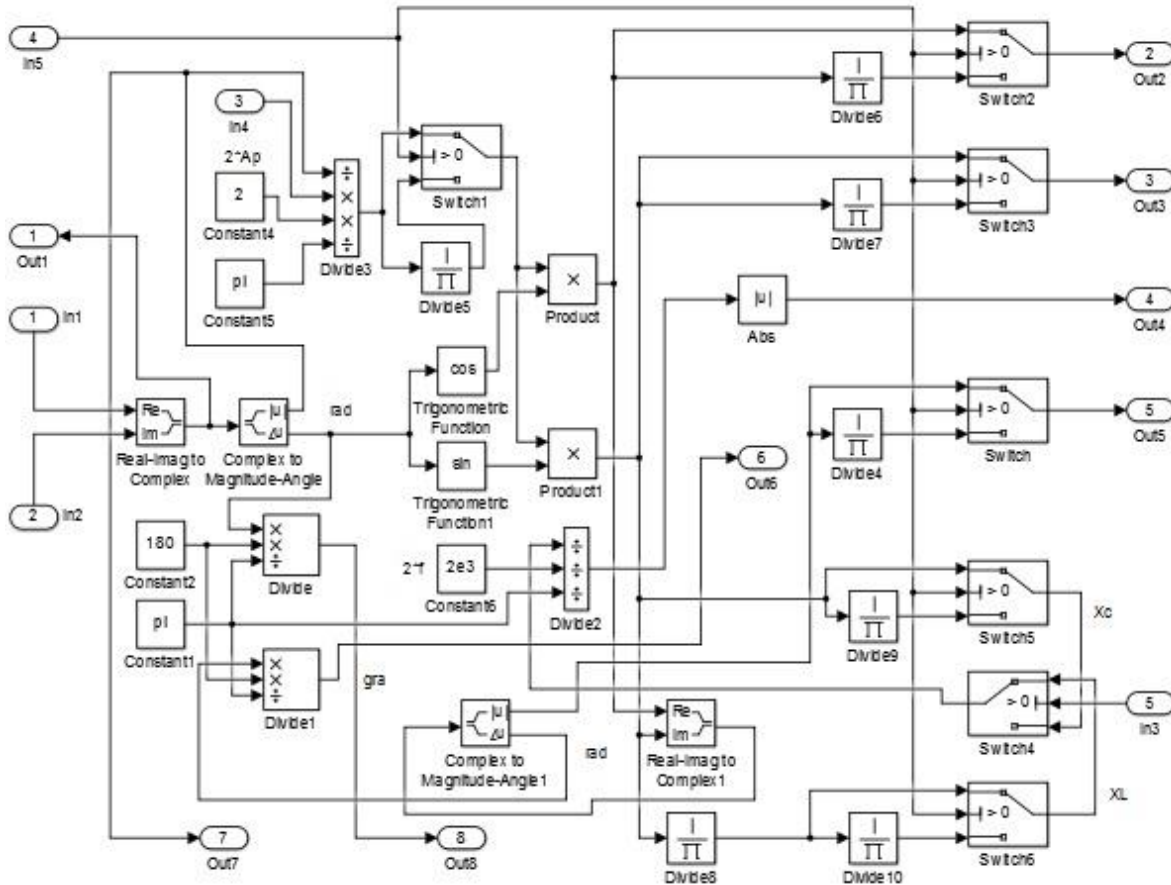


Figura 10. Algoritmo para evaluar las ecuaciones (1) a (9), implementado en el Módulo de Procesamiento Aritmético.

El algoritmo de las operaciones matemáticas se implementó utilizando las librerías con que cuenta Simulink, en el diagrama de operaciones matemáticas es necesario establecer el valor del voltaje pico de la amplitud A_p de la señal V_{ac} multiplicado por dos ($2A_p$) en **Constant4** y el valor de la frecuencia f multiplicado por dos ($2f$) en **Constant6**.

La entrada **In3** determina el circuito para analizar, RC o RL por medio de la condición de **Constant1**, la entrada **In5** determina la conexión del circuito, serie o paralelo por medio de la condición de **Constant2** y en la entrada **In4** se ingresa el valor de la resistencia de ganancia R_f (R_{fs} , R_{fp} , R_{fl}) dependiendo del tipo de circuito eléctrico que se analizará. Las entradas **In1** e **In2** corresponden a los voltajes en fase y cuadratura V_{SDf} y V_{SDc} obtenidos en el **MDSF**. El valor de estos se visualiza en forma rectangular en el **Display2** y en forma polar en los **Display 3 y 4 (amarillos)**. Las variables de R , X_c , X_L , C , L , Z y θ se visualizan en los **Display 5 a 9 (azul agua)**, respectivamente como se muestra en la figura 2.

4 Simulaciones de circuitos eléctricos resistivos-reactivos

Las simulaciones se realizaron siguiendo la siguiente metodología de configuración de parámetros y variables en el SSRCLZ de la figura 11:

- Determinar si es un circuito RC o RL con la condición (-1 o 1) de **Constant1 (en verde)**, respectivamente.
- Determinar si la conexión es en paralelo o serie con la condición (-1 o 1) de **Constant2 (en verde)**, respectivamente.
- Indicar los valores de R, C o L haciendo doble clic en **Subsystem1** o **Subsystem5** y después sobre los símbolos de R, C o L, según sea el caso, figura 3.
- Verificar la amplitud de la fuente de voltaje V_{ac} (**Sine Wave**) sea $A_{PP} = 2V_{PP}$ y tenga una frecuencia de $f \approx 1\text{kHz}$, haciendo doble clic sobre su ícono (naranja).
- Determinar el valor de la resistencia de ganancia R_f de la etapa de amplificación (**Finite-Gain Op-Amp**) para el tipo de circuito y conexión según sea el caso, realizando doble clic en el **MA (Subsystem2)** y después sobre R_f , figura 4.
- Determinar el valor de la resistencia de ganancia R_f de la etapa de amplificación (**Finite-Gain Op-Amp**) para el tipo de circuito y conexión según sea el caso, realizando doble clic en **Constant3 (verde)**, en la entrada **In4** del **MPA (Subsystem6)**.
- Determinar el tiempo de simulación en la ventana **Simulation stop time** de la barra de herramientas a **500e-3** segundos (recomendado), figura 11.
- Seleccionar el botón **Run**, parte superior de la figura 11.
- Los datos se despliegan en los visualizadores **Display**.

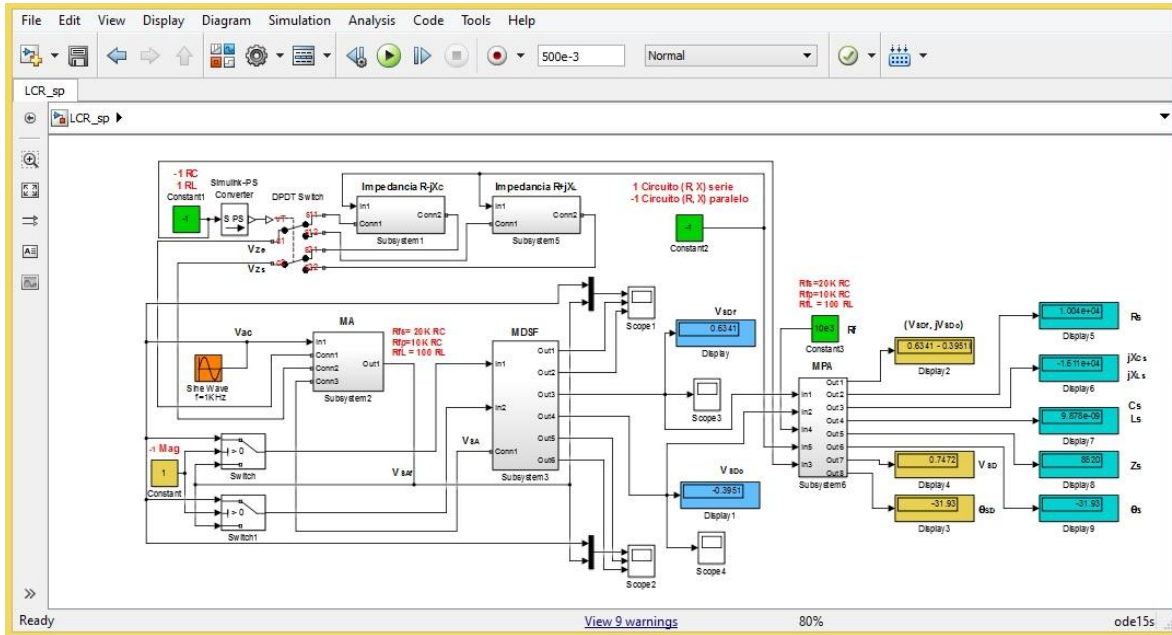


Figura 11. Pantalla completa del SSRCLZ implementado en Simulink, Asur et al. 2021.

5 Resultados obtenidos con el SSRCLZ

Se realizaron simulaciones de circuitos RC y RL conectados en serie y en paralelo, se propusieron 6 valores de R y uno para C y L para cada circuito. Se propuso una amplitud $A_{PP} = 2V_{PP}$ y una frecuencia $f=1\text{kHz}$ para la señal senoidal de excitación del circuito en estudio. Las ganancias se determinaron para que la señal de salida del **MA** se mantuviera dentro del rango dinámico de la fuente de alimentación de los **Op-Amp** ($\pm 5V$). Se realizaron cálculos teóricos de las variables Z_T y θ_T en función de los valores de R , C y L propuestos. Se realizaron simulaciones para calcular los valores de R_S , C_S , L_S , Z_S y θ_S en función de las componentes de voltaje en fase y en cuadratura V_{Sdf} y V_{SDc} obtenidas a la salida del **MDSF**. Se compararon los resultados teóricos y simulados y se evaluó el error relativo para R , C y L como,

$$\%E_{rR} = \left| \frac{R_S - R}{R} \right| \times 100 \quad (10)$$

$$\%E_{rC} = \left| \frac{C_S - C}{C} \right| \times 100 \quad (11)$$

$$\%E_{rL} = \left| \frac{L_S - L}{L} \right| \times 100 \quad (12)$$

Los resultados obtenidos se muestran en la tabla 1, Asur et al. 2021.

Tabla 1. Resultados teóricos y simulados obtenidos con el SSRCLZ implementado en Simulink, Asur et al. 2021 .

Datos iniciales propuestos			Datos teóricos		Datos calculados en función de V_{Sdf} y $-V_{Sdc}$ obtenidos de las simulaciones del SSZRCL							
Circuito RC serie												
f [kHz]	R [k Ω]	C [nF]	$ Z_T $ [k Ω]	θ_T [gra]	V_{Sdf} [V]	$-V_{Sdc}$ [V]	R_S [k Ω]	C_S [nF]	$ Z_S $ [k Ω]	$-\theta_S$ [gra]	%Er R	%Er C
1	5.6	10	16.87	70.62	0.259	0.705	5.85	10	16.95	69.8	4.46	0
1	10	10	18.8	57.86	0.369	0.567	10.26	10.08	18.82	56.98	2.6	0.8
1	20	10	25.56	38.51	0.397	0.306	20.1	10.28	25.37	37.6	0.5	2.8
Circuito RC paralelo												
1	5.6	10	5.28	19.38	1.131	0.398	5.63	9.94	5.31	19.36	0.53	0.6
1	10	10	8.5	32.14	0.634	0.395	10.04	9.89	8.52	31.93	0.4	1.1
1	20	10	12.45	51.5	0.32	0.396	19.92	9.9	12.51	51.09	0.4	1
Circuito RL serie												
f [kHz]	R [Ω]	L [mH]	$ Z_T $ [Ω]	θ_T [gra]	V_{Sdf} [V]	V_{Sdc} [V]	R_S [Ω]	L_S [mH]	$ Z_S $ [Ω]	θ_S [gra]	%Er R	%Er L
1	50	10	80.3	51.49	0.492	0.614	50.57	10.05	80.91	51.31	1.14	0.5
1	100	10	118.1	32.14	0.453	0.285	100.7	10.08	118.9	32.17	0.7	0.8
1	200	10	209.6	17.44	0.287	0.09	202.1	10.12	211.9	17.47	0.05	1.2
Circuito RL paralelo												
1	50	10	39.12	38.51	1.267	0.99	50.23	10.23	39.57	38.02	0.46	2.3
1	100	10	53.21	57.86	0.635	0.994	100.3	10.1	53.96	57.44	0.3	1
1	200	10	59.94	72.56	0.321	0.996	198.2	10.17	60.81	72.13	0.1	1.7
Datos iniciales propuestos			Datos teóricos		Datos calculados en función de V_{Sdf} y V_{Sdc} obtenidos de las simulaciones del SSZRCL							

6 Conclusiones

Los resultados obtenidos con el SSRCLZ con respecto de los obtenidos teóricamente se mantienen dentro del 5% lo cual puede deberse a la solución del algoritmo de los modelos de los componentes que conforman las diferentes etapas del sistema propuesto. Sin embargo, es una buena aproximación y muestra que la solución del algoritmo del sistema esta basado en las ecuaciones derivadas de un análisis de función de transferencia del sistema completo. Los valores de los componentes de los circuitos eléctricos se eligieron en función de la ganancia del MA y del tipo de conexión para que no rebasara el rango dinámico del Op-Amp ($\pm 5V$).

El SSRCLZ puede ser una herramienta didáctica para la enseñanza de los temas relacionados en instrumentación electronica y circuitos eléctricos para carreras de Ingeniería Eléctrica a nivel de Licenciatura y Posgrado en las modalidades de clases presenciales y a distancia.

ANEXO

Introducción a Simulink

El programa de MathWorks para simulación (modelación y análisis) de sistemas dinámicos no lineales fue presentado en 1990, con el nombre de SIMULAB para computadoras personales y con el nombre de SIMULINK para estaciones de trabajo. Su aparición estuvo unida a la primera versión de MATLAB para Windows. Desde mayo de 1994, que está disponible la versión 1.3, SIMULINK tiene un tratamiento similar a los otros Toolboxes de MATLAB, en el sentido que se instala de forma separada, pero sigue siendo la mejor herramienta para aprovechar toda la potencia de MATLAB y de los otros "Toolboxes".

Simulink es una herramienta interactiva para modelar, simular y analizar sistemas dinámicos. Nos permite construir diagramas de bloque gráficos, evaluar el rendimiento del sistema y refinar sus diseños. Simulink está firmemente integrado con Stateflow para modelar comportamiento even-driven. Simulink es la herramienta a escoger para el diseño de sistemas de control, diseños DSP, diseños de sistemas de comunicaciones y otras aplicaciones de simulación. Como una extensión de Matlab, Simulink adiciona muchas características específicas a los sistemas dinámicos, mientras conserva toda la funcionalidad de propósito general de Matlab. Así Simulink no es completamente un programa separado de Matlab, sino un anexo a él. El ambiente de Matlab está siempre disponible mientras se ejecuta una simulación en Simulink. Simulink tiene dos fases de uso: la definición del modelo y el análisis del modelo. La definición del modelo significa construir el modelo a partir de elementos básicos construidos previamente, tal como, integradores, bloques de ganancia o servomotores. El análisis del modelo significa realizar la simulación, linealización y determinar el punto de equilibrio de un modelo previamente definido.

Para simplificar la definición del modelo Simulink usa diferentes clases de ventanas llamadas ventanas de diagramas de bloques. En estas ventanas se puede crear y editar un modelo gráficamente usando el mouse. Simulink usa un ambiente gráfico lo que hace sencillo la creación de los modelos de sistemas. Después de definir un modelo este puede ser analizado seleccionando una opción desde los menús de Simulink o entrando comandos desde la línea de comandos de Matlab.

Simulink puede simular cualquier sistema que pueda ser definido por ecuaciones diferenciales continuas y ecuaciones diferenciales discretas. Esto significa que se puede modelar sistemas continuos en el tiempo, discretos en el tiempo o sistemas híbridos.

Simulink usa diagramas de bloques para representar sistemas dinámicos. Mediante una interfase gráfica con el usuario se pueden arrastrar los componentes desde una librería de bloques existentes y luego interconectarlos mediante conectores.

Algunas Características Principales:

Facilidad de Uso:

Una biblioteca extensa de bloques predefinidos para construir modelos gráficos de su sistema.

El Debugger gráfico de Simulink para localizar y diagnosticar fallos en el modelo.

El Visualizador de modelos para navegar por las jerarquías del modelo.

Utilidad gráfica para buscar modelos y bibliotecas.

Bloques personalizables que pueden incorporar código ya existente del usuario en C, Ada, MATLAB, y Fortran.

Ayuda para aplicaciones más grandes:

Sistemas lineales, no lineales, tiempo continuo, tiempo discreto, multirate, ejecutados condicionalmente, e híbridos.

Los modelos pueden agruparse en jerarquías para crear una vista simplificada de los componentes o los subsistemas.

Los objetos de datos de Simulink le permiten crear tipos de datos de MATLAB específicos de la aplicación para los modelos de Simulink.

Simulink Explorer GUI para visualizar y editar objetos de datos de Simulink.

Visualizador de la biblioteca para seleccionar los bloques en una manera conveniente.

Protección de propiedad intelectual utilizando funciones S (requiere Real-Time Workshop 4.0)

Puede ejecutar las simulaciones desde la línea de comando de MATLAB, de forma interactiva o en modo batch.

Ayuda Computacional:

Ayuda para las señales y las operaciones de matrices.

El bloque Bitwise Logical Operator enmascara, invierte, o desplaza los bits de una señal entero y sin signo en una manera lógica

Acelerador de Simulink

Para incrementar la velocidad de Simulink se debe instalar el acelerador "Accelerator". Este permite automáticamente generar una versión mejorada de los modelos los cuales correrán diez veces más rápido que el original. El acelerador puede ser usado sobre modelos continuos, discretos en el tiempo e híbridos.

El acelerador trabaja generando y compilando un código-C para un modelo dado. Una vez se completa la compilación, la simulación es ejecutada en la ventada de modelos de Simulink exactamente igual que antes sólo que más rápidamente. El propósito del acelerador es aumentar la velocidad de simulación.

Si el programa MatLab posee instalado el "Accelerator" podrá iniciarse la acción aceleradora seleccionando la opción simulation en el menú principal del Simulink y dentro de esta seleccionando la opción Accelerate. Esta acción es totalmente transparente en el sentido de que el incremento de la velocidad se presenta sin ningún otro requerimiento por parte del usuario.

Generador de código-C en Simulink

Una vez se ha creado un modelo dinámico en Simulink, se puede invocar el generador de código-C que permite convertir el diagrama de bloques implementado en un código C. Este puede ser útil para varios propósitos: puede ser usado para control en tiempo real, simulación en tiempo real o simulación acelerada en tiempo no real. Sus aplicaciones pueden ser control de movimiento, control de procesos, sistemas automotores, equipos médicos, robótica, etc.

El código-C es diseñado tal que puede ser ejecutado en tiempo real. No requiere ser escrito manualmente por un programador pues es creado a nivel de diagramas de bloques en Simulink. El código generado puede correr sobre un amplio rango de hardware ubicado en estaciones de trabajo, PC o microprocesadores. Este código es la forma en la que puede usarse el Simulink para adquisición de datos.

Construcción de modelos

Simulink le permite crear modelos y simulaciones de prácticamente cualquier tipo de sistema dinámico del mundo real. Usando las potentes funciones de simulación de Simulink, puede crear modelos, evaluar diseños y corregir imperfecciones en éstos antes de construir los prototipos.

Algunas recomendaciones para construir modelos:

Use sistemas jerárquicos: Los modelos complejos se benefician de la incorporación de jerarquías usando subsistemas. La agrupación de bloques simplifica el nivel superior del modelo haciéndolo fácil de entender.

Cree modelos claros: Modelos bien organizados y documentados son fáciles de entender e interpretar. Para lo anterior, utilice etiquetas para las señales y anotaciones sobre el modelo.

Diseñe el modelo en un papel y después use el ordenador, coloque primero los bloques que se van a utilizar y luego realice las conexiones, esto último reduce el tiempo de diseño a través de sucesivas aperturas de las bibliotecas de bloques.

Los elementos fundamentales para construir modelos con Simulink son: Líneas, Bloques y Subsistemas.

Qué hace el Simulink

Simulink permite definir de una forma lógica las diferentes etapas del proceso de cálculo y el flujo de datos entre ellas. Cada bloque puede representar un solo elemento del proceso o bien un subsistema, además de ser fácilmente modificable para reflejar un cambio en el algoritmo o el enfoque del diseño. Estos diagramas de bloques permiten una descripción en alto nivel del sistema además de ser fácilmente modificables con la finalidad de conseguir el comportamiento deseado, es decir, proporcionan una estructura jerárquica para evaluar el comportamiento de algoritmos alternativos bajo diferentes condiciones de funcionamiento.

En los grupos de investigación y desarrollo, Simulink permite que cada persona integrante del mismo pueda entender y modificar de una forma sencilla los detalles del algoritmo y el progreso hacia la fase de implementación.

Simulink incluye muchas características para el análisis detallado de sistemas. Las capacidades principales incluyen: linealización, determinación de puntos de equilibrio, animación, optimización de los parámetros, y análisis paramétrico.

Extracción de Modelos Lineales

Las dinámicas de diagramas de bloques no lineales pueden ser aproximadas por linealización, permitiendo aplicar técnicas de diseño que requieran representaciones de modelos lineales. Puede usar la función `linmod` de Simulink para conseguir modelos lineales del espacio de estados de su diagrama de bloques.

Animación

Simulink proporciona acceso inmediato a los potentes gráficos 2-D y 3-D y capacidades de animación de MATLAB. Puede usar MATLAB para mejorar las visualizaciones y entender mejor el comportamiento de su sistema durante el progreso de la simulación. Para acceder a las capacidades de animación de MATLAB, usted puede usar las funciones S de Simulink para incorporar comandos gráficos estándar de MATLAB.

Análisis Paramétrico

Con Simulink, el proceso de diseño puede ser automatizado con el uso de archivos con comandos de MATLAB (M-files) para ejecutar simulaciones múltiples con parámetros variables. Los gráficos resultantes muestran familias de curvas que ilustran la respuesta temporal en función de los parámetros variados. Puede también mejorar el rendimiento del sistema con el uso de funciones contenidas en el Toolbox de Optimización o el Blockset de Diseño de Control no Lineal para sintonizar los parámetros del modelo.

Integración con MATLAB

Dado que Simulink está construido a partir de MATLAB, proporciona un entorno único de desarrollo. Este sistema permite ejecutar las simulaciones de manera interactiva, con el uso de las interfaces gráficas de Simulink, o de manera sistemática, con la ejecución de conjuntos de experimentos en el modo batch desde la línea de comandos de MATLAB. Entonces puede generar vectores de prueba y analizar los resultados colectivamente, *Cubillos (2013)*.

Principios de operación y gestión de Simulink

Para comenzar Simulink, se debe arrancar primero MATLAB. En el indicador de MATLAB, se introduce la orden Simulink. El computador incluirá la ventana de comandos de MATLAB y la ventana Simulink Library Browser (SLB), figura A1, *Ataurima (2013)*.

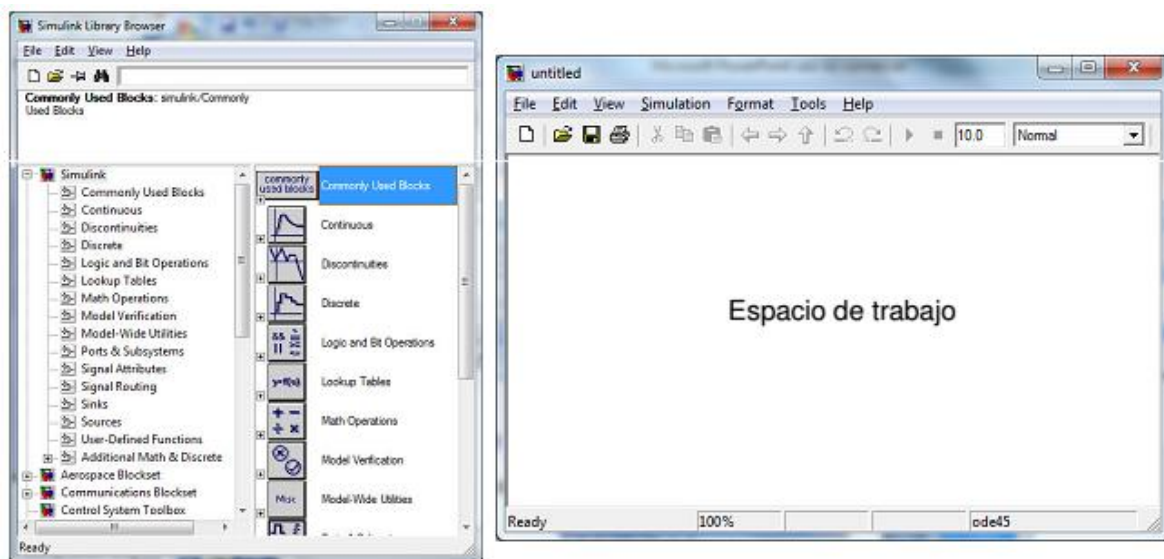


Figura A1. Ventana de Simulink Library Browser y ventana de trabajo.

La ventana Simulink Library Browser visualiza los íconos de sus bibliotecas de bloque. Construye modelos copiando bloques de la biblioteca de bloques en una ventana de trabajo.

Cuando se ejecuta una simulación y se analizan sus resultados, se puede utilizar comandos de MATLAB que se introducen desde la ventana de comandos de MATLAB. Simulink utiliza ventanas separadas para visualizar una biblioteca de bloques, un modelo o la salida de una simulación gráfica. Estas ventanas no son ventanas de figura de MATHLAB y no se pueden manipular utilizando los comandos del entorno gráfico de MATLAB.

Los menús de Simulink aparecen cerca de la parte superior de cada ventana Simulink. Los comandos del menú se aplican a los contenidos de esa ventana.

Un error frecuente de los nuevos usuarios de Simulink es comenzar una simulación mientras el Simulink Block Library es la ventana activa. Hay que asegurarse de que la ventana del modelo es la ventana activa antes de comenzar una simulación.

Para finalizar una sesión de Simulink se escoge Exit MATLAB en el menú File.

Construcción de un modelo

Para crear un nuevo modelo, se escoge la orden New en el menú File. Simulink crea una nueva ventana. Se puede mover la ventana de la misma forma que se hace con otras ventanas.

Para editar el diagrama de un modelo existente, se debe hacer una de las dos acciones siguientes:

Escoger la orden Open en el menú File especificar el archivo-M que describe el modelo que se desea editar.

Introducir el nombre del modelo en la ventana de orden de Matlab.

Simulink crea una nueva ventana y visualiza ese modelo en la ventana.

Selección de objetos

Muchas acciones de edición y construcción de modelos requieren que primero se seleccione uno o más bloques y líneas (objetos), figura A2.



Figura A2. Selección del bloque Sine Wave para utilizarlo en la ventana de trabajo.

Cuando selecciona un objeto pulsando encima de él, cualquier otro que lo estuviera deja de estarlo.

Selección de más de un objeto

Puede seleccionar más de un objeto seleccionándolos uno a uno o todos a la vez si están próximos utilizando un recuadro que los englobe.

Se selecciona una librería para desplegar los bloques contenidos, se seleccionan los bloques requeridos y se arrastran a la ventana de trabajo de Simulink, figura A3.

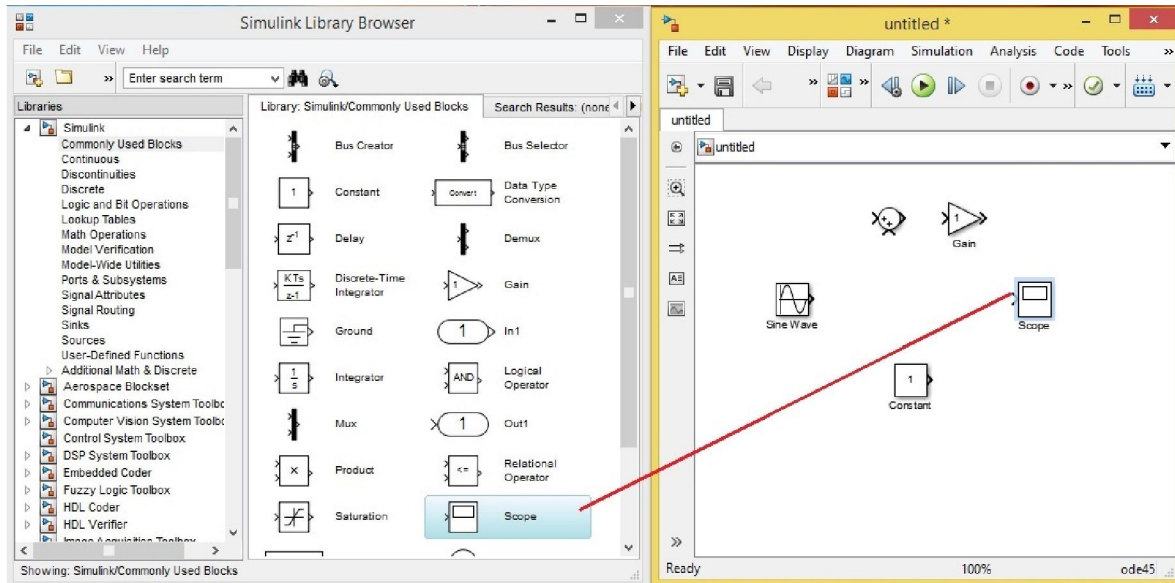


Figura A3. Selección de diferentes bloques del Simulink Library Browser para colocarlos en la ventana de trabajo.

Para interconectar los bloques se selecciona la salida de uno y se arrastra hasta la entrada del siguiente siguiendo el orden del flujo de datos y así hasta interconectar todos los bloques necesarios para realizar la simulación del modelo, figura A4.

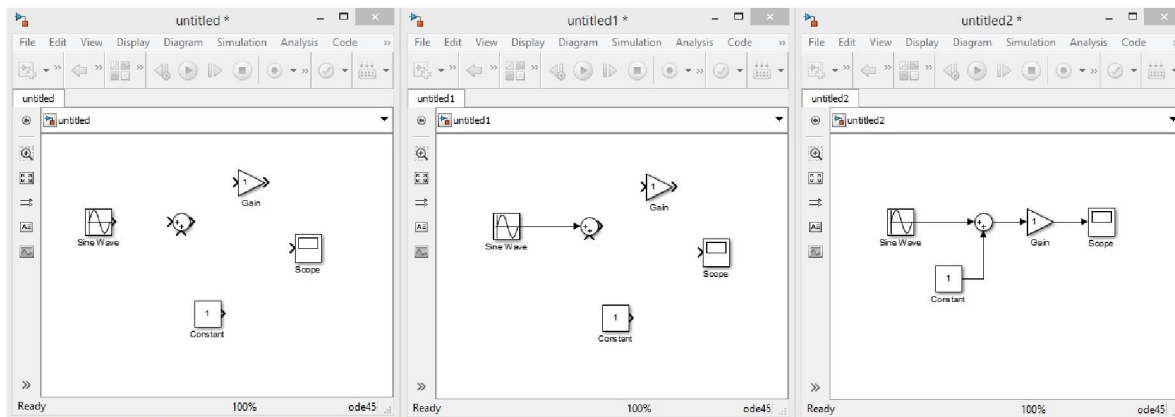


Figura A4. Interconexión de bloques que conforman un modelo en la ventana de trabajo de Simulink.

De esta forma se van añadiendo objetos y/o bloques interconectados hasta completar el modelo en estudio.

Elementos de un modelo en Simulink

- Entradas, Sources o Inputs
- Constantes, generadores de funciones (ondas senoidales, escalón o señales creadas en Matlab)
- Sistema modelado, representado por el diagrama de bloques
- Salidas, Sinks u outputs
- Gráficos, osciloscopios, ficheros

Pasos para Crear un nuevo modelo en Simulink

- Abrir la ventana de un nuevo modelo
- Conectar los bloques
- Añadir bloques
- Cambiar el tamaño de los bloques
- Modificar etiquetas y añadir anotaciones
- Parametrizar los bloques y la simulación
- Ejecutar y modificar
- Guardar el modelo (formato Simulink y .m)

Referencias Bibliográficas

Asur Guadarrama Santana, Daniel Martínez, Jorge Uc Martín. (2021). *Simulación de un sistema electrónico de medición RCL e impedancia didáctico*. Memorias del SOMI XXXV Congreso de Instrumentación, 27-29 de octubre de 2021, ISSN 2395-8499.

Francisco Cubillos M. (2013), Apuntes Introducción a Simulink,
https://www.labcontrol.cl/sites/labcontrol/files/apuntes_simulink_2013_0.pdf

Miguel Ataurima Arellano (2013), MATLAB & Simulink para Ingeniería: Nivel 1, 1ª Edición, eBook