



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

TEMA:

**Instrumentación virtual utilizando LabView para la asignatura de  
señales y sistemas en la carrera de ingeniería en telecomunicaciones.**

AUTOR:

Escalante Bacilio, Krissy Stefany

Trabajo de Titulación previo a la obtención del grado de  
**INGENIERO EN TELECOMUNICACIONES**

TUTOR:

Zamora Cedeño, Néstor Armando

Guayaquil, Ecuador

13 de Marzo del 2017



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

**CERTIFICACIÓN**

Certificamos que el presente trabajo fue realizado en su totalidad por la Srta.  
**Escalante Bacilio, Krissy Stefany** como requerimiento para la obtención  
del título de **INGENIERO EN TELECOMUNICACIONES**.

TUTOR

---

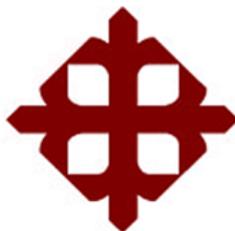
Zamora Cedeño, Néstor Armando

DIRECTOR DE CARRERA

---

Heras Sánchez, Miguel Armando

Guayaquil, a los 13 del mes de marzo del año 2017



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

**DECLARACIÓN DE RESPONSABILIDAD**

Yo, **Escalante Bacilio, Krissy Stefany**

**DECLARO QUE:**

El trabajo de titulación **“Instrumentación virtual utilizando LabView para la asignatura de señales y sistemas en la carrera de ingeniería en telecomunicaciones”** previo a la obtención del Título de **Ingeniero en Telecomunicaciones**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

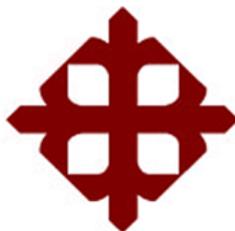
En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 13 del mes de Marzo del año 2017

EL AUTOR

---

ESCALANTE BACILIO, KRISSY STEFANY



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

**AUTORIZACIÓN**

Yo, **Escalante Bacilio, Krissy Stefany**

Autorizó a la Universidad Católica de Santiago de Guayaquil, la publicación, en la biblioteca de la institución del Trabajo de Titulación: “**Instrumentación virtual utilizando LabView para la asignatura de señales y sistemas en la carrera de ingeniería en telecomunicaciones**”, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, a los 13 del mes de Marzo del año 2017

EL AUTOR

---

ESCALANTE BACILIO, KRISSEY STEFANY

# REPORTE DE URKUND

**URKUND**

<b>Documento</b>	<a href="#">ProyectoTitulacionEscalante.docx</a> (D26005175)
<b>Presentado</b>	2017-02-24 14:18 (-05:00)
<b>Presentado por</b>	Néstor Zamora (nestor.zamora@cu.ucsg.edu.ec)
<b>Recibido</b>	nestor.zamora.ucsg@analysis.orkund.com
<b>Mensaje</b>	Analisis Proyecto Titulacion Escalante <a href="#">Mostrar el mensaje completo</a> 0% de esta aprox. 20 páginas de documentos largos se componen de texto presente en 0 fuentes.

## **DEDICATORIA**

En primer lugar y por sobre todo dedico este trabajo de titulación a Dios, el ser supremo que me ha permitido llegar hasta aquí, por darme las fuerzas necesarias para no abandonar mis metas, para luchar y cumplirlas.

A mi madre, Janeth Bacilio quien siempre me ha apoyado incondicionalmente.

A mi tío Jui Micol, por hacer el papel de padre, por su total apoyo tanto emocional como económico quedo totalmente agradecida de ello.

A mis tías, Celinda Bacilio y Vanessa Bacilio, por siempre estar empujándome hacia adelante, por los innumerables consejos brindados a lo largo de mi vida.

Y no puedo dejar de dedicar este trabajo a mis hermanos, Thiago Escalante y Fergie Carbo, a ellos por ser los motores que me han impulsado a obtener este título de Ingeniera en Telecomunicaciones.

**EL AUTOR**

**ESCALANTE BACILIO, KRISSY STEFANY**

## **AGRADECIMIENTO**

Agradezco a Dios por la bendición de estar viva, de tener salud y de estar cumpliendo la primera de las muchas metas que me he propuesto, por permitirme crecer con personas tan maravillosas como lo son mi familia.

A mi familia, por su apoyo infinito, tanto económico como emocional, gracias a ellos este título se hace posible.

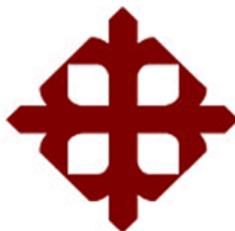
A los docentes que llegaron a compartir sus conocimientos conmigo, que de una u otra forma me motivaron a seguir soñando con ser Ingeniera en tan hermosa carrera.

A mi grupo de amigos: Michael, Eddie, Erick, Terry y Jhon, más que amigos son mi segunda familia, gracias por el apoyo y los ánimos para seguir mi objetivo.

Especial agradecimiento a Michael Rochina, por tan maravillosa amistad en estos años de estudio.

EL AUTOR

ESCALANTE BACILIO, KRISY STEFANY



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

**TRIBUNAL DE SUSTENTACIÓN**

---

**ZAMORA CEDEÑO, NESTOR ARMANDO**  
TUTOR

---

**HERAS SANCHEZ, MIGUEL ARMANDO**  
DIRECTOR DE CARRERA

---

**PALACIOS MELÉNDEZ, EDWIN FERNANDO**  
COORDINADOR DE TITULACIÓN

## Índice General

Índice de Figuras .....	XII
Índice de Tablas.....	XV
Resumen.....	XVI
Abstract .....	XVII
CAPÍTULO 1: INTRODUCCIÓN.....	2
1.1. Introducción.....	2
1.2. Antecedentes.....	3
1.3. Justificación del Problema.....	3
1.4. Definición del Problema.....	3
1.5. Objetivos del Problema de Investigación.....	4
1.5.1. Objetivo General.....	4
1.5.2. Objetivos Específicos.....	4
1.6. Hipótesis.....	4
1.7. Metodología de Investigación.....	5
CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA DE SEÑALES Y SISTEMAS.....	6
2.1. ¿Qué es una señal? .....	6
2.1.1. Señal continua.....	7
2.1.2. Señal discreta.....	7
2.2. Señales Básicas de Tiempo Continuo.....	8
2.2.1. Función Escalón Unitario.....	8
2.2.2. Función Impulso Unitario.....	9
2.3. Señales Básicas de Tiempo Discreto.....	10
2.3.1. Señal Sinusoidal.....	10
2.3.2. Secuencia Escalón Unitario.....	10
2.4. Sistemas.....	11

2.4.1.	Sistema Continuo .....	11
2.4.2.	Sistema Discreto .....	12
2.4.3.	Muestreo de señal análoga .....	13
2.5.	Funciones trigonométricas .....	14
2.5.1.	Funciones trigonométricas básicas .....	14
2.6.	Introducción a series de Fourier .....	15
2.6.1.	Coeficientes de Fourier .....	16
2.6.2.	Series de Fourier .....	16
2.7.	Transformada de Fourier .....	17
2.7.1.	Transformada de Fourier en tiempo continuo .....	18
2.7.2.	Transformada de Fourier en tiempo discreto .....	18
2.7.3.	Transformada inversa de Fourier en tiempo discreto .....	18
2.8.	Transformada de Laplace .....	20
2.8.1.	Definición de la transformada de Laplace .....	20
2.8.2.	Transformada de Laplace de funciones básicas .....	21
2.8.3.	Transformada inversa de Laplace .....	21
2.9.	Transformada Z .....	21
2.9.1.	Definición de la transformada z .....	22
2.9.2.	Transformada z de funciones básicas .....	22
2.10.	Filtros Analógicos.....	23
2.10.1.	Tipos de filtros analógicos .....	23
2.11.	Filtros digitales .....	25
2.11.1.	Tipos de filtros digitales .....	27
CAPÍTULO 3: INTRODUCCIÓN A LABVIEW .....		29
3.1.	Entorno .....	30
3.1.1.	Panel frontal.....	31
3.1.2.	Diagrama de bloques .....	34

3.2.	Programación <i>LabView</i> .....	40
3.3.	Ejecución <i>LabView</i> .....	41
CAPÍTULO 4: SIMULACION Y RESULTADOS OBTENIDOS .....		42
4.1.	Simulación 1. Gráfica de la función <i>Sen.</i> .....	42
4.2.	Simulación 2. Transformada de Fourier .....	52
4.3.	Simulación 3. Teorema de Parseval .....	59
4.4.	Simulación 4. Modulador FM .....	64
CAPÍTULO 5: CONCLUSIONES Y RECOMENDACIONES. ....		66
5.1.	Conclusiones.....	66
5.2.	Recomendaciones. ....	67
REFERENCIAS BIBLIOGRÁFICAS .....		68

## Índice de Figuras

### CAPÍTULO 2

Figura 2.1: Muestreo de una señal analógica .....	7
Figura 2.2: Muestreo de una señal discreta .....	8
Figura 2.3: Función escalón unitario .....	9
Figura 2.4: Función Impulso Unitario .....	9
Figura 2.5: Amplitud de una señal sinusoidal .....	10
Figura 2.6: Secuencia Escalón Unitario .....	11
Figura 2.7: Circuito RC .....	12
Figura 2.8: Aplicaciones de sistema discreto .....	13
Figura 2.9: Muestreo de señal análoga .....	13
Figura 2.10: Representación gráfica de series de Fourier .....	17
Figura 2.11: Muestreo de una señal $x[n]$ y relación entre DFT e IDFT. ....	19
Figura 2.12: Tipos de filtros análogos. ....	23
Figura 2.13: Esquema de diseño de un filtro digital. ....	26
Figura 2.14: Diagrama de un filtro digital de 150khz. ....	27
Figura 2.15: Muestras de una señal entrante y saliente en un filtro FIR. ....	28
Figura 2.16: Muestras de una señal entrante y saliente en un filtro IIR. ....	28

### CAPÍTULO 3

Figura 3. 1: Panel frontal. ....	31
Figura 3. 2: Paleta de controles .....	32
Figura 3. 3: Controles e Indicadores de cadena de caracteres .....	33
Figura 3. 4: Controles e Indicadores booleano .....	33
Figura 3. 5: <b>1.</b> Incremento/Reducción de control numérico. <b>2.</b> Control. <b>3.</b> Indicador .....	34
Figura 3. 6: Diagrama de bloques .....	35
Figura 3. 7: Terminales de indicador, cables, nodos, terminales de control. ....	35
Figura 3. 8: Panel Frontal del Área de un triángulo .....	36
Figura 3. 9: Control, Indicador y Constante .....	37
Figura 3. 10: Detalle de las estructuras .....	38
Figura 3. 11: Panel conector .....	38
Figura 3. 12: Paleta de funciones .....	40

## CAPÍTULO 4

Figura 4. 1: Nuevo VI .....	42
Figura 4. 2: Panel frontal. ....	42
Figura 4. 3: Panel frontal – control “graph” . ....	43
Figura 4. 4: Control “waveform graph”.....	43
Figura 4. 5: Front panel y Block Diagram – control numeric “numeric control”. .....	44
Figura 4. 6: Controles numéricos ingresados. ....	45
Figura 4. 7: Estructura “For Loop” ingresada. ....	45
Figura 4. 8: Función trigonométrica “sin”.....	46
Figura 4. 9: Conexiones de indicadores. ....	46
Figura 4. 10: Función sen con un valor de 100 en número de puntos. ....	47
Figura 4. 11: Constante $2\pi$ agregada en la estructura for loop. ....	48
Figura 4. 12: Funciones “división” y “multiply” . ....	49
Figura 4. 13: Función sen con amplitud 1. ....	50
Figura 4. 14: Función sen con amplitud 4. ....	51
Figura 4. 15: Estructura while loop .....	52
Figura 4. 16: Creación de un control. ....	53
Figura 4. 17: Función “simulator signal”. ....	53
Figura 4. 18: Configuración del simulador de señal. ....	54
Figura 4. 19: Configuración del segundo simulador de señal. ....	55
Figura 4. 20: Función “spectral”.....	56
Figura 4. 21: Configuración del espectro. ....	56
Figura 4. 22: Ventana “block diagram” .....	57
Figura 4. 23: Ventana “front panel” .....	57
Figura 4. 24: Ventana “block diagram” con las conexiones realizadas. ....	58
Figura 4. 25: Resultado final “transformada de Fourier”. ....	58
Figura 4. 26: Funciones “sine pattern” y “uniform noise” - diagrama de bloque.....	59
Figura 4. 27: DBL numeric control - diagrama de bloque.....	60
Figura 4. 28: funciones numéricas - diagrama de bloque. ....	60
Figura 4. 29: Funciones en diagrama de bloque. ....	61
Figura 4. 30: Conversion “to double precision float” .....	61

Figura 4. 31: Numeric constant aplicado. ....	62
Figura 4. 32: Teorema de parseval – diagrama de bloque. ....	63
Figura 4. 33: Teorema de parseval – Front panel. ....	64
Figura 4. 34: Modulador FM en diagrama de bloque. ....	65

## Índice de Tablas

### CAPÍTULO 2

Tabla 2. 1: Características de una señal .....	6
Tabla 2. 2: Transformada de Laplace de funciones básicas.....	21
Tabla 2. 3: Transformada z de funciones básicas. ....	22

## Resumen

El presente trabajo de titulación tiene como propósito dar solución a la complejidad que puede tener la materia de señales y sistemas para los alumnos de la facultad técnica de la Universidad Católica Santiago de Guayaquil, mediante el uso de un software que es ampliamente utilizado en áreas de ingeniería, se busca hacer que la asignatura se torne más sencilla, con la ayuda de este software se invita a interactuar, realizar simulaciones y tener una más amplia idea de donde aplicar conocimientos dictados en clase. Existen varias herramientas para simular señales y procesamiento de las señales como lo son MatLab/Simulink y LabView, ambas son herramientas útiles para el diseño de sistemas, en este trabajo de titulación la plataforma escogida fue LabView que dispone de diversos instrumentos virtuales. Mediante la fundamentación teórica se incentiva al lector a conocer principios básicos de señales y sistemas, se ha tomado como ejemplo el teorema de Parseval, la modulación FM y transformada de Fourier para mostrar al lector la manera en que LabView puede graficar tanto nociones básicas como algunas más estructuradas.

**Palabras claves:** SEÑALES Y SISTEMAS, LABVIEW, PROCESAMIENTO DE SEÑALES, TEOREMA DE PARSEVAL, MODULACION FM, TRANSFORMADA DE FOURIER

## **Abstract**

The main purpose of the present work is to bring a solution to the complicity that can have the subject Signals and Systems to the students of Universidad Católica de Santiago de Guayaquil. Through the use a software that is commonly used in engineering, it is sought to make the subject easier for the student, with the help of this, we encourage to interact, perform simulations and to have a better idea in where to apply the knowledge taken from the class. There are various tools to perform signals simulations such as MatLab/Simulink and LabView, both are useful tools for the design of systems, in this work the platform chosen was LabView because of the several virtual instruments. The theory encourage the reader to know the basic principles of signals and systems, the Parseval theorem, the FM modulation and the Fourier transform are examples taken to show the reader how LabView can graph both basic notions and some more structured.

**Keywords:** SIGNALS AND SYSTEMS, LABVIEW, SIGNAL PROCESSING, PARSEVAL THEOREM, MODULATION FM, FOURIER TRANSFORM

## **CAPÍTULO 1: INTRODUCCIÓN**

### **1.1. Introducción.**

Las telecomunicaciones engloban un campo muy amplio, partiendo de su base, se define telecomunicación como una conexión en donde más de un dispositivo busca transmitir y recibir señales utilizando como medio el espectro radioeléctrico. En este punto es donde nacen las señales y se observa la importancia de ellas en una comunicación; es por esto que la asignatura de señales y sistemas es una materia esencial y necesaria para la formación de ingenieros en telecomunicaciones.

La asignatura señales y sistemas, interactúa en el área de las telecomunicaciones y no es únicamente en dicha área donde ha adquirido importancia; también en otros campos del conocimiento, tales como aeronáutica, astronáutica, acústica, sismología, ingeniería biomédica, medicina en general, procesos químicos y el procesamiento de la voz. Teniendo así todas estas disciplinas características en común y son las señales (Alonso & Kuhlmann, 2017).

En cada una de esas áreas del conocimiento, las señales utilizadas son de distinta naturaleza: en acústica se trata de señales generadas por fuentes de sonido como la voz, la música o cualquier clase de ruido; en control de procesos pueden ser señales de tipo térmico, mecánico o eléctrico generadas por los procesos mismos; en medicina pueden ser señales eléctricas o

magnéticas generadas por el organismo humano; en sismología se trata de señales mecánicas, es decir, movimientos de la corteza terrestre.

## **1.2. Antecedentes.**

La materia señales y sistemas es una materia base para la formación de ingenieros en Telecomunicaciones, dado que no se han encontrado trabajos de titulación que relacionan implementación virtual con la materia de señales y sistemas, tan solo con el procesamiento de señales digitales, hace necesario contar con trabajos que se vinculen con esta área de las telecomunicaciones.

## **1.3. Justificación del Problema.**

En la actualidad contar con un laboratorio para cada asignatura de la carrera es un reto, sin embargo es necesario para el desarrollo de los profesionales en formación, el uso de herramientas que le permitan al estudiante simular procesos y vincularse con la carrera sería de gran ayuda, el presente trabajo se presta para ello, utilizando la herramienta de instrumentación virtual LabView.

## **1.4. Definición del Problema.**

En la Facultad Técnica para el Desarrollo (FETD), la materia de señales y sistemas no dispone de un laboratorio para que los estudiantes realicen prácticas del contenido aprendido en clase, con éste trabajo se busca cubrir la necesidad de utilizar un programa que permita la simulación de sistemas y

de apertura a la realización de proyectos que contribuyan con el crecimiento académico de los estudiantes.

## **1.5. Objetivos del Problema de Investigación.**

### **1.5.1. Objetivo General.**

Realizar simulaciones de sistemas y señales de comunicaciones con la instrumentación virtual LabView.

### **1.5.2. Objetivos Específicos.**

- Describir las funciones, entornos y bibliotecas de la herramienta de programación virtual LabView.
- Evaluar los diferentes entornos en los que pueda ser útil la herramienta LabView.
- Desarrollar los entornos de programación G para la instrumentación virtual de las señales.

## **1.6. Hipótesis.**

La simulación de procesos de señales por medio del programa LabView permitirá a los estudiantes diseñar un entorno de sistemas en el que podrán desarrollar los conocimientos en la asignatura de señales y sistemas permitiendo así el desarrollo de proyectos útiles, en tiempo real y minimizando costos que la utilización de otros dispositivos puedan implicar.

## **1.7. Metodología de Investigación.**

Uno de los métodos más importantes en una investigación científica es el método empírico, este método conlleva a la observación, la medición y el experimento (Velasco, 2017). El tema propuesto del trabajo de titulación es la implementación virtual del software LabView, por lo cual se determinó que el método investigativo es del tipo empírico, ya que conlleva a la observación de los resultados obtenidos dentro del programa para luego ser analizados e interpretados.

Otra metodología que será empleada en este trabajo es la metodología exploratoria, este tipo de investigación conlleva al investigador a tener un primer contacto de aquello que en lo que se busca investigar y de lo cual no hay antecedentes expuestos (El pensante, 2017).

Debido a que la investigación exploratoria se basa en aproximaciones o cálculos, este trabajo se proyecta a utilizar dicha investigación, ya que para llevarla a cabo se implementara un software tecnológico, será dicho software el que muestre mediante graficas los cálculos que se requieran.

## CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA DE SEÑALES Y SISTEMAS

### 2.1. ¿Qué es una señal?

Una señal  $x(t)$  es una función ya sea de valor escalar o de valor real de una variable  $t$  (tiempo). La señal como función podrá tener una o varias variables. En Telecomunicaciones una señal contiene información, y esta podrá ser una señal continua o una señal discreta dependiendo de su variación en el tiempo (Kamen & Heck, 2008).

La señal posee características las cuales son nombradas brevemente a continuación en la tabla 2.1

Tabla 2. 1: Características de una señal

Característica	Valores	
Numero de variables	1	múltiples
Dimensionalidad	escalar	vectorial
Variables independientes	discretas	continuas
Valores de la señal	discretos	continuos
Naturaleza estadística	determinadas	aleatorias

Fuente: (Alvarado, 2008)

Elaborado por: Autor

En telecomunicaciones una señal que transporta información pasa a ser modulada haciendo así que las características de dicha señal varíen, la modulación sirve de gran ayuda como una técnica para el tratamiento de una onda portadora, que al ser modulada aprovechara de manera óptima el canal para la comunicación. Una señal puede ser análoga o digital, un claro ejemplo de señal análoga es la voz humana, por otra parte como ejemplo de señal digital se puede nombrar a un velocímetro digital.

### 2.1.1. Señal continua

Una señal  $x(t)$  es continua en el tiempo cuando su variable independiente tiempo tiene un valor continuo, por ello  $x(t)$  tendrá un valor determinado en todos los instantes de tiempo  $t$ . Normalmente cuando se refiere a la variable utilizada para representar una señal, se habla de la variable  $t$  (tiempo), pero no únicamente ocupará esta variable, podrá utilizar diferentes variables, dependiendo de la aplicación que se le da a la señal (Morón, 2011). Como resultado de muestrear una señal continua, se obtiene una señal analógica, a continuación en la figura 2.1 se ilustra el muestreo de la señal, con variables (euros) y (días) para el ejemplo.

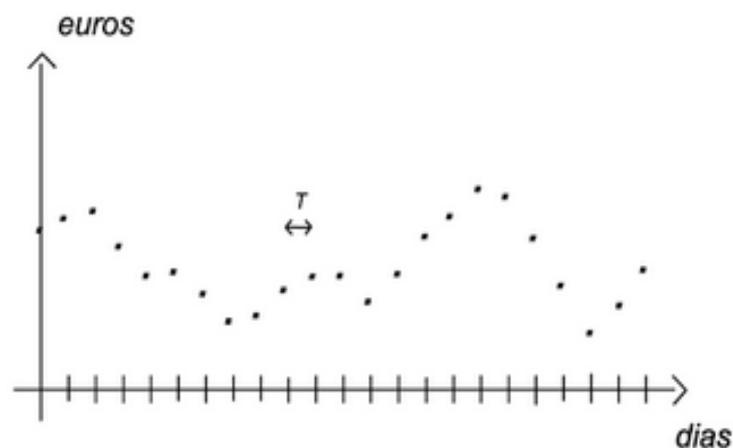


Figura 2.1: Muestreo de una señal analógica  
Fuente: (Albertí & Upc, 2009)

### 2.1.2. Señal discreta.

La señal discreta se define en tiempos discretos, esta señal solo tomará un conjunto de valores de la variable independiente, la función de una señal discreta encerrará en corchetes su variable, de esta forma la función queda expresada por:  $x[t]$ , donde a la variable tiempo se le asignara números

enteros reales (Oppenheim, Willsky, & Nawab, 2015). En la figura 2.2 se observa el muestreo de una señal discreta.

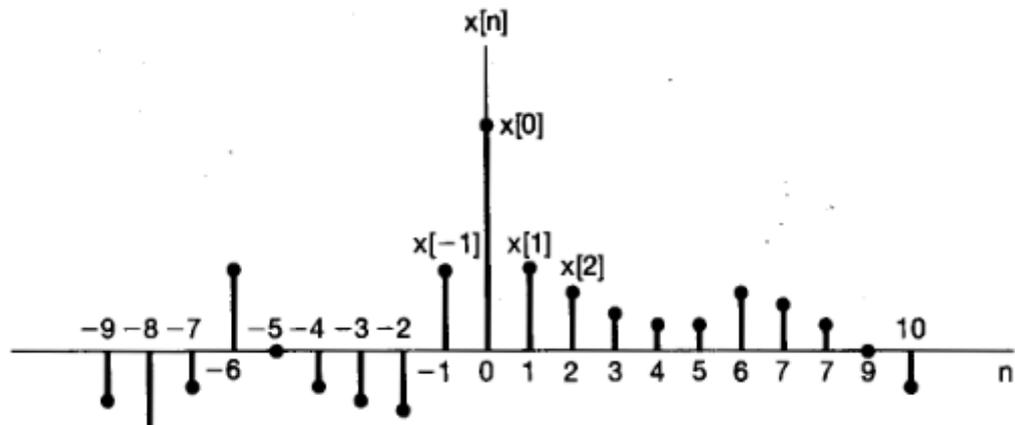


Figura 2.2: Muestreo de una señal discreta  
Fuente: (Oppenheim et al., 2015)

## 2.2. Señales Básicas de Tiempo Continuo

Las señales de tiempo continuo básicas, sirven como base para la construcción de otras señales, a continuación se citan brevemente algunas de estas señales.

### 2.2.1. Función Escalón Unitario

Una función escalón unitario es aquella donde la variable independiente  $t$  es discontinua con el valor 0, el valor para  $t = 0$  no está definido (Morón, 2011). En la figura 2.3 se muestra la gráfica de la función de escalón unitario, la cual está definida por siguiente expresión:

$$u(t) = \begin{cases} 1, & t > 0 \\ 0, & t < 0 \end{cases}$$

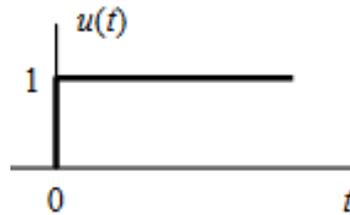


Figura 2.3: Función escalón unitario  
Fuente: (Morón José, 2011)

### 2.2.2. Función Impulso Unitario

La función impulso unitario también es llamada función delta de Dirac, esta función forma parte de las funciones generalizadas o también llamadas distribuciones, en la figura 2.4 se muestra la gráfica de la función impulso unitario, dicha función se define por  $\delta(t)$ , y tiene varias propiedades que son las siguientes:

1. La señal tendrá valor = 0 en todas las partes pero no en el origen:

$$\delta(t) = \begin{cases} 0, & t \neq 0 \\ \text{no esta definida en } & t = 0 \end{cases}$$

2. Esta función tiene una derivada la cual es la función delta, dando como resultado la siguiente función:

$$\delta(t) \frac{du(t)}{dt}$$

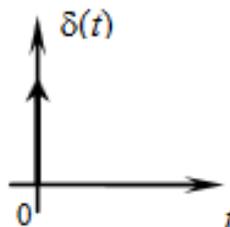


Figura 2.4: Función Impulso Unitario  
Fuente: (Briceño José E., 2012)

### 2.3. Señales Básicas de Tiempo Discreto

Dos de las señales básicas de tiempo discreto se citan a continuación, como son: la señal sinusoidal y la señal secuencia escalón unitario.

#### 2.3.1. Señal Sinusoidal

Una señal sinusoidal está dada por las siguientes funciones:

$$f(t) = F \cos(\omega t + \phi)$$

$$f(t) = F \sen(\omega t + \phi)$$

En la figura 2.5 se ha tomado para ejemplo la amplitud de una señal sinusoidal.

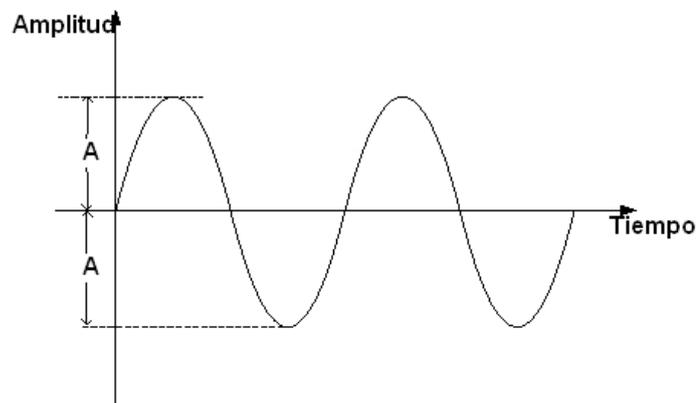


Figura 2.5: Amplitud de una señal sinusoidal  
Fuente: (Córdova Darwin, 2017)

#### 2.3.2. Secuencia Escalón Unitario

Para valores discretos se hace uso de corchetes, el valor de  $u[n]$  se definirá con  $n = 0$  (Morón, 2011). En la figura 2.6 se ilustra la secuencia escalón unitario, cuya función queda definida por:

$$u[n] = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

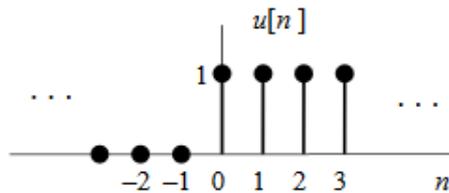


Figura 2.6: Secuencia Escalón Unitario  
Fuente: (Morón José, 2011)

## 2.4. Sistemas

Un sistema implica el manejo de más de un proceso o dispositivo, tendrá siempre una señal entrante y una señal saliente, su clasificación depende del tipo de señales que utilice en sus procesos, por consiguiente existen dos sistemas: sistema continuo o también llamado analógico y el sistema discreto.

### 2.4.1. Sistema Continuo

Un sistema continuo es aquel cuya señal entrante es continua en el tiempo  $x(t)$  y a su vez la señal saliente también se presenta de forma continua  $y(t)$  (Kamen & Heck, 2008). El sistema continuo presenta varias ventajas sobre el sistema discreto, este sistema permite que el procesamiento de la señal sea más fácil, por ser un sistema análogo es menos común que la señal se vea afectada por ruidos en el ambiente. Otra notable ventaja del sistema es que la señal resultante puede ser ampliada y restaurada a la vez, en la figura 2.7 se ha tomado como ejemplo un circuito RC para ilustrar un sistema continuo análogo, en donde la señal de entrada será la corriente que pasa por las líneas conectadas en paralelo  $i(t)$  y la señal saliente será el voltaje  $V_c(t)$  obtenido en las líneas del capacitor. Ambas señales tanto de entrada como

de salida se presentan de forma continua tanto en su magnitud como en la variable de tiempo.

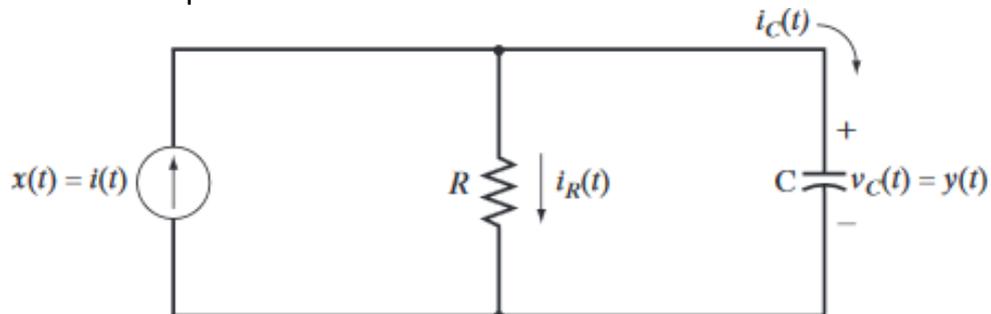


Figura 2.7: Circuito RC.

Fuente: (Kamen Edward W. & Heck Bonnie S, 2008)

#### 2.4.2. Sistema Discreto

Un sistema discreto es aquel en que su señal de entrada presenta valores discretos de tiempo, es decir muestreos en instantes y la señal de salida resultante también se produce en instantes discretos de tiempo. La característica de este sistema es que las señales son discontinuas en magnitud y tiempo (Albertí, 2009). Este sistema requiere previamente una convergencia análoga-digital y un ancho de banda superior al sistema análogo para el tratamiento de las señales, los sistemas discretos o digitales tienen varias áreas de aplicación entre las cuales se puede mencionar: el procesamiento de la voz, procesamiento del video, así como los distintos equipos que se utilizan en áreas de medicina.

Una ventaja de este sistema es su fiabilidad, es de gran utilidad ya que un bloque de sistema digital puede ser utilizado en varias áreas como la comunicación, televisión, entre otras, sin importar el origen de las señales tanto de entrada como de salida, el sistema funcionara correctamente. En la

figura 2.8 se observa un breve ejemplo de aplicación de sistemas discretos utilizados tanto en resoluciones como en un sistema de radar.

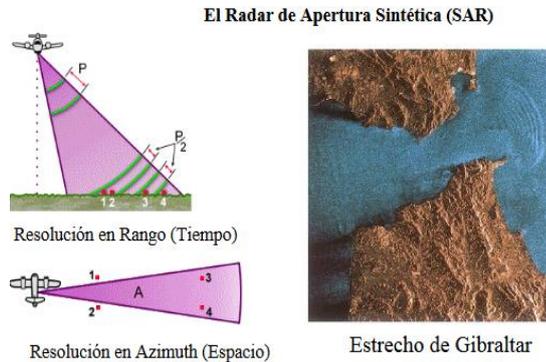


Figura 2.8: Aplicaciones de sistema discreto.

Fuente: (Morales Luis, 2017)

### 2.4.3. Muestreo de señal análoga

Es necesario digitalizar una señal analógica para poder realizar la manipulación con un sistema digital, para esto, se requiere previamente un convertidor análogo – digital; la digitalización se la realiza tanto en amplitud como en espacio, en la figura 2.9 se observa el muestreo de una señal análoga.

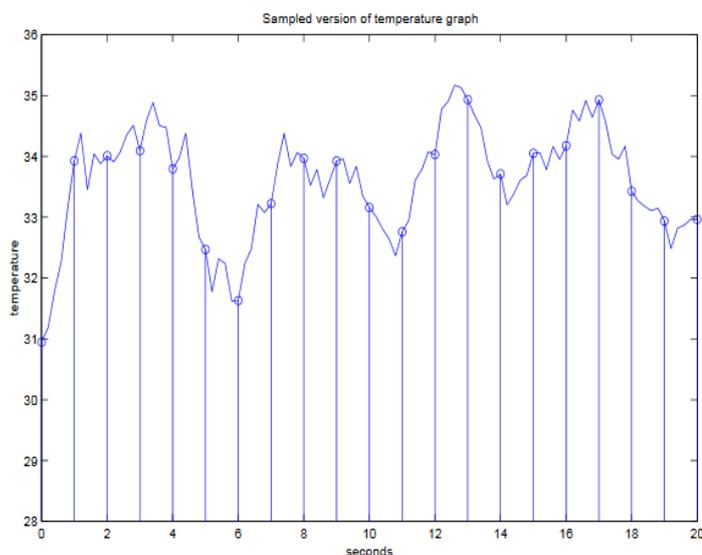


Figura 2.9: Muestreo de señal análoga.

Fuente: (Kulkarni S.R., 2012)

## 2.5. Funciones trigonométricas

Antes de abarcar el tema series de Fourier es necesario nombrar algunas propiedades que son esenciales en las funciones trigonométricas, a continuación se enlista algunas de las propiedades básicas.

### 2.5.1. Funciones trigonométricas básicas

El valor del período para las funciones *sen* y *cos* es de  $2\pi$  de esta manera satisfacen las ecuaciones:

$$\text{sen}(x + 2\pi) = \text{sen}(x)$$

$$\text{cos}(x + 2\pi) = \text{cos}(x)$$

Dónde  $x$  representa una cifra real (Bruzual & Domínguez, 2003). Las siguientes igualdades son básicas y son las más utilizadas:

$$\text{sen}(x \pm y) = \text{sen}x \text{cos}y \pm \text{sen}y \text{cos}x$$

$$\text{cos}(x \pm y) = \text{cos}x \text{cos}y \mp \text{sen}x \text{sen}y$$

$$\text{sen}^2x = \frac{1 - \text{cos}(2x)}{2}$$

$$\text{cos}^2x = \frac{1 + \text{cos}(2x)}{2}$$

$$\text{sen}x \text{cos}y = \frac{1}{2} (\text{sen}(x - y) + \text{sen}(x + y))$$

$$\text{sen}x \text{sen}y = \frac{1}{2} (\text{cos}(x - y) - \text{cos}(x + y))$$

$$\text{cos}x \text{cos}y = \frac{1}{2} (\text{cos}(x - y) + \text{cos}(x + y))$$

## 2.6. Introducción a series de Fourier

La serie de Fourier se define como la forma de representar una función (onda) como la suma de varias ondas sinusoidales que serán lo más simples posible. Más específicamente las series de Fourier descomponen cualquier función periódica o señal periódica en la suma de un conjunto posiblemente infinito de funciones oscilantes simples, ya sean senos o cosenos.

Las series de Fourier también son centrales en la prueba original del teorema de muestreo de Nyquist-Shannon, estas series permiten expresar en una suma de senos y cosenos una función que sea de período  $T$ ; las series de Fourier son de gran utilidad en la resolución de ecuaciones diferenciales, pueden resultar complejas a simple vista pero la aplicación se realiza de manera simple.

También existen las series de Taylor, las cuales minimizan una serie de potencias mediante una función que se aproximará al valor de estas series utilizando el número de términos que se utilizan para determinar cuan aproximado es el valor que obtiene la serie y cuánto será el fallo que se comete en el desarrollo de la misma (Hernández, 2017).

Las series de Taylor son una herramienta bastante utilizada, sin embargo no son tan generales como las series de Fourier; las funciones periódicas que sean discontinuas pueden ser desarrolladas mediante una serie de Fourier, pero estas funciones no se las puede representar mediante una serie de

Taylor (Alcaraz, 2017). Antes de establecer las series de Fourier es necesario determinar los coeficientes y esto se realiza mediante las fórmulas de Euler, a continuación se menciona cuáles son estos coeficientes.

### 2.6.1. Coeficientes de Fourier

Las fórmulas de Euler para el cálculo de los coeficientes de Fourier son los nombrados a continuación:

- Para el cálculo del coeficiente  $a_n$  la ecuación está dada por:

$$a_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(t) \cos(kx) dx \text{ para todo } k \geq 0$$

- Para el cálculo del coeficiente  $b_n$  la ecuación está dado por:

$$b_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(t) \sin(kx) dx \text{ para todo } k > 0 \text{ (Alcaraz, 2017).}$$

### 2.6.2. Series de Fourier

La sumatoria de una función  $f$  es también llamada serie de Fourier, en la figura 2.10 se observa varias graficas resultantes de las sumas parciales de Fourier; la serie de Fourier está definida por la siguiente suma:

$$\frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos kx + b_k \sin kx$$

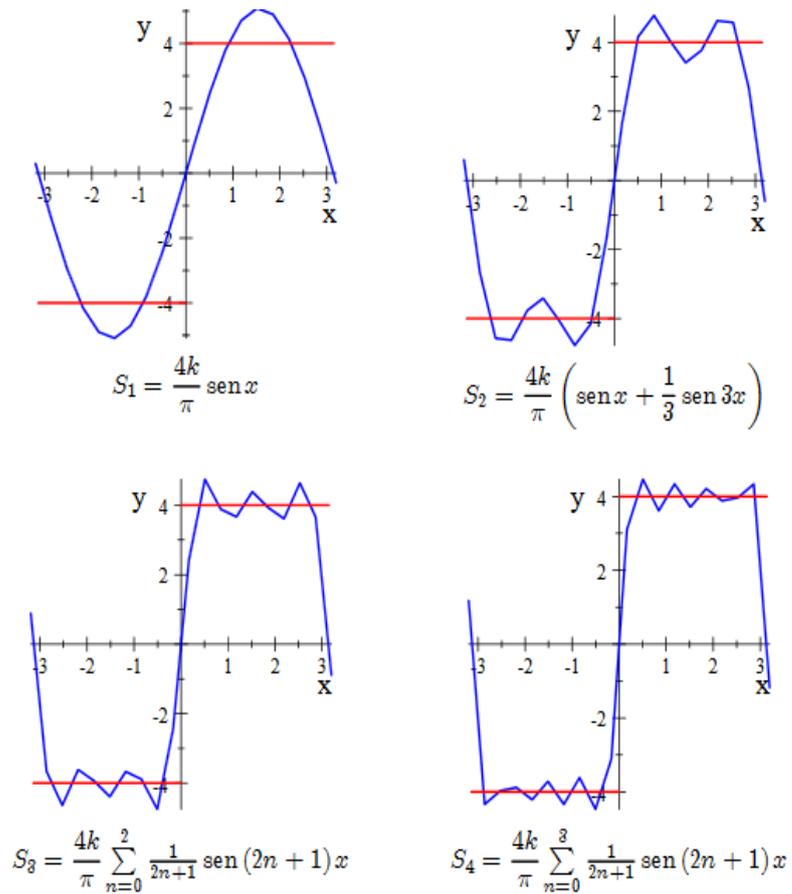


Figura 2.10: Representación gráfica de series de Fourier  
Fuente: (Alcaraz Domingo, 2017)

## 2.7. Transformada de Fourier

La transformada de Fourier es útil para expresar en términos de frecuencia una señal no periódica, las componentes de frecuencia están determinadas tanto para valores discretos como valores reales de  $\omega$  (frecuencia). En el dominio del tiempo una señal puede tener información que no es visible, con la transformada de Fourier se obtiene dicha información de la señal. En telecomunicaciones se da el uso de la transformada de Fourier al digitalizar la voz humana y en el diseño de filtros entre algunos de los usos que se puede mencionar.

### 2.7.1. Transformada de Fourier en tiempo continuo

En la transformada de Fourier en tiempo continuo una señal no periódica que resulta tener energía limitada tendrá un espectro continuo si se lleva su representación en términos de frecuencia (Cáceres, 2007). La expresión de la transformada de Fourier en tiempo continuo se define por la siguiente ecuación:

$$X(f) = \int_{-\infty}^{\infty} x(t) \cdot e^{-j \cdot 2\pi \cdot f \cdot t} dt$$

### 2.7.2. Transformada de Fourier en tiempo discreto

Según Cáceres (2017) “La transformada de Fourier de tiempo discreto utiliza frecuencia y tiempo en valores discretos; utilizando un rango de  $0 \leq k \leq N - 1$  su ecuación característica es”:

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{-\frac{2\pi j}{N} k \cdot n}$$

### 2.7.3. Transformada inversa de Fourier en tiempo discreto

También suele abreviarse como IDFT, esta transformada mediante sus muestras  $X[k]$  (muestras de espectro) puede generar una nueva señal  $x[n]$ , en donde  $x[n]$  deberá tomar valores en un rango de  $0 \leq n \leq N - 1$  la expresión que lo hace posible es:

$$x[n] = \frac{1}{N} \cdot \sum_{k=0}^{N-1} X[k] \cdot e^{j \cdot \frac{2\pi}{N} k \cdot n}$$

De este modo, sí se utiliza la ecuación de la transformada discreta de Fourier para valores fuera del rango especificado el resultado será una secuencia periódica. Del mismo modo si en la transformada discreta inversa de Fourier se utilizaran valores fuera del rango definido, el resultado sería una señal periódica de  $x[n]$ . Por lo tanto la transformada de Fourier como la IDFT en una señal no periódica dan como resultado una secuencia periódica (EHU, 2017).

Una ventaja de la transformada de discreta de Fourier es considerar una señal no periódica como una señal periódica como resultado de utilizar algoritmos. En la figura 2.11 se muestra la relación entre la transformada discreta de Fourier y la transformada inversa de la misma mediante muestreo de una señal  $x[n]$ .

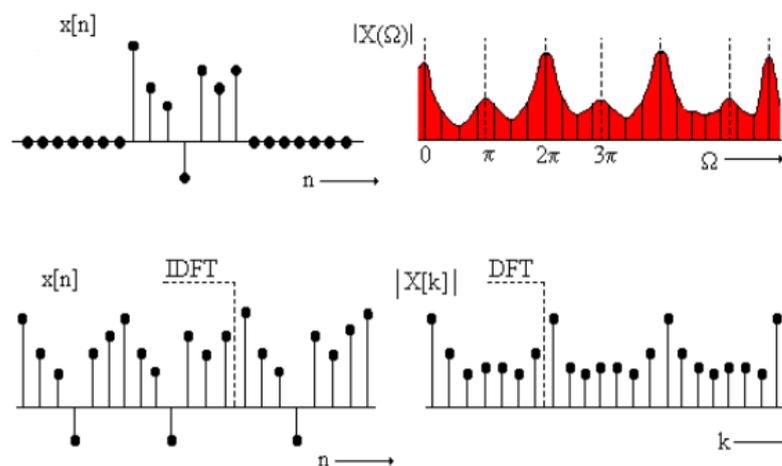


Figura 2.11: Muestreo de una señal  $x[n]$  y relación entre DFT e IDFT.  
Fuente: (EHU, 2017)

Como resultado del análisis de la transformada discreta de Fourier se concluye, que es necesario el empleo de funciones discretas, puesto que un

sistema digital no puede procesar funciones continuas, es necesario expresar dichas funciones en un muestreo de valores discretos que permitan la aproximación a la función continua.

## **2.8. Transformada de Laplace**

La transformada de Laplace es también una herramienta que reduce la complejidad de los sistemas lineales que presentan variación en el tiempo llevando el análisis al dominio de la frecuencia, la transformada de Laplace permite manejar ecuaciones diferenciales como ecuaciones algebraicas, haciendo así que su análisis sea menos complejo (Cornejo, Villalobos, & Quintana, 2008). Laplace lleva al dominio  $s$  las derivadas e integrales que se encuentren en el dominio de tiempo a sencillos polinomios, cuando ya se tiene la expresión simplificada en el dominio de la frecuencia, mediante la inversa de Laplace se regresa al dominio del tiempo, de esta manera no se pierde el problema original; Laplace es utilizada en sistemas de control, sistemas de potencia, tecnología del espacio, sistema mecánico, circuitos eléctricos, entre otros.

### **2.8.1. Definición de la transformada de Laplace**

Laplace transforma la función  $f(t)$  en una función  $F(s)$  en el dominio  $s$ , donde  $s$  pertenece a cualquier número real (James & Burley, 2002). La transformada de la Laplace de una función se define con la expresión:

$$F(s) = \mathcal{L}\{f(t)\} = \int_0^{\infty} e^{-st} f(t) dt = \lim_{T \rightarrow \infty} \int_0^T e^{-st} f(t) dt$$

### 2.8.2. Transformada de Laplace de funciones básicas

En la tabla 2.2 se expresan las funciones básicas más utilizadas para el cálculo de Laplace (James & Burley, 2002).

Tabla 2. 2: Transformada de Laplace de funciones básicas.

$f(t)$	$F(s)$
$c$	$\frac{c}{s}$
$\delta$	1
$t$	$\frac{1}{s^2}$
$e^{kt}$	$\frac{1}{s - k}$
$\sin at$	$\frac{a}{s^2 + a^2}$
$\cos at$	$\frac{s}{s^2 + a^2}$
$e^{-\alpha t} \sin \omega t$	$\frac{\omega}{(s + \alpha)^2 + \omega^2}$
$e^{-\alpha t} \cos \omega t$	$\frac{s + \alpha}{(s + \alpha)^2 + \omega^2}$

Fuente: (James & Burley, 2002)

Elaborado por: Autor

### 2.8.3. Transformada inversa de Laplace

Se puede obtener una función  $f(t)$  si se conoce  $F(s)$  mediante la siguiente ecuación:

$$x(t) = \frac{1}{2\pi j} \int_{c-j\infty}^{c+j\infty} X(s) \cdot e^{st} ds$$

### 2.9. Transformada Z

La transformada z realiza la misma función que la transformada de Laplace pero utilizando ecuaciones diferenciales que se encuentre en tiempo discreto, estas ecuaciones serán transformadas a un dominio de frecuencia (Moreno, Garrido, & Balaguer, 2003). Por tanto, así como Laplace utiliza

señales analógicas, la transformada z utilizara señales de tiempo discreto, es mayormente utilizada en sistemas causales y lineales, buscando simplificar las ecuaciones diferenciales en ecuaciones algebraicas simples que faciliten el tratamiento de las mismas.

### 2.9.1. Definición de la transformada z

De la misma manera, teniendo una señal en tiempo discreto o una ecuación diferencial, la transformada z que se obtiene está dada por la siguiente ecuación:

$$X(z) = Z[x(k)] = \sum_{k=0}^{\infty} x(k)z^{-k}$$

En donde el valor de  $x(t)$  será 0 cuando para todo  $k < 0$  (Moreno et al., 2003).

### 2.9.2. Transformada z de funciones básicas

En la tabla 2.3 se expresan las funciones básicas utilizadas para la aplicación de la transformada z.

Tabla 2. 3: Transformada z de funciones básicas.

$f(k)$	$F(z)$
$\delta(k)$	1
$\delta(k - n)$	$\frac{z^{-n}}{z}$
$u_0(k)$	$\frac{z - 1}{z}$
$a^k$	$\frac{z - a}{z}$
$k$	$\frac{z}{(z - 1)^2}$
$k + 1$	$\frac{z^2}{(z - 1)^2}$
$\sin(\alpha k)$	$\frac{(\sin \alpha)z}{z^2 - (2 \cos \alpha)z + 1}$
$\cos(\alpha k)$	$\frac{z^2 - (\cos \alpha)z}{z^2 - (2 \cos \alpha)z + 1}$

Fuente: (Moreno, 2003)

Elaborado por: Autor

## 2.10. Filtros Analógicos

Un filtro es un sistema por el cual entra una señal que luego de ser procesada tendrá como resultado una señal saliente, un filtro analógico se caracteriza por procesar señales entrantes analógicas, todas las señales que se tomen como muestra serán analógicas, de allí recibe su nombre. También utilizan términos de ecuaciones diferenciales, estos filtros se diseñan basándose en la convergencia de magnitudes o fases por funciones racionales.

### 2.10.1. Tipos de filtros analógicos

Existen cinco tipos de filtros analógicos y son:

- i) Filtro pasa-baja (LP)
- ii) Filtro pasa-alta (HP)
- iii) Filtro pasa-banda (BP)
- iv) Filtro rechazo de banda (BR)

En la figura 2.12 se muestran los tipos de filtros análogos.

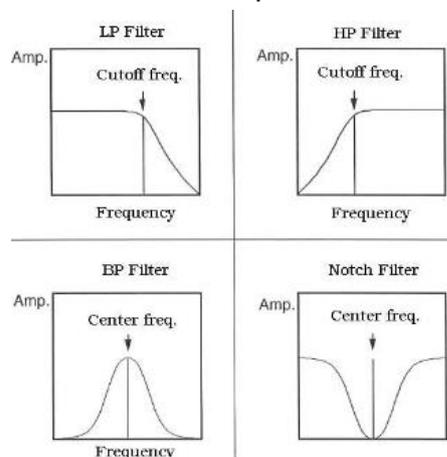


Figura 2.12: Tipos de filtros análogos.  
Fuente: (Gómez Emilia, 2009)

### **i. Filtro pasa-baja**

También llamado filtro (LP) por sus siglas en inglés, la característica de este filtro es que si la señal tiene una frecuencia mayor a la que será la frecuencia de corte, el filtro atenuará dicha frecuencia que está sobrepasando su valor y solo mantendrá las demás. La ecuación característica del filtro LP es la siguiente:

$$H(s) = K_{LP} \cdot \frac{\omega_0}{s + \omega_0}$$

### **ii. Filtro pasa-alta**

Filtro también llamado (HP), este filtro se caracteriza por atenuar las frecuencias de la señal que sean menores a la frecuencia de corte y conserva aquellas frecuencias que sean mayores a la frecuencia de corte. La ecuación que define al filtro pasa-alta es:

$$H(s) = K_{HP} \cdot \frac{s}{s + \omega_0}$$

### **iii. Filtro pasa-banda**

También llamado filtro (BP), este filtro elimina factores cuya frecuencia no esté entre dos valores de la frecuencia de corte, solo conserva aquellos factores en donde su frecuencia esté entre dichos valores de la frecuencia de corte, El filtro BP se define por la siguiente ecuación:

$$H(s) = K_{BP} \cdot \frac{\omega_0}{Q_F} \cdot \frac{s}{s^2 + \frac{\omega_0}{Q_F} \cdot s + \omega_0^2}$$

#### iv. Filtro rechazo de banda

También llamado filtro (BR), este filtro elimina las frecuencias que se encuentren entre dos valores de la frecuencia de corte y solo permite las frecuencias que no se encuentren comprendidas entre los dos valores de la frecuencia de corte. Su ecuación característica es:

$$H(s) = K_{BR} \cdot \frac{s^2 + \omega_0^2}{s^2 + \frac{\omega_0}{Q_F} \cdot s + \omega_0^2}$$

### 2.11. Filtros digitales

Básicamente un filtro digital es aquel que representa un proceso en donde la señal entrante es procesada y transformada a un muestreo en serie, o a su vez a una señal digital, si la señal entrante es analógica se requiere de un conversor que transforme la señal analógica a su equivalente digital, este conversor es también llamado (ADC), si se requiere volver a la forma original de la señal se requiere de un elemento llamado (DAC), que es un conversor digital - analógico (Gómez, 2009). Estos filtros utilizan términos de ecuaciones de diferencias y tienen su uso en el área de procesado de señal, señales como audio, video, imágenes, etc.)

El filtro digital permite el paso de una señal a otra, no es más que un algoritmo que se utiliza tanto para obtener información específica de la señal en análisis, como para independizar la misma. Es posible alterar los filtros, rediseñándolos con la agregación de resistencias, capacitores, bobinas entre otras piezas posibles de ser incluidas (Pinoargote, 2016). Por tanto un filtro

digital es un sistema discreto que permite la alteración de la señal entrante, esta alteración se ve reflejada en su frecuencia. En la figura 2.13 se observa es esquema de un filtro digital.

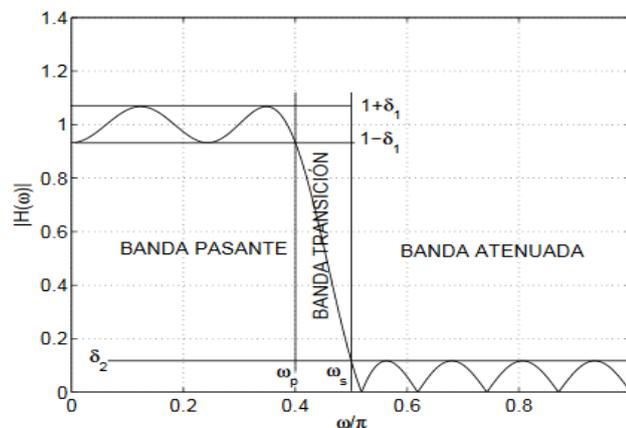


Figura 2.13: Esquema de diseño de un filtro digital.  
Fuente: (Soria Olivas, 2003)

Una de las ventajas que poseen los filtros digitales a diferencia de los filtros analógicos es su mejorada aplicación ya que en la actualidad existen varios softwares que hacen posible su programación, produciendo así un rápido muestreo y procesamiento de la señal, en cambio, un filtro analógico posee en la entrada y en la salida una señal analógica, lo que implica que al muestrearla se obtenga una infinita cantidad de valores. En sistemas de comunicaciones es común utilizar filtros ya sean analógicos o digitales, debido a que es necesario disminuir la presencia de ruidos o distorsiones que son comunes en este tipo de sistemas. Uno de los filtros más utilizados por su gran eficiencia en la eliminación de distorsión de sonido es el filtro LTC1562-2. En la figura 2.14 se ilustra el diagrama de un filtro de 150khz, que es bastante utilizado en radiocomunicación.

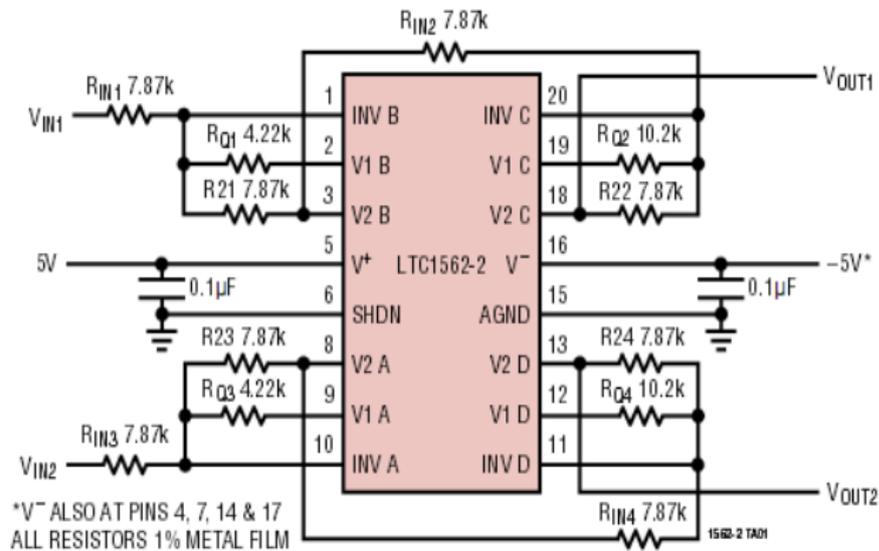


Figura 2.14: Diagrama de un filtro digital de 150khz.  
Fuente: (Pinoargote Edwin, 2016)

### 2.11.1. Tipos de filtros digitales

Dentro de la categoría de filtros digitales con respuesta al impulso se encuentran los siguientes:

- i) Filtros FIR
- ii) Filtros IIR

#### i) Filtros FIR

También llamado filtro con respuesta impulsional infinita, en este filtro la señal de salida es resultante de la suma de las muestras de entrada y muestras pasadas que ingresan al filtro. La figura 2.15 ilustra la entrada y salida de las muestras en un filtro FIR, cuya ecuación característica es:

$$y[n] = a_0 \cdot x[n] + a_1 \cdot x[n - 1] + a_2 \cdot x[n - 2] + \dots + a_N \cdot x[n - N]$$

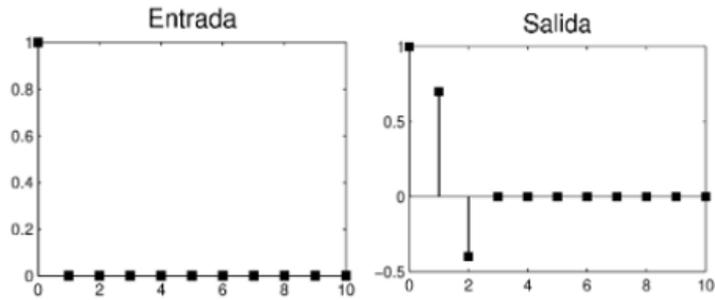


Figura 2.15: Muestras de una señal entrante y saliente en un filtro FIR.  
Fuente: (EME, 2017)

## ii) Filtros IIR

Este filtro tiene una respuesta infinita al impulso, una vez que la señal de saliente ha sido procesada por el filtro, ésta se introduce nuevamente en la señal entrante, creando así un circuito retroalimentado y capaz de añadir filtros con respuesta más compleja sin la necesidad de agregar una mayor cantidad de datos (Gómez, 2009). En la figura 2.16 se muestra la señal de entrada y salida en un filtro IIR; la ecuación del filtro IIR está definida de la siguiente forma:

$$y[n] = a_0 \cdot x[n] + a_1 \cdot x[n - 1] + a_2 \cdot x[n - 2] + \dots + a_N \cdot x[n - N] + \\ -b_1 \cdot y[n - 1] - b_2 \cdot y[n - 2] - b_3 \cdot y[n - 3] - \dots - b_M \cdot y[n - M]$$

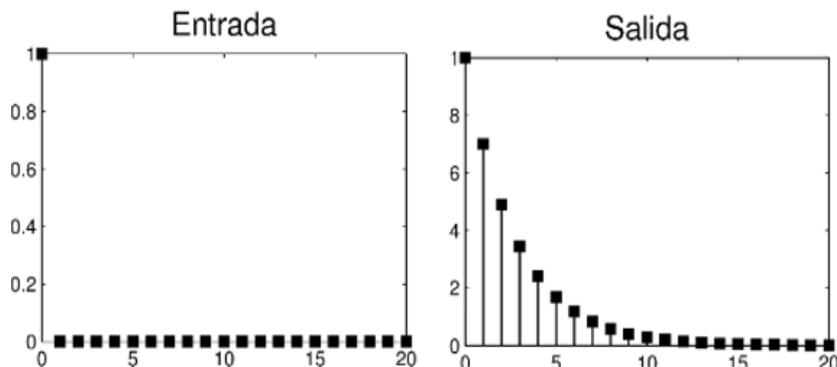


Figura 2.16: Muestras de una señal entrante y saliente en un filtro IIR.  
Fuente: (EME, 2017)

### CAPÍTULO 3: INTRODUCCIÓN A LABVIEW

**LabView** (*Laboratory Virtual Instrument Engineering Workbench*) es un lenguaje y un sistema innovador de programación gráfica que facilita el diseño de aplicaciones. El usuario de este sistema operará en un entorno de programación diferente al de otros sistemas de programación. En *LabView* no se emplean líneas de instrucciones basadas en texto, sino que se codifican las ideas de programación seleccionando y modelando convenientemente los íconos disponibles. Un sistema *LabView* consta de dos ventanas: el panel frontal y el diagrama de bloques. Una vez finalizado un programa, el usuario verá el panel frontal con su propio diseño (Essick, 2016).

*LabView* es un entorno de desarrollo de software creado por *National Instruments*. A principios de su desarrollo, *LabView* se centró en permitir al usuario desarrollar instrumentos virtuales para adquirir y mostrar datos sin la ayuda de un instrumento hardware equivalente, como un osciloscopio o un multímetro. El nombre del software, *LabView*, y el nombre dado a los programas desarrollados por *LabView*, llamados Instrumentos Virtuales (VI) con una extensión de archivo ".vi", son el resultado de este enfoque inicial de los programas de software *LabView*.

*LabView* se ha utilizado tradicionalmente para la medición y control de equipos y procesos, que siguen siendo importantes áreas de aplicación automatizada. El software *LabView* permite el desarrollo rápido de VIs para

medir, procesar, analizar, visualizar y almacenar datos, así como generar datos (voltajes o corrientes) para controlar instrumentos y hardware, como motores, válvulas o interruptores. La constante evolución del software *LabView* por parte de *National Instruments* le ha permitido convertirse en una herramienta de programación general, permitiendo el desarrollo de algoritmos, análisis matemáticos y tareas de comunicación que van más allá de la adquisición de datos.

El *LabView* es un software basado en un lenguaje de programación gráfico, conocido como código G, y el concepto de flujo de datos para controlar la ejecución del programa. El lenguaje de programación G está representado por iconos de función conectados por cables virtuales, que permiten que los datos fluyan entre los iconos de función. Las funciones pueden tener cualquier número de terminales de entrada y salida. Una función no se ejecuta hasta que los datos llegan a todos los terminales de entrada. La ejecución del programa, por lo tanto, es controlada por el flujo de los datos. Esto es conceptualmente diferente de los lenguajes de programación basados en texto donde la ejecución del programa es determinada principalmente por el orden de las declaraciones del programa (Stamps, 2015).

### **3.1. Entorno**

El entorno de programación gráfica *LabView* se estructura básicamente de dos secciones, el Panel Frontal y el Diagrama de Bloques, ambos se conectan a través de terminales.

### 3.1.1. Panel frontal

El panel frontal es la interfaz de usuario del programa, esto es, la interacción entre las órdenes del usuario y la observación de los resultados de los programas a medida que se ejecuta (Essick, 2013). Éste aparece cuando se abre un programa *LabView*, nuevo o existente, como se muestra en la figura 3.1.

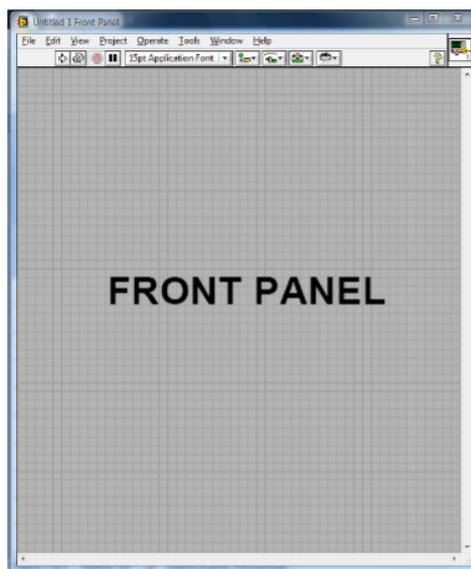


Figura 3. 1: Panel frontal.  
Elaborado por: Autor

### Paleta de Controles

Contiene los controles e indicadores para crear el panel frontal. Se divide en varias categorías y se puede exponer algunas o todas según los requerimientos del usuario. Se tiene acceso a ésta seleccionando la opción **View » Controls Palette** o sencillamente presionando el click derecho en cualesquier espacio de la ventana del panel frontal. Para divisar u ocultar las categorías se selecciona o deshabilita la opción **View » Always Visible Categories**. (Véase la figura 3.2)

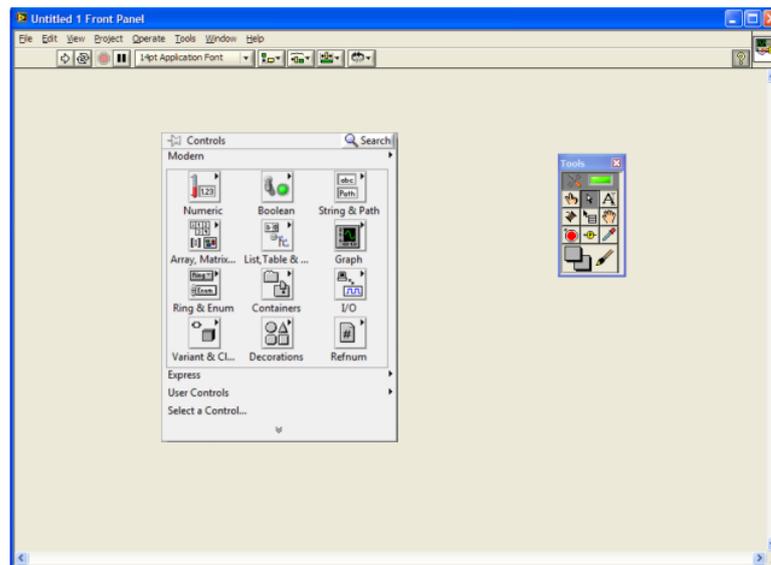


Figura 3. 2: Paleta de controles  
Elaborado por: Autor

## Controles e indicadores

El menú de controles se muestra en el Panel Frontal y aquí se seleccionan los terminales con los que el usuario interactuará. Suele clasificarse los terminales en controles e indicadores. Los controles son terminales interactivos de entradas de datos, con los que se crea el panel frontal. Los controles son perillas, botones, barras deslizantes, entre otros, representan dispositivos de entrada de instrumentos y proporcionan datos al diagrama de bloques del VI.

Los indicadores son terminales interactivos de salida de datos, pueden ser gráficas, LEDs y otras pantallas, simulan dispositivos de salida y exhiben los datos que el diagrama de bloques genera. El usuario puede modificar el valor de entrada para los controles y luego verá el valor generado en el indicador. Los valores generados para los indicadores se basan en el código

creado en el diagrama de bloques. Cada uno de los controles e indicadores pertenece a un tipo de datos, los más empleados son cadena de caracteres, booleanos y numéricos.

**Controles e indicadores de cadena de caracteres.** Son una secuencia de caracteres ASCII. Los controles en cadena se utilizan para recibir un texto del usuario como un nombre de usuario o una contraseña, los indicadores en cadena muestran texto al usuario. Estos controles e indicadores se muestran en la figura 3.3.

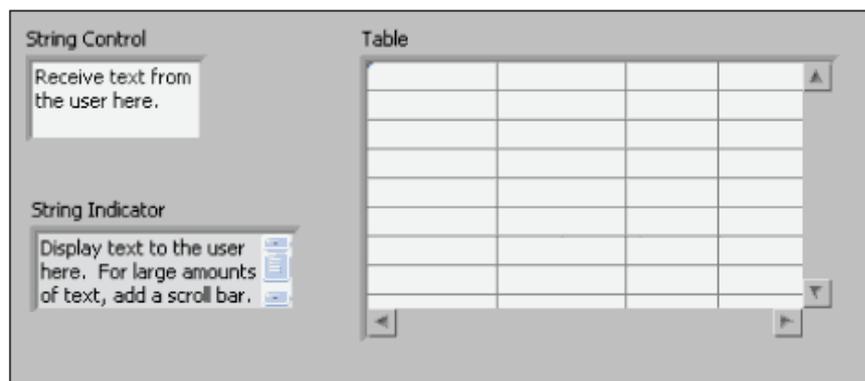


Figura 3. 3: Controles e Indicadores de cadena de caracteres  
Elaborado por: Autor

**Controles e Indicadores booleanos.** Este tipo de datos solo admite dos estados, como ON y OFF o TRUE y FALSE. Se utiliza este tipo de controles e indicadores para suministrar y representar valores booleanos. Simbolizan interruptores, botones y LEDs. Estos controles se ilustran en la figura 3.4.

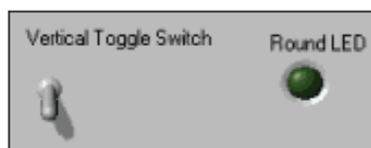


Figura 3. 4: Controles e Indicadores booleano  
Elaborado por: Autor

**Controles e Indicadores numéricos.** Este tipo de datos pueden representar números de diversas clases, como un entero o un real. Los objetos como medidores y perillas pueden también simular datos numéricos. Son mostrados en la figura 3.5.

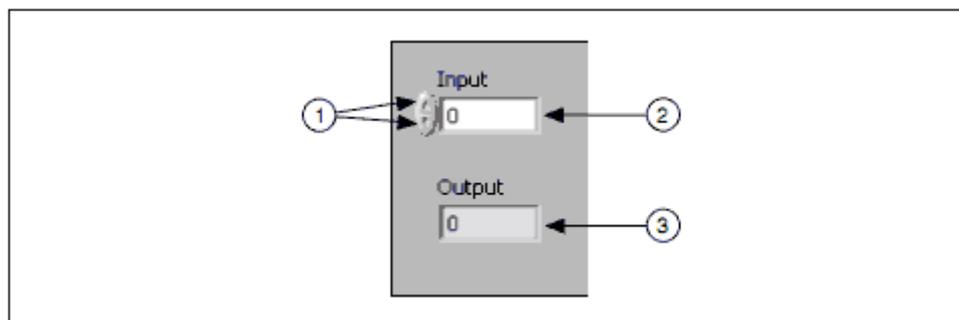


Figura 3. 5: 1. Incremento/Reducción de control numérico. 2. Control. 3. Indicador  
Elaborado por: Autor

### 3.1.2. Diagrama de bloques

El diagrama de bloques contiene el código fuente gráfico de LabView. Aquí residen las imágenes gráficas que se seleccionan apropiadamente de las surtidas bibliotecas de iconos de LabView (ver figura 3.6). Cada icono representa un bloque de código ejecutable subyacente que realiza una función útil en particular. Su tarea de programación es hacer las conexiones adecuadas entre estos iconos utilizando un proceso llamado cableado, de modo que el flujo de datos entre las imágenes gráficas logren el propósito deseado. Las librerías de iconos están diseñadas específicamente de acuerdo a las necesidades de los científicos y los ingenieros. *LabView* permite escribir programas que realizan todas las tareas de laboratorio necesarias para cualquier proyecto, incluyendo el control de instrumentos, la adquisición de datos, la presentación de datos y el almacenamiento de datos. (Essick, 2013)

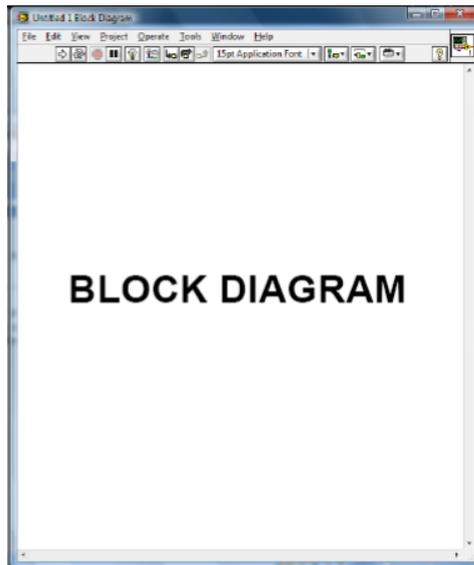


Figura 3. 6: Diagrama de bloques  
Elaborado por: Autor

Los nodos son objetos de diagramas de bloques que tienen entradas y salidas y efectúan operaciones cuando el programa se ejecuta. Sus equivalentes en lenguajes de programación basados en texto incluyen terminales, subVIs, funciones, constantes, estructuras y cables, los cuales transfieren datos junto con otros objetos del diagrama de bloques. (Ver figura 3.7)

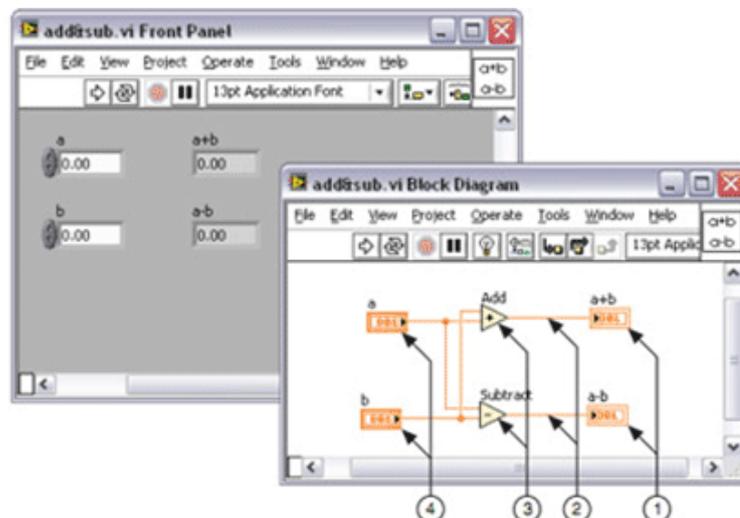


Figura 3. 7: Terminales de indicador, cables, nodos, terminales de control.  
Elaborado por: Autor

## Terminales de Diagrama de Bloques

Los objetos del panel frontal aparecen como terminales en el diagrama de bloques. Las terminales equivalen a parámetros y constantes en lenguajes de programación basados en texto y simulan puertos de entrada y salida que intercambian información entre el panel frontal y el diagrama de bloques. Entre los tipos de terminales se encuentran los de control o indicador y los de nodo. Los terminales de control e indicador corresponden a sus análogos del panel frontal. Es así que los datos suministrados a los controles del panel frontal se transmiten al diagrama de bloques por medio de los terminales de control. Cuando las funciones finalizan los cálculos proyectan nuevos valores de datos que van a los terminales de indicador actualizando los indicadores del panel frontal.

## Controles, indicadores y constantes

Los controles, indicadores y constantes actúan como entradas y salidas del algoritmo del diagrama de bloques, se muestran en la figura 3.8.

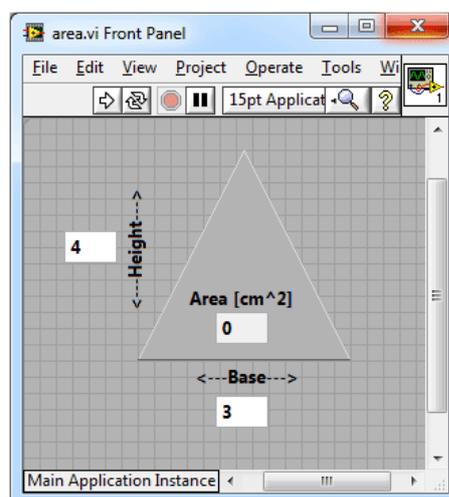


Figura 3. 8: Panel Frontal del Área de un triángulo  
Elaborado por: Autor

Las constantes pueden ser definidas por el usuario, por lo que no se tendrá acceso a ellas ni se mostrarán en el panel frontal a menos que se incluya como documentación del algoritmo; o predefinidas por el programa encontrándose disponibles en la paleta de funciones. Seguido se muestra una sencilla representación de lo que podría ser la implementación del algoritmo para la figura anterior en el diagrama de bloques de LabView, el cual presenta cuatro terminales generadas por dos controles, una constante y un indicador. (Ver figura 3.9)

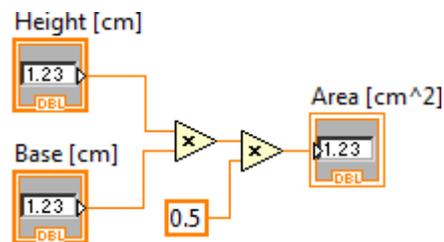


Figura 3. 9: Control, Indicador y Constante  
Elaborado por: Autor

Una diferencia entre un control y un indicador en el diagrama de bloques es que el control es una flecha en el terminal que indica la dirección del flujo de datos, mientras que el indicador tiene una flecha que muestra los datos que ingresan al terminal. La segunda diferencia es que el control presenta un borde grueso y el indicador de un borde fino.

## Estructuras

Son elementos de control de procesos, tales como Estructuras de Casos, Ciclos For o Ciclos While. Se encuentran disponibles en la paleta de funciones del diagrama de bloques. (Ver figura 3.10)

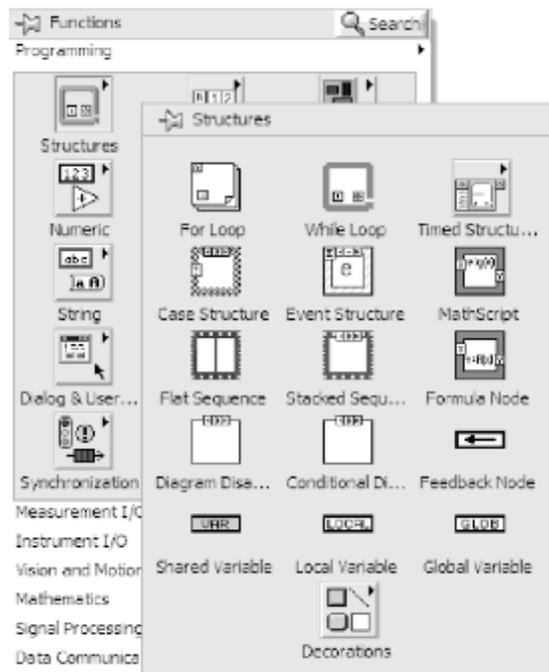


Figura 3. 10: Detalle de las estructuras  
Elaborado por: Autor

## Funciones

Son elementos de operación fundamentales, no poseen ventanas del panel frontal ni del diagrama de bloques y no tienen paneles conectores. SubVIs. No son más que Vis que el usuario crea dentro de otro VI, y es análogo a una función en lenguajes de programación basados en texto. Para usar un VI como un subVI se debe construir un panel conector, que no es más que un conjunto de terminales en el ícono que pertenece a los controles e indicadores de ese VI, equivalente a la lista de parámetros de una función en lenguaje de programación basado en texto. (Ver figura 3.11)

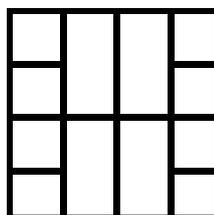


Figura 3. 11: Panel conector  
Elaborado por: Autor

Se puede tener acceso al panel conector desde la ventana del panel frontal, haciendo click derecho sobre el ícono, o desde la ventana del diagrama de bloques en el ícono correspondiente. Un subVi puede ser Express Vis, que es un nodo que se configura con ventanas de diálogo por lo que requiere un mínimo cableado y es utilizado para tareas de medidas comunes. El empleo de subVIs ayuda a gestionar más eficientemente modificaciones y eliminar errores del diagrama de bloques, además de facilitar los esquemas de cableado mejorando su comprensión.

### **Paleta de funciones**

Contiene los Vis, funciones y constantes que se utilizan para diseñar el diagrama de bloques. La paleta se divide en categorías conteniendo todos los objetos utilizados en la implementación del programa VI, tales como funciones aritméticas, de entrada/salida de señales, entrada/salida de datos a fichero, entre otras. Se tiene acceso a la paleta de funciones del diagrama de bloques seleccionando View » **Functions Palette**. (Ver figura 3.12)

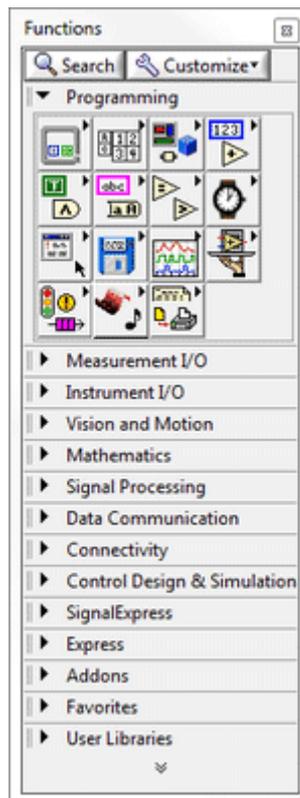


Figura 3. 12: Paleta de funciones  
Elaborado por: Autor

### 3.2. Programación *LabView*

La programación en *LabView* se la empieza desde el panel frontal. Definiéndose primero, los controles o entradas de datos que proporcionará el usuario y los indicadores o salidas que mostrarán los resultados en pantalla. Luego de realizada la selección de los objetos necesarios se pasa a la ventana *Diagram* que corresponde al diagrama de bloques y que es donde se lleva a cabo la programación.

Una vez en la ventana del diagrama de bloques, los terminales correspondientes a los objetos seleccionados en el panel frontal se encuentran dispuestos de manera automática. Las funciones, estructuras y

demás elementos necesarios para el diseño del programa se van ubicando y se conectan a los terminales por medio de cables

### **3.3. Ejecución *LabView***

La ejecución se realiza al finalizar la programación. En la ventana del panel frontal se pulsa el botón *Run* de la barra de herramientas y el programa se ejecutará una sola vez.

## CAPÍTULO 4: SIMULACION Y RESULTADOS OBTENIDOS

### 4.1. Simulación 1. Gráfica de la función $\text{Sen}$ .

1. Abrir un nuevo VI (ver figura 4.1)

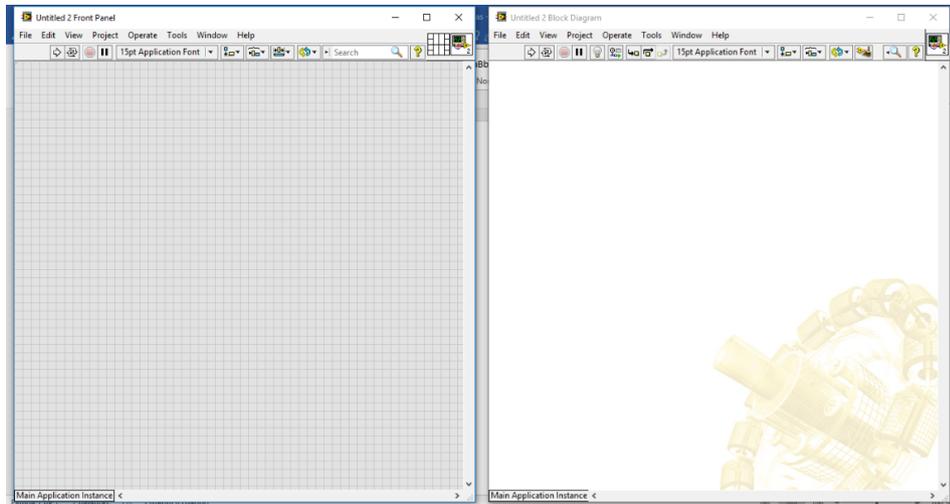


Figura 4. 1: Nuevo VI.

Elaborador por: Autor

2. En el panel frontal con el click derecho del ratón se busca el control “*Graph*”: *Waveform Graph*, obteniéndose como se muestra en las figuras 4.2, 4.3 y 4.4

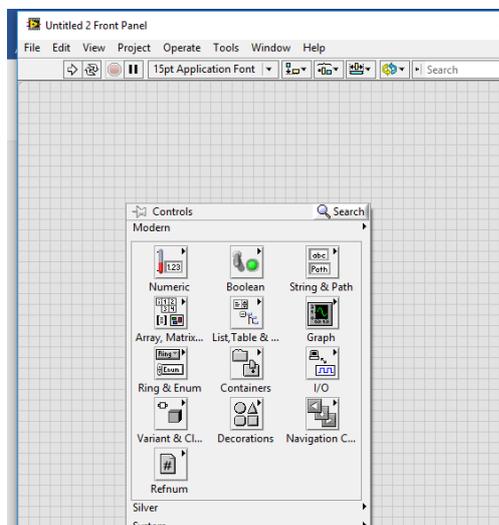


Figura 4. 2: Panel frontal.

Elaborador por: Autor

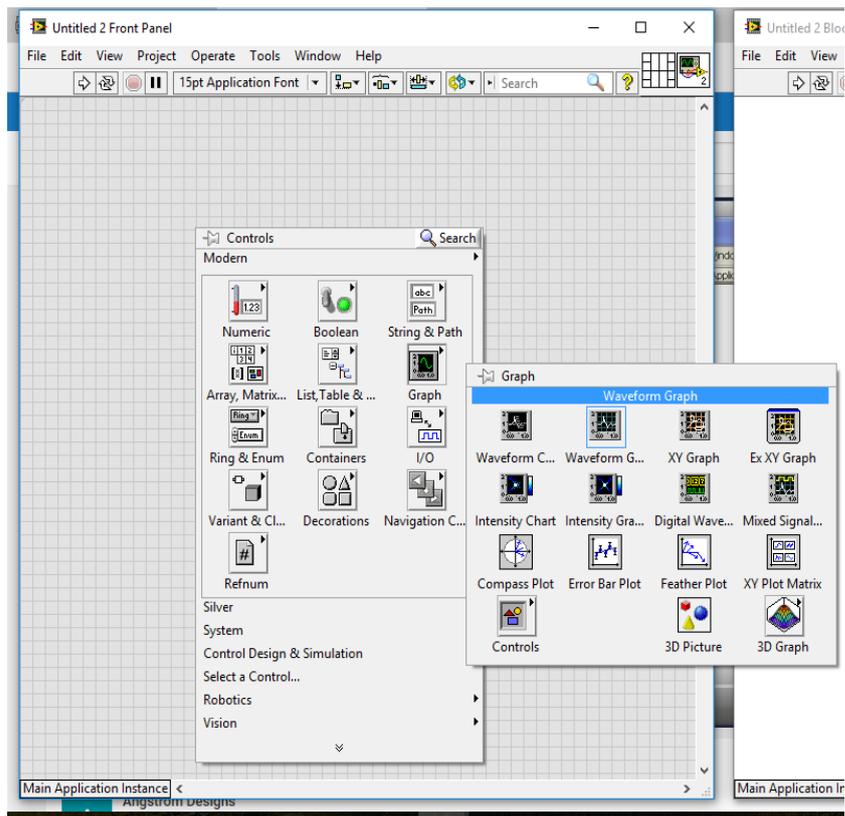


Figura 4. 3: Panel frontal – control “graph”.  
Elaborador por: Autor

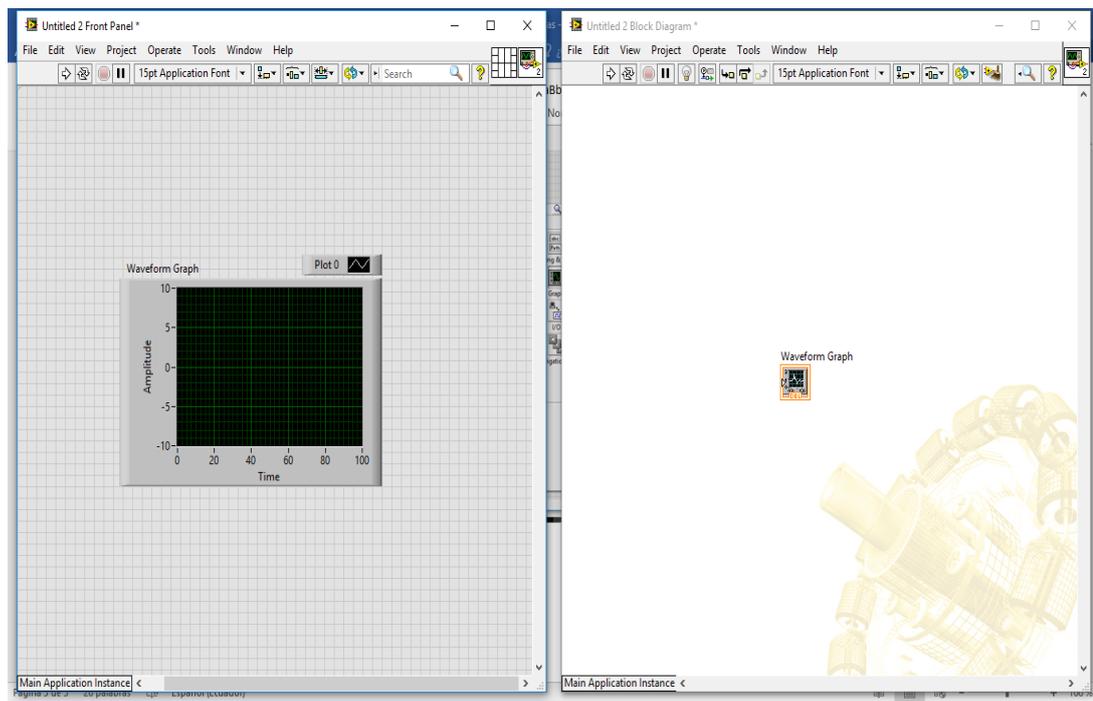


Figura 4. 4: Control “waveform graph”.  
Elaborador por: Autor

3. En la ventana de panel frontal se añade, dando click derecho al ratón, dos controles numéricos: “*Numeric Control*” cambiando el nombre (dando doble click en el control numérico respectivo) como se ilustra en la figura 4.5.

En uno de ellos “Número de puntos” (el cual nos va a dar el período de la función *sen*) y en el otro “Amplitud” (para cambiar la amplitud de la señal *sen*) como se muestra en la figura 4.6.

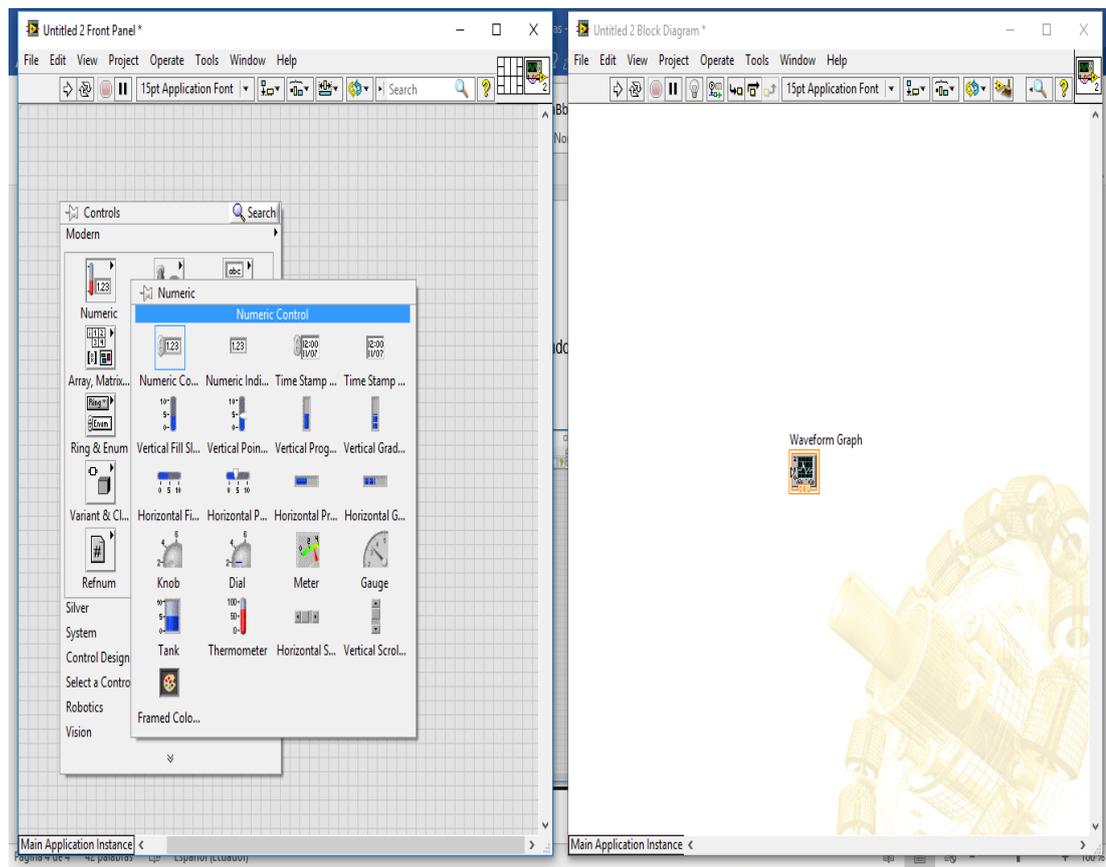


Figura 4. 5: Front panel y Block Diagram – control numeric “numeric control”.

Elaborador por: Autor

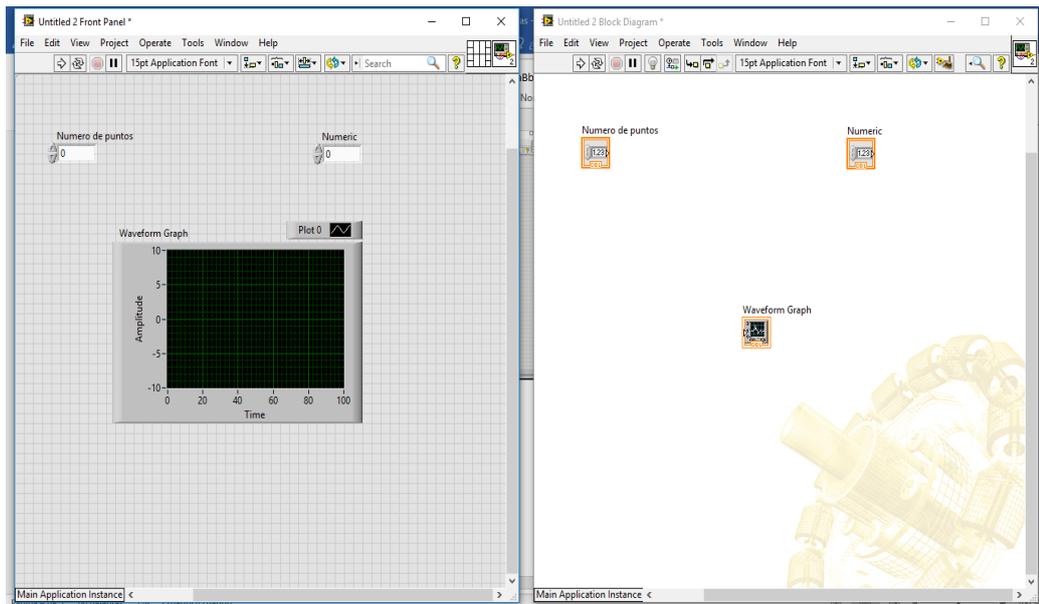


Figura 4. 6: Controles numéricos ingresados.  
Elaborador por: Autor

4. En la ventana Diagrama de Bloques se agrega una estructura *For Loop*, dándole click derecho al ratón en la opción de *Functions* (ver figura 4.7)

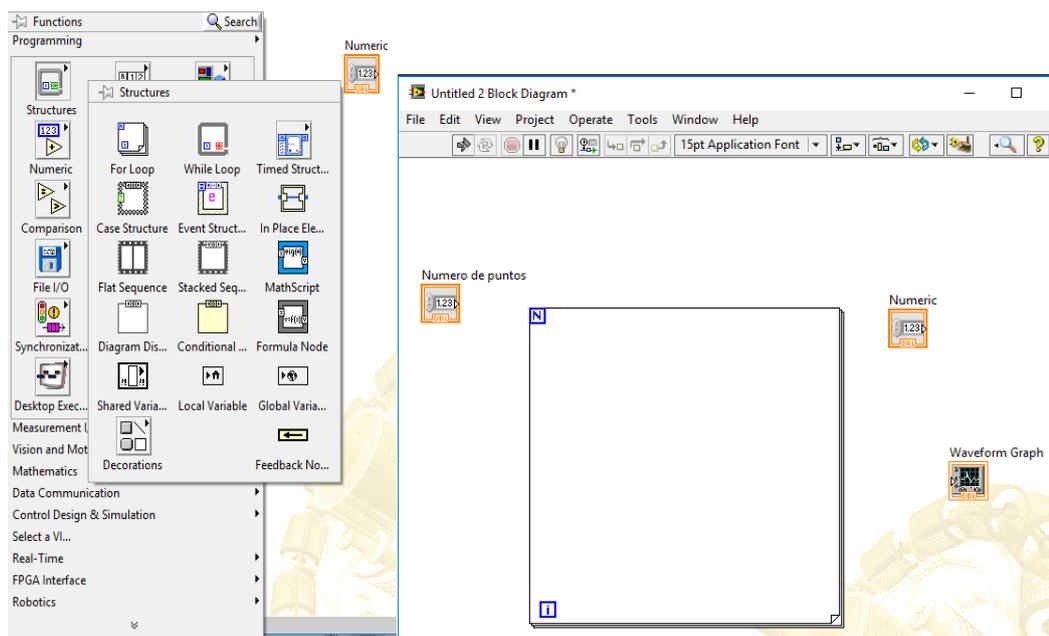


Figura 4. 7: Estructura "For Loop" ingresada.  
Elaborador por: Autor

5. En la ventana Diagrama de Bloques, se da click derecho al ratón y se busca en <Functions: Mathematics: Elementary & Special Functions: Trigonometric Functions: sin> (ver figura 4.8), realizando luego las conexiones iniciales mostradas (se conecta el inicio del lazo al bloque sin y el bloque Número de puntos al fin del lazo n, lo que dará el período de la señal a graficar). (Véase la figura 4.9)

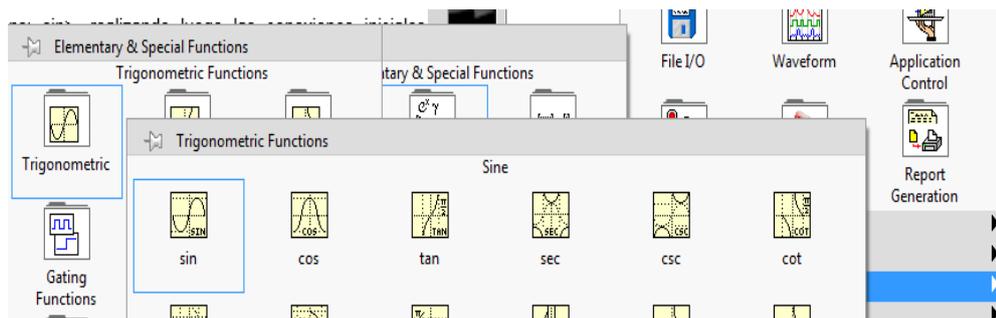


Figura 4. 8: Función trigonométrica “sin”.  
Elaborador por: Autor

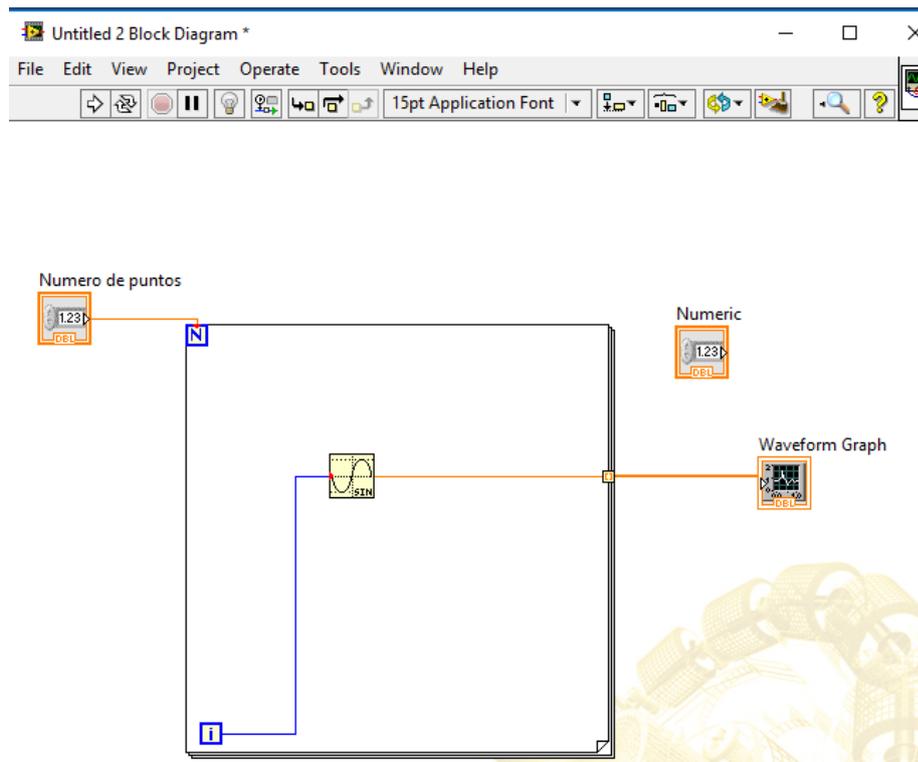


Figura 4. 9: Conexiones de indicadores.  
Elaborador por: Autor

Probando tal como está, por ejemplo colocando 100 en el control numérico de número de puntos, y pulsando el botón de “run”, se obtiene la gráfica mostrada en la figura 4.10.

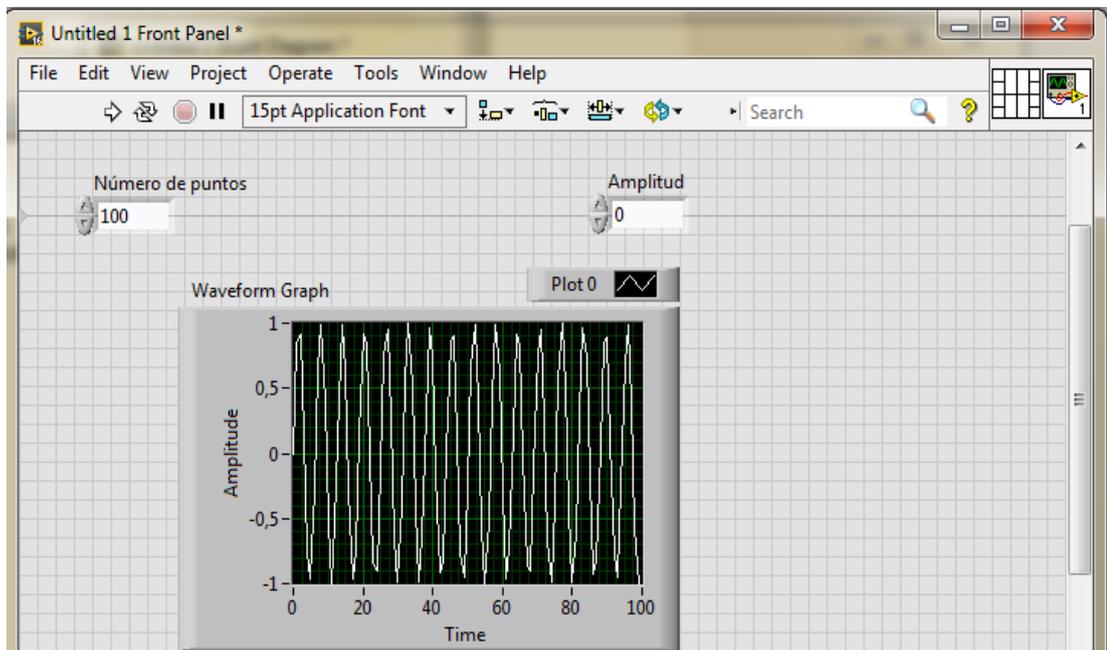


Figura 4. 10: Función sen con un valor de 100 en número de puntos.

Elaborador por: Autor

Todavía no se obtiene adecuadamente la señal *sen*, cuyo periodo es 100.

6. Se coloca dando click derecho en la ventana Diagrama de Bloques, *<Functions, Numeric, Math & Scientific Constants, 2π>* como se muestra en la figura 4.11.

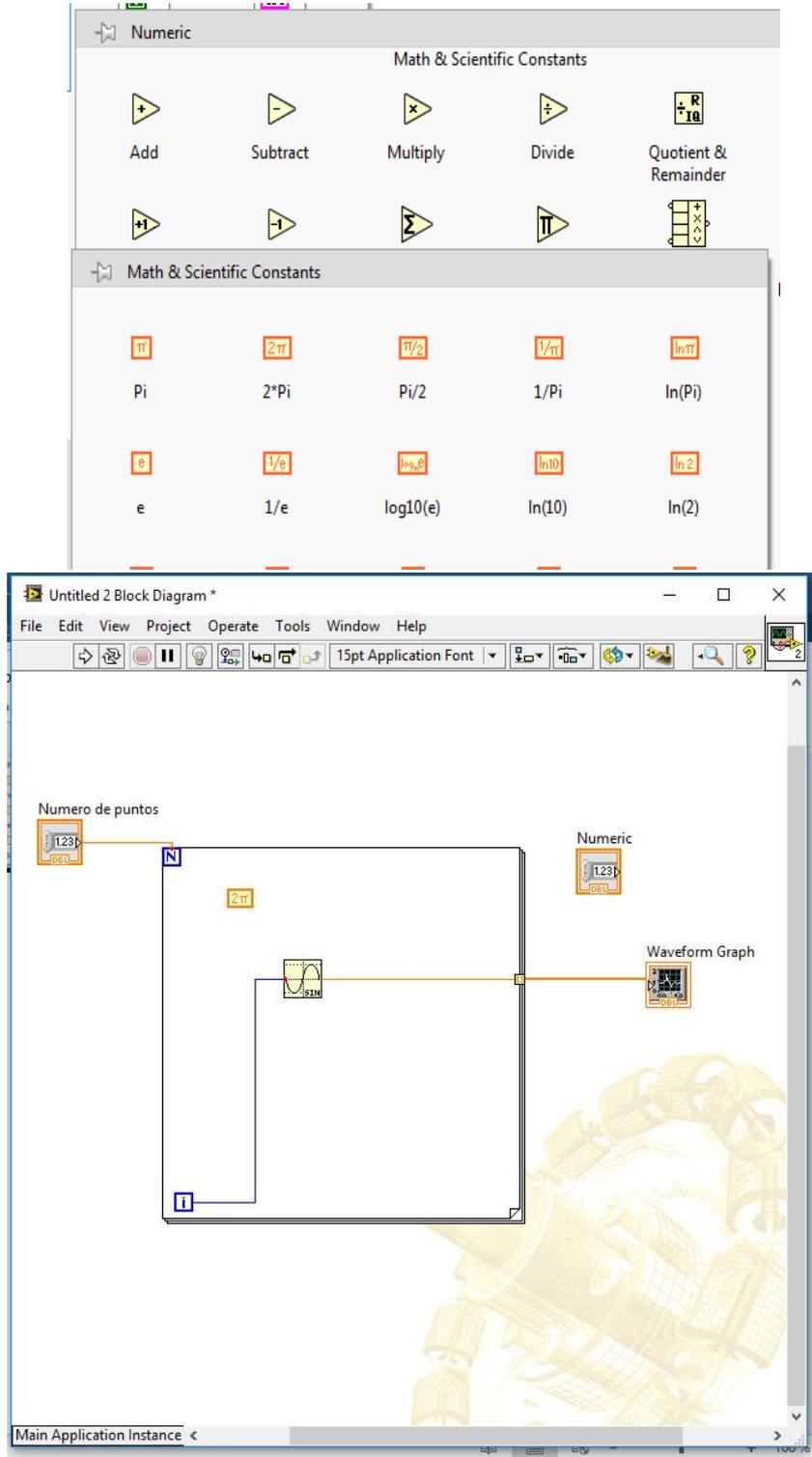


Figura 4. 11: Constante  $2\pi$  agregada en la estructura for loop.  
Elaborador por: Autor

Luego se agrega el bloque de división (click derecho en ventana diagrama de bloques: <Functions, Numeric, Division>), luego el bloque de multiplicación (click derecho en ventana diagrama de bloques: <Functions, Numeric, Multiply>) como se muestra en la figura 4.12. Luego se realiza las siguientes conexiones para obtener la frecuencia angular adecuada (inverso de periodo, considerando la fórmula  $\omega = \frac{2\pi}{T}$ , donde  $T$  es el número de puntos dado).

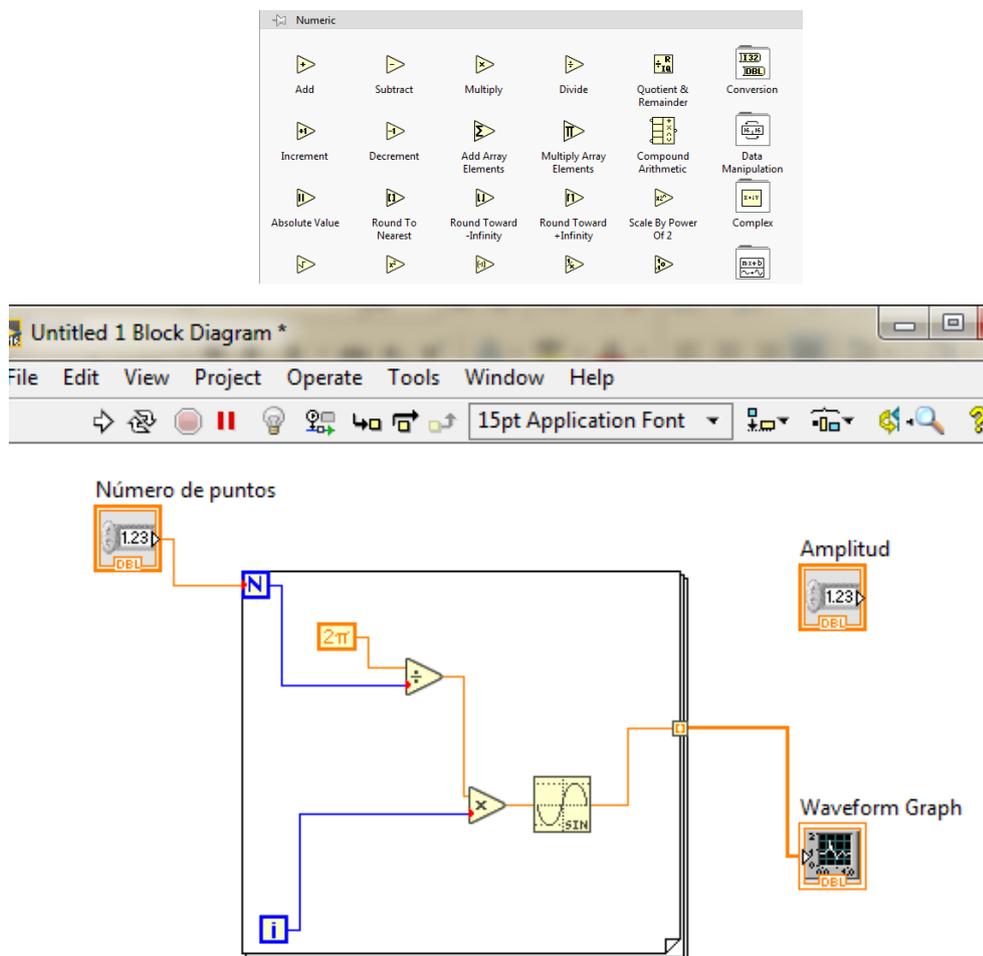


Figura 4. 12: Funciones “división” y “multiply”.  
Elaborador por: Autor

Obteniendo la gráfica deseada (se muestra un ciclo). Observe que la amplitud es 1, hasta este momento no se puede variar. (Ver figura 4.13)

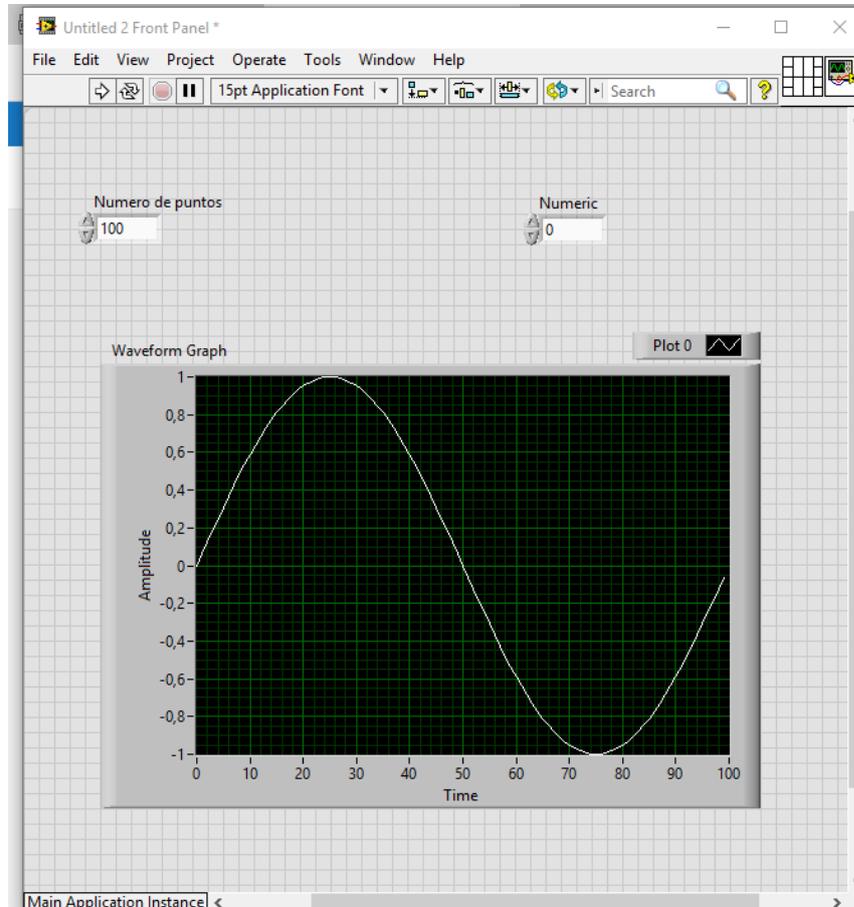


Figura 4. 13: Función sen con amplitud 1.  
Elaborador por: Autor

7. Para variar la amplitud se agrega el bloque de multiplicación (click derecho en ventana diagrama de bloques: *<Functions, Numeric, Multiply>*) y se lo conecta al bloque de Amplitud y bloque *sen*:  
Probando con amplitud 4 y periodo 100 como se muestra en la figura 4.14

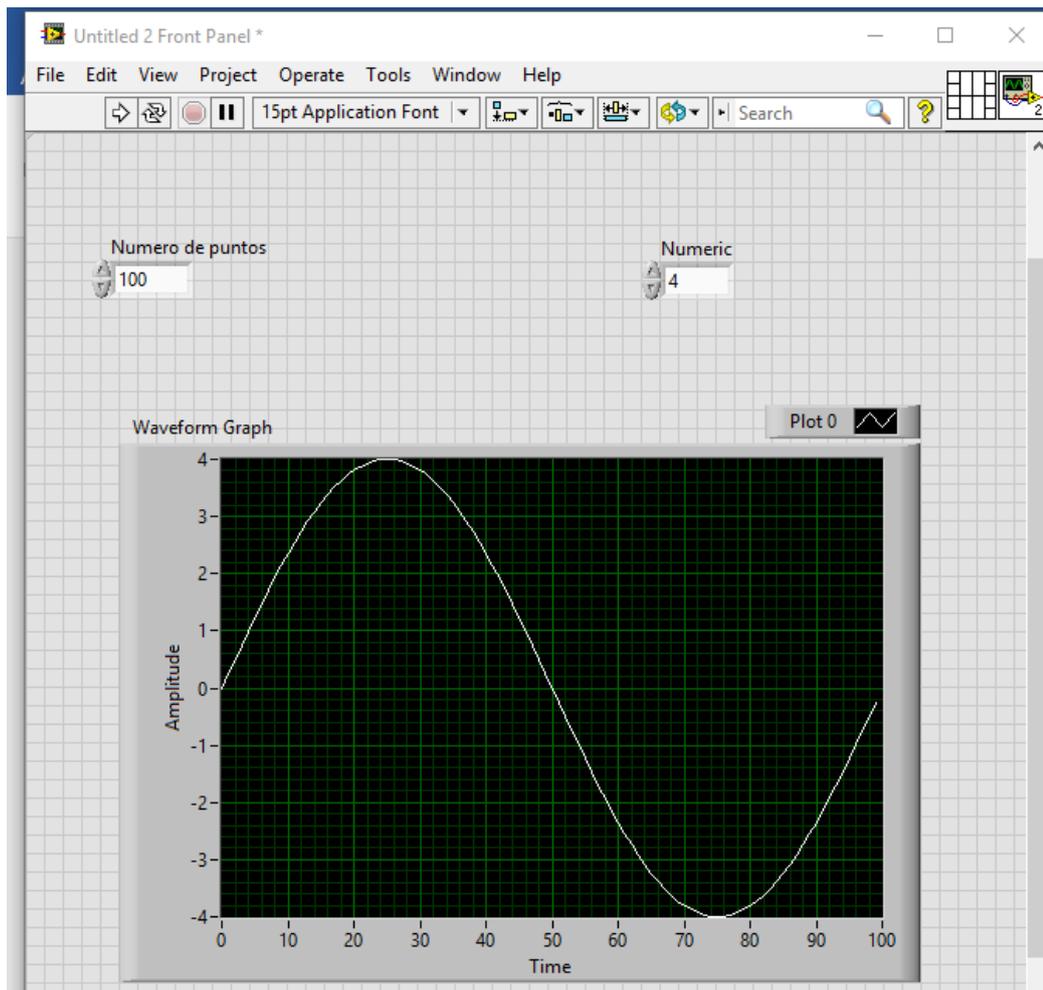


Figura 4. 14: Función sen con amplitud 4.  
Elaborador por: Autor

8. Se procede a guardar el VI en este caso con el nombre sen. Tomar en cuenta que para número de puntos menores que 30 la gráfica no sale bien debido a que son pocos puntos para poder obtener un dibujo adecuado.

## 4.2. Simulación 2. Transformada de Fourier

Como se indicó anteriormente, se debe iniciar creando un VI nuevo.

1. En la ventana Diagrama de Bloques con el click derecho del ratón buscar en *<Functions; Programming; Structures; While Loop>* para ingresar la nueva estructura como se muestra en la figura 4.15.

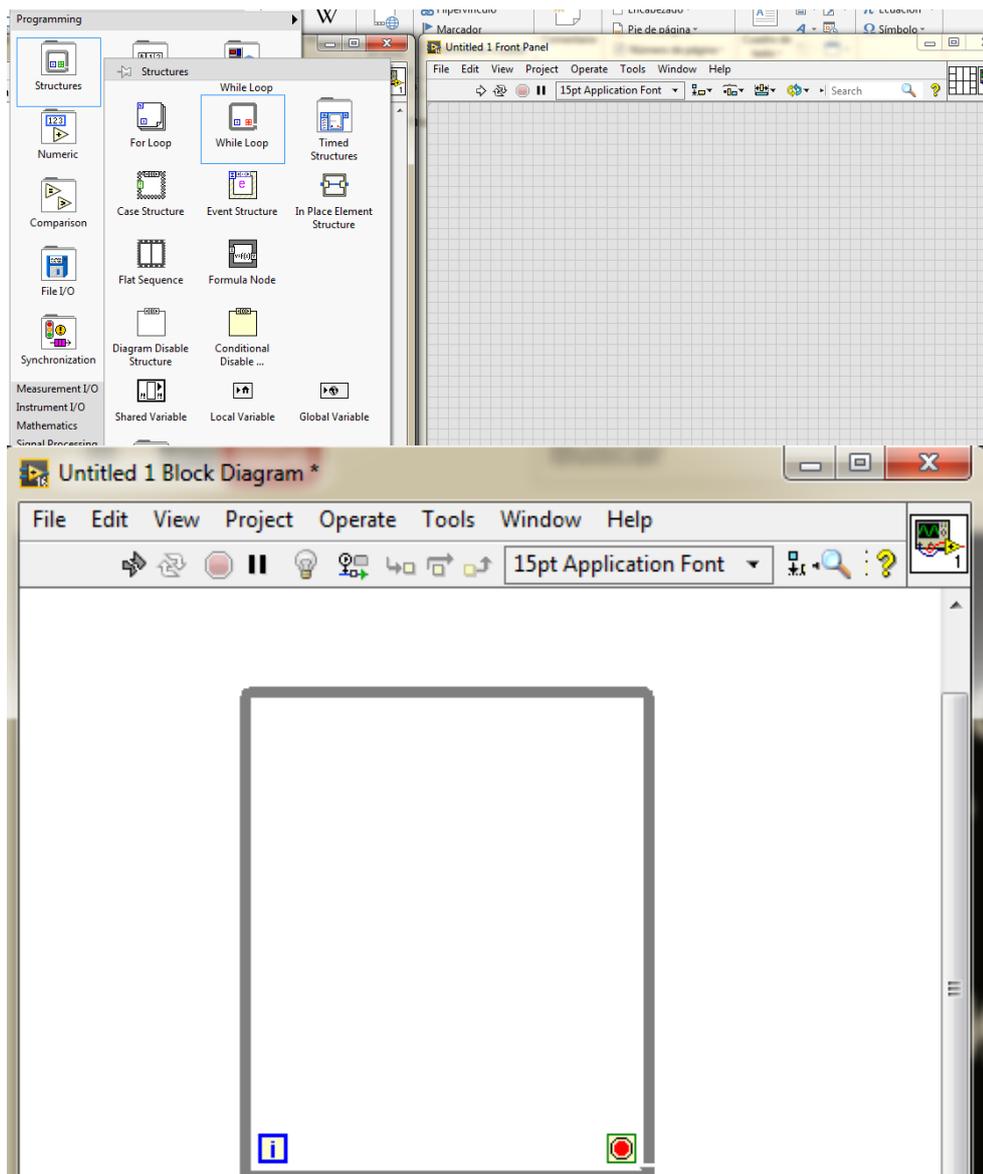


Figura 4. 15: Estructura while loop  
Elaborador por: Autor

- Para crear el control “stop” se debe dar click derecho en el botón rojo, y colocar *<Create Control>*, como se muestra a continuación en la figura 4.16.

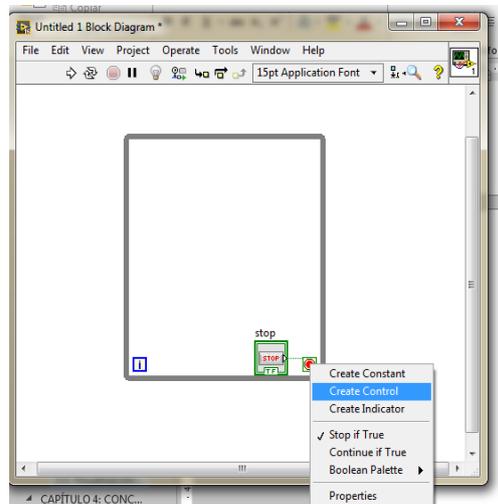


Figura 4. 16: Creación de un control.  
Elaborador por: Autor

- En la ventana diagrama de bloques se da click derecho *<Functions, Signal Proccesing; Waveform Generation; Simulate Signal>* para insertar lo que será el simulador, el proceso se muestra en la figura 4.17.

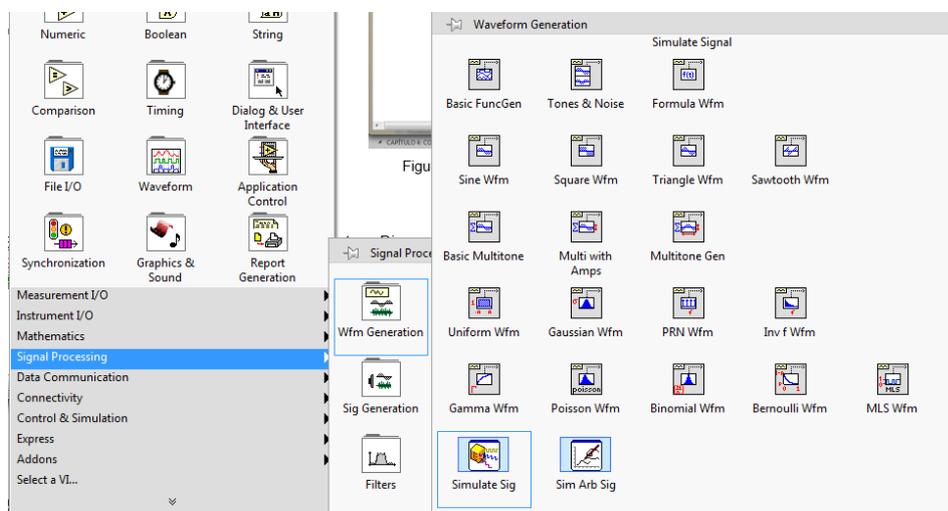


Figura 4. 17: Función “simulator signal”.  
Elaborador por: Autor

4. Luego para configurar la señal se agregan los siguientes valores en la ventana con los siguientes valores:

$$f = 0,05 \text{ Hz};$$

$$\text{Amplitude} = 1;$$

$$\text{Samples per second (Hz)} = 20;$$

$$\text{Number of samples} = 1024$$

En la figura 4.18 se muestran los datos ingresados.

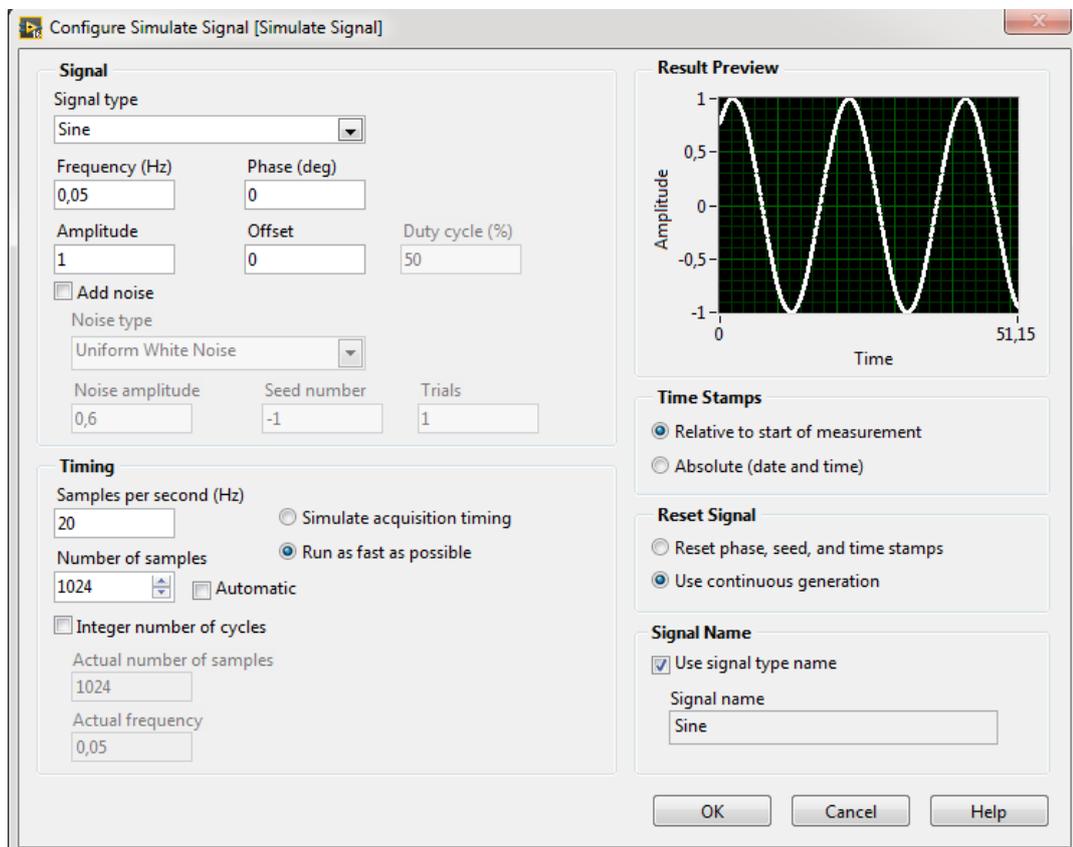


Figura 4. 18: Configuración del simulador de señal.

Elaborador por: Autor

5. Luego se copiará otro bloque *<Simulate Signal>* y se configurará con los siguientes valores para una nueva observación:

$$f = 0,25 \text{ Hz};$$

$$\text{Amplitude} = 0,5;$$

$$\text{Samples per second (Hz)} = 20;$$

$$\text{Number of samples} = 1024$$

6. En la figura 4.19 se muestra la nueva configuración

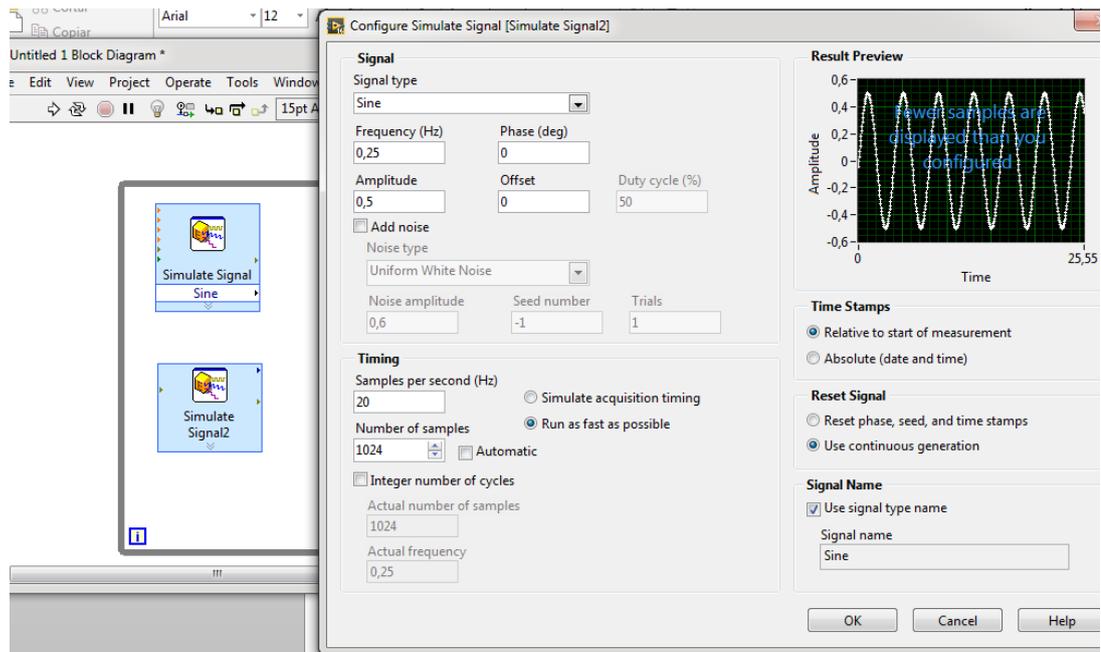


Figura 4. 19: Configuración del segundo simulador de señal.

Elaborador por: Autor

7. En la ventana diagrama de bloques, se da click derecho al ratón y se busca en *<Functions: Programming: Waveform; Analog Waveform: Measurements; Spectral>* como se muestra en la figura 4.20. Para poder configurar esta función, se selecciona *<Selected Measurement: Power Spectrum; Result: Linear>* dar click en OK, como se ilustra en la figura 4.21

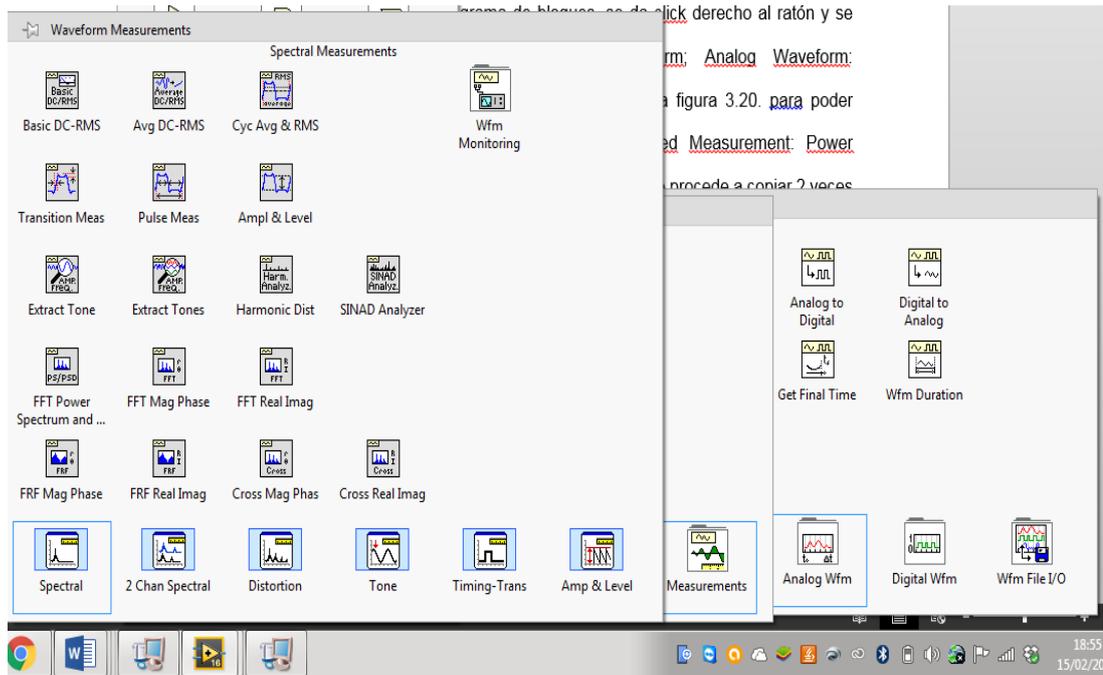


Figura 4. 20: Función "spectral".  
Elaborador por: Autor

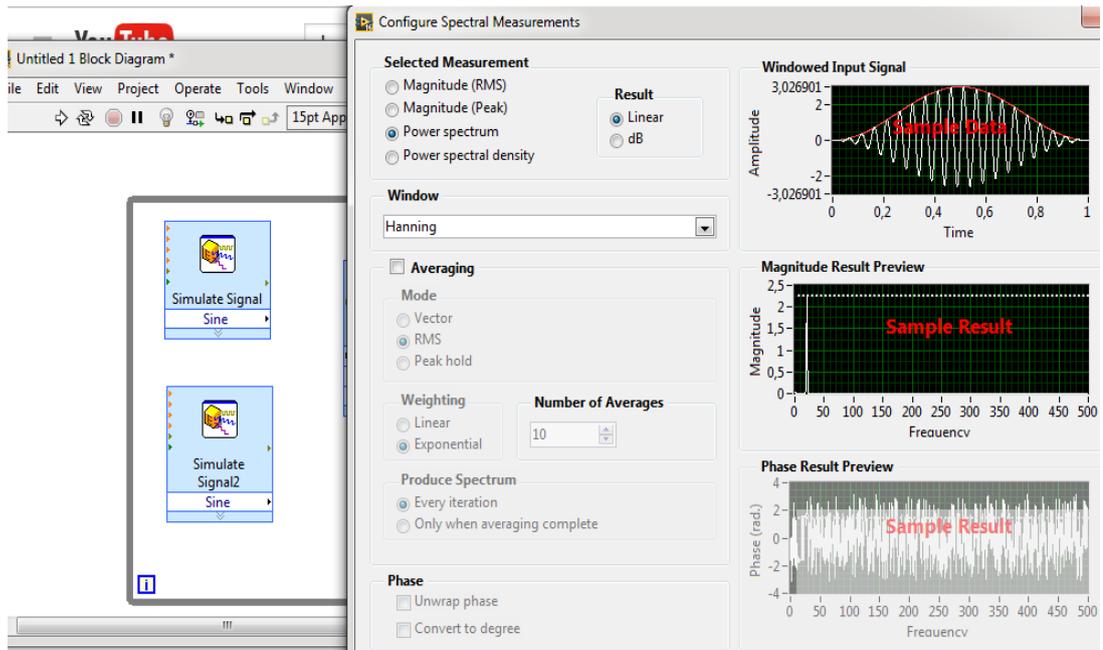


Figura 4. 21: Configuración del espectro.  
Elaborador por: Autor

8. Luego se procede a copiar 2 veces el bloque *<Spectral>* para obtener nuevas observaciones del simulador (ver figura 4.22).

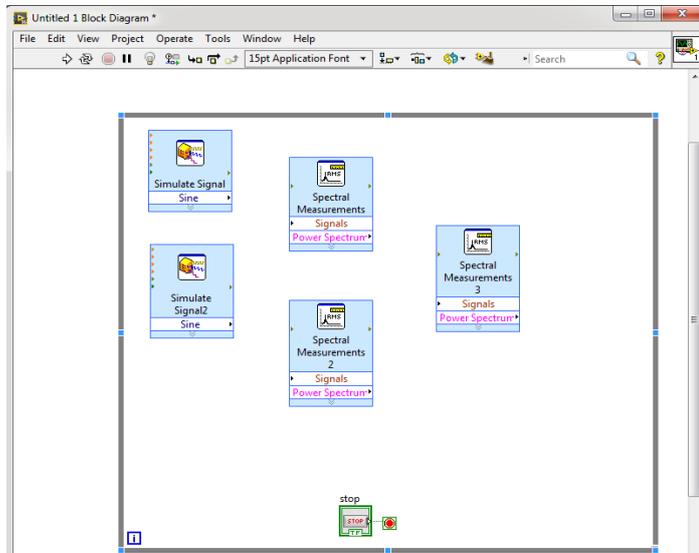


Figura 4. 22: Ventana “block diagram”  
Elaborador por: Autor

9. Ahora se configura el panel frontal creando “Graph” el cual se coloca dando click derecho en la ventana panel frontal, *<Controls, Express, Graph Indicators, Graph>*. Luego se copia 5 veces el mismo control en la ventana panel frontal (ver figura 4.23).

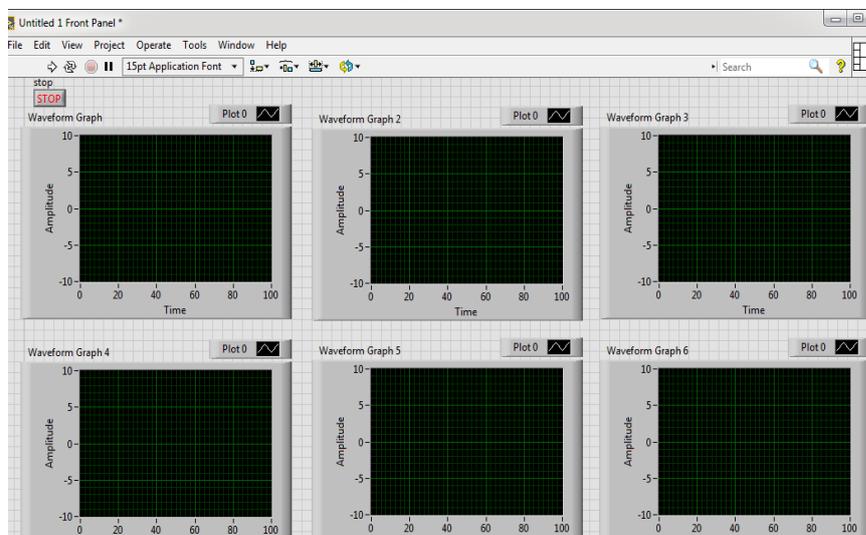


Figura 4. 23: Ventana “front panel”  
Elaborador por: Autor

10. Realizando luego las conexiones iniciales, conectando los indicadores gráficos a cada simulador de señal y a los medidores espectrales, para esto se grafica el procedimiento en la figura 4.24

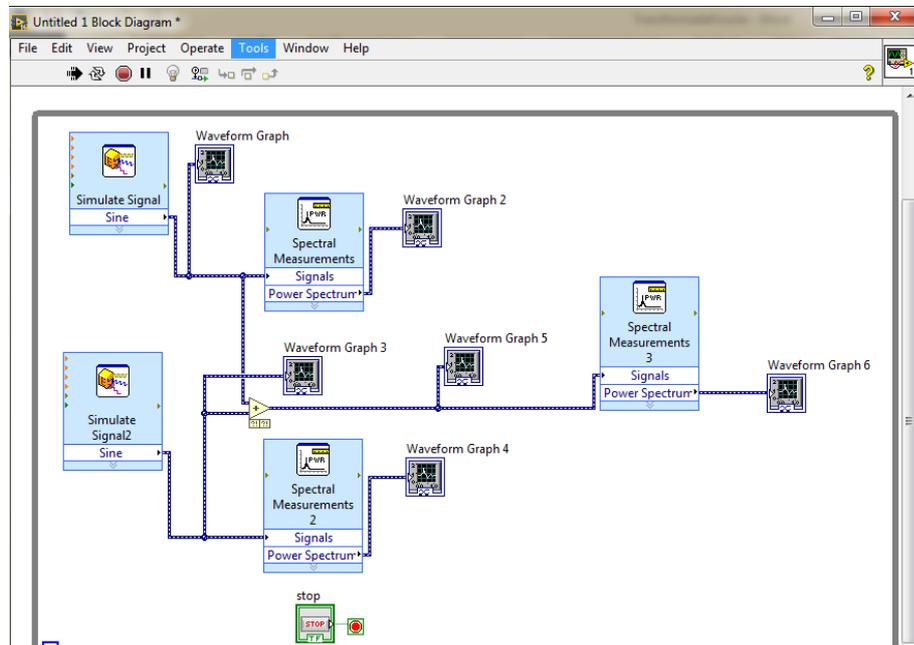


Figura 4. 24: Ventana “block diagram” con las conexiones realizadas.  
Elaborador por: Autor

11. Se procede a correr el programa dando como resultado lo mostrado en la figura 4.25

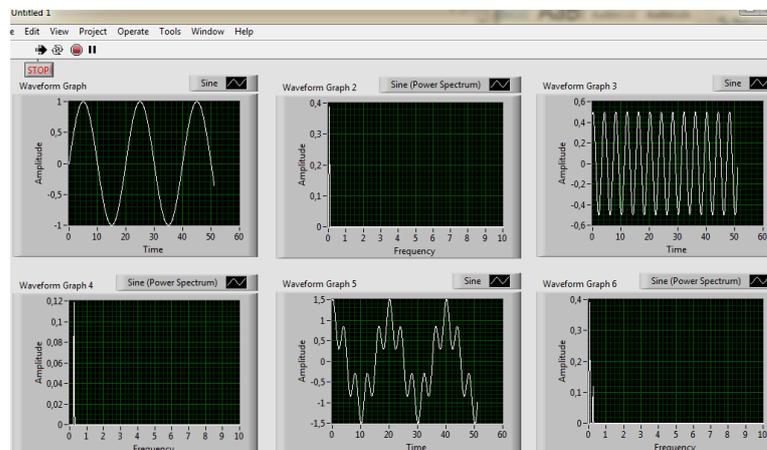


Figura 4. 25: Resultado final “transformada de Fourier”.  
Elaborador por: Autor

### 4.3. Simulación 3. Teorema de Parseval

En esta simulación se demuestra la relación de Parseval en tiempo discreto, la cual establece que la energía total en una señal es la misma en ambos dominios: tiempo y frecuencia.

1. Una vez creado el VI nuevo, en la ventana diagrama de bloques, se crea 2 “sine patter” esto se realiza desde *<Functions: programming; signal processing: sig generation; sine pattern>*, como se muestra en la figura 4.26.

Adicional se crea un “uniform noise” que se encuentra en *<Function: programming; signal processing: sig generation; uniform noise>*

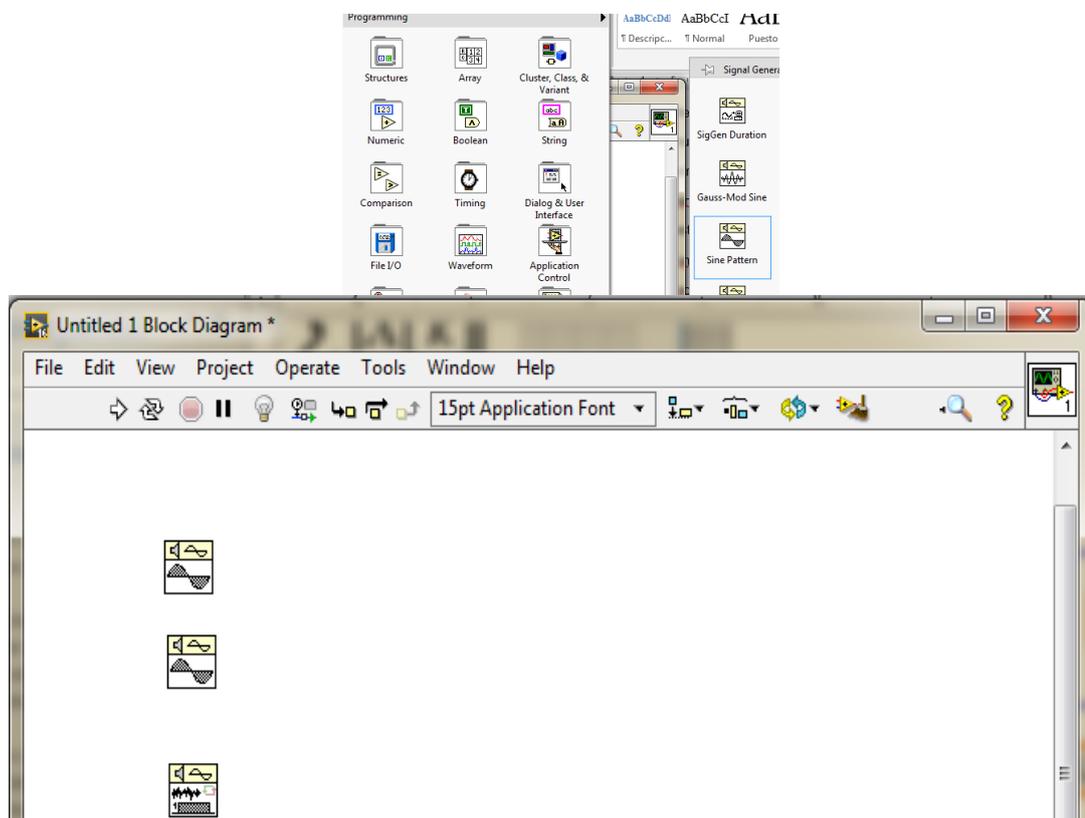


Figura 4. 26: Funciones “sine pattern” y “uniform noise” - diagrama de bloque.  
Elaborador por: Autor

- Se colocan 3 “DBL numeric constant” para esto se debe seleccionar `<function: numeric; “DBL numeric constant”>` como se muestra en la figura 4.27.

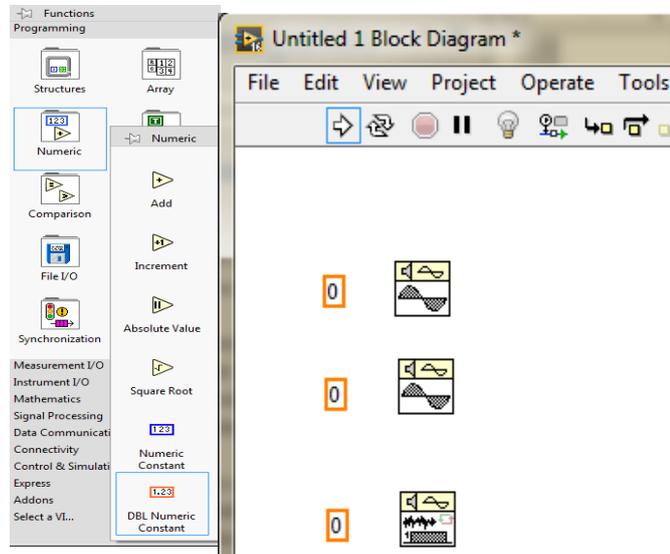


Figura 4. 27: DBL numeric control - diagrama de bloque.  
Elaborador por: Autor

- Luego se ingresa varias funciones numéricas que son las que ejecutaran el procedimiento para la demostración del teorema de Parseval. Dichas componentes se pueden observar en la figura 4.28.

4.

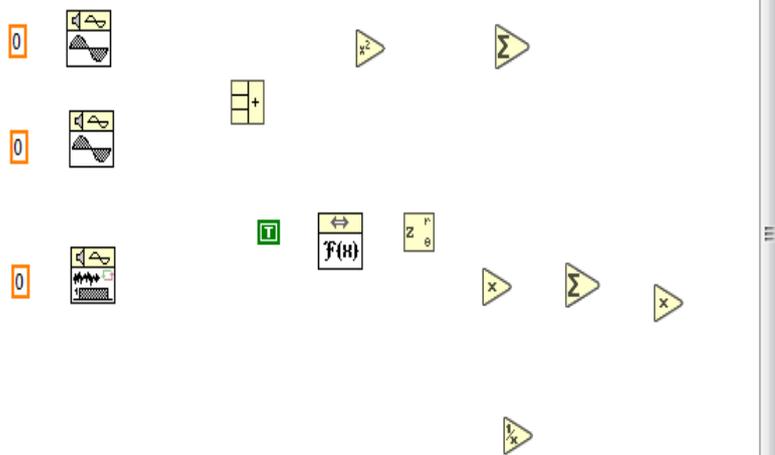


Figura 4. 28: funciones numéricas - diagrama de bloque.  
Elaborador por: Autor

- Se procede hacer las respectivas conexiones entre las funciones numéricas y las funciones de señales ingresadas anteriormente. (Ver figura 4.29)

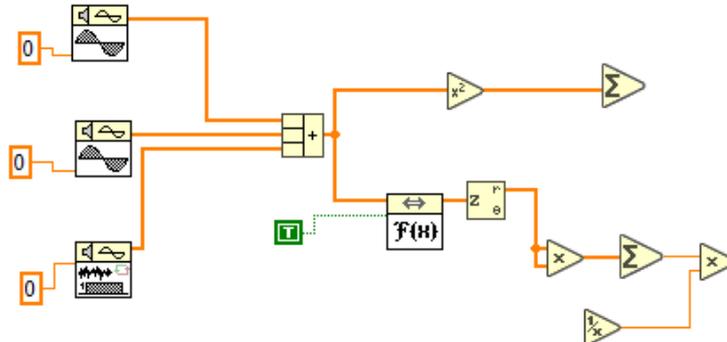


Figura 4. 29: Funciones en diagrama de bloque.  
Elaborador por: Autor

- Se ingresan 4 “to double precisión float” desde <Function: numeric; conversión: “to double precision float”> como se muestra en la figura 4.30. Éstos indicaran lo valores resultantes de las operaciones.

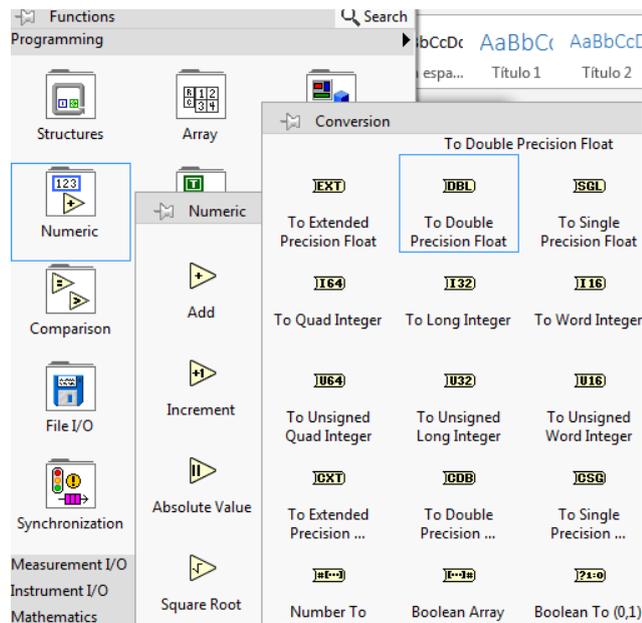


Figura 4. 30: Conversión “to double precision float”  
Elaborador por: Autor

Las dos primeras DBL indican la suma de los cuadrados de la señal en el dominio del tiempo. Mientras que los dos DBL restantes indican la suma de los valores de la señal en el dominio de la frecuencia.

- Se realizan las conexiones y se ingresa un “numeric constant”, esta constante es la que permite establecer un valor para la muestra. (Ver figura 4.31).

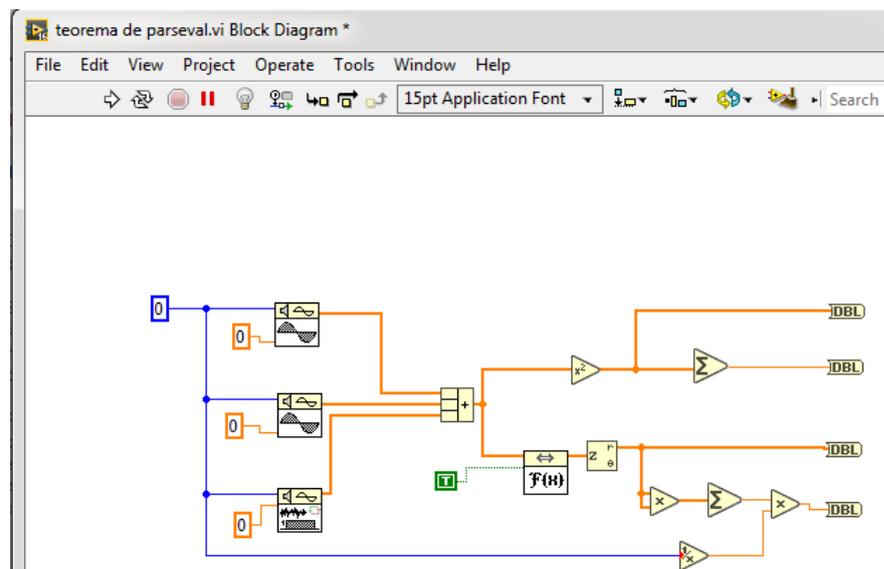


Figura 4. 31: Numeric constant aplicado.  
Elaborador por: Autor

- Para observar la gráfica de la señal es necesario agregar dos “waveform graph” y conectarlos a los convertidores DBL, también se agregan dos indicadores numéricos, estos indicadores mostrarán en pantalla el resultado obtenido, se agregan y se conectan a los convertidores DBL. (Ver figura 4.32)

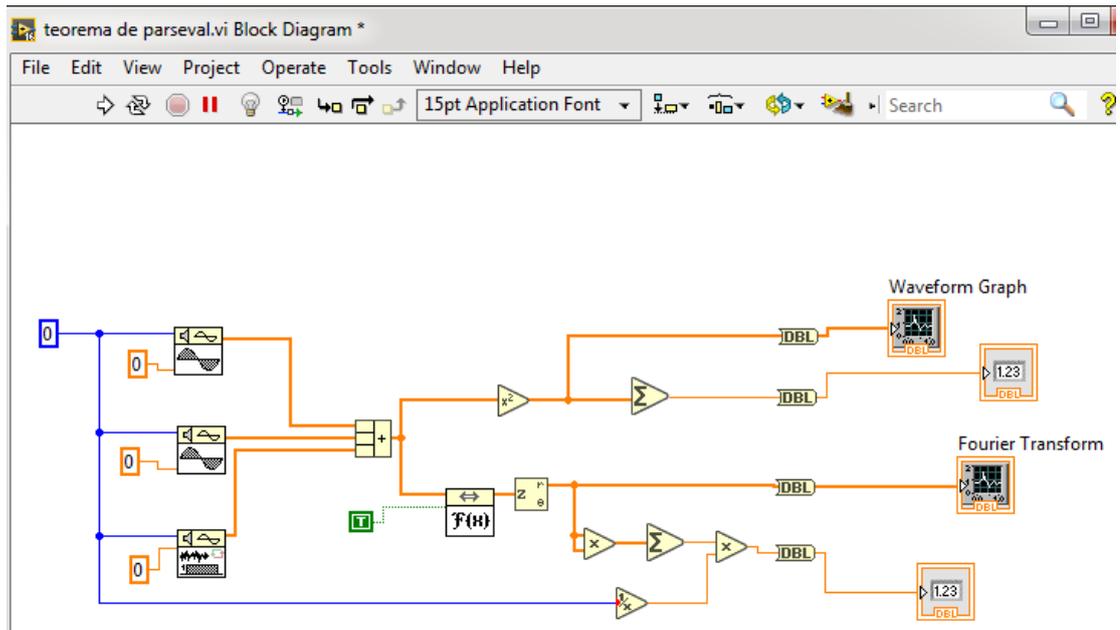


Figura 4. 32: Teorema de Parseval – diagrama de bloque.  
Elaborador por: Autor

En esta simulación se tomara como ejemplo una muestra de 1024, que se registra en el diagrama de bloques en la constante numérica.

9. En la ventana de Front panel se ingresan dos *waveform graph* y dos indicadores numéricos que grafican las ondas producidas por el teorema realizado en el diagrama de bloque. (Ver figura 4.33)

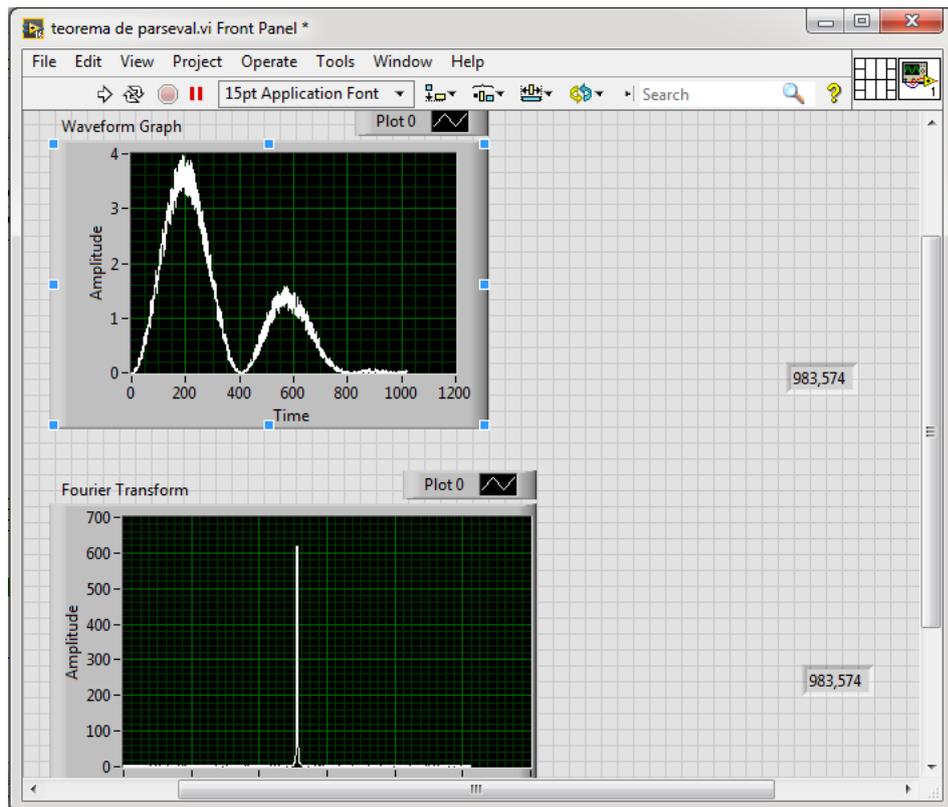


Figura 4. 33: Teorema de Parseval – Front panel.  
Elaborador por: Autor

#### 4.4. Simulación 4. Modulador FM

En la figura 4.34 se muestra lo que sería la configuración en la ventana diagrama de bloque para un modulador FM, aplicando funciones dentro del programa LabView.

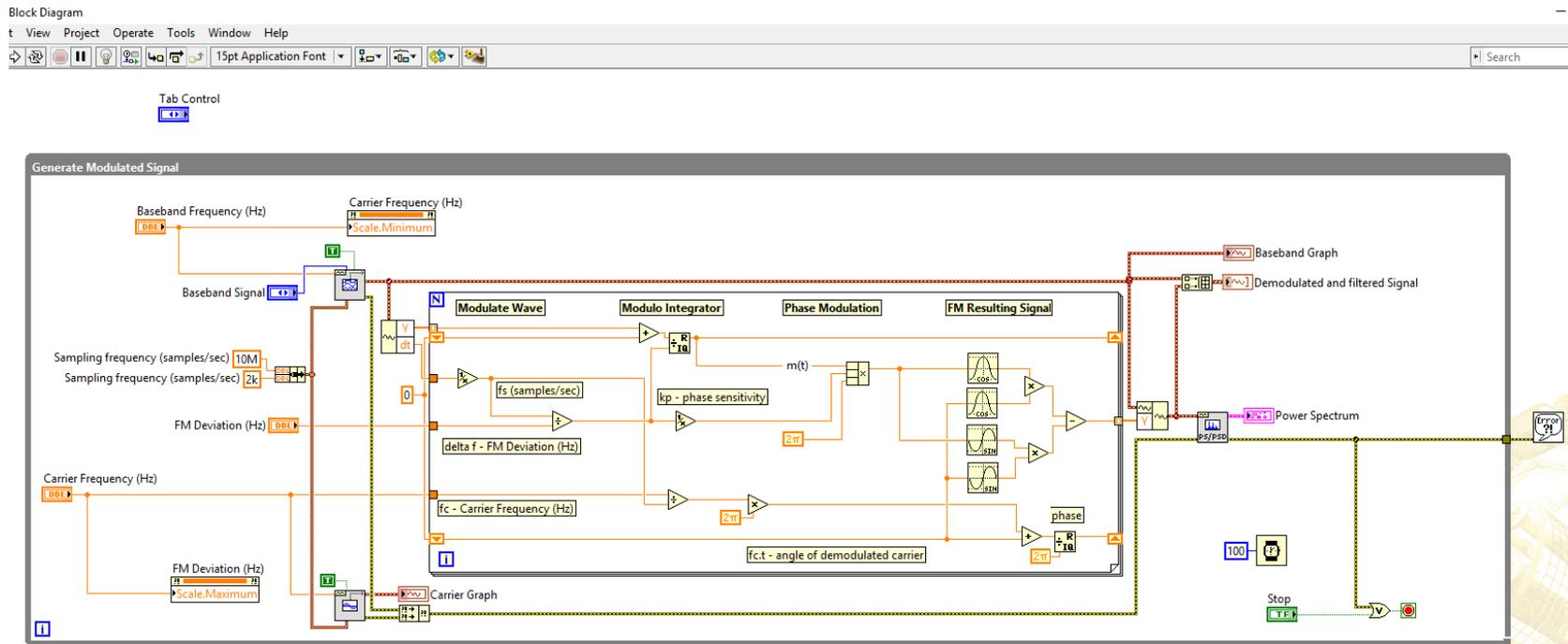


Figura 4. 34: Modulador FM en diagrama de bloque.  
Elaborador por: Autor

## **CAPÍTULO 5: CONCLUSIONES Y RECOMENDACIONES.**

### **5.1. Conclusiones.**

- Analizando los principios de los tipos de sistemas como son los continuos y discretos, se llegó a la conclusión que el sistema continuo presenta varias ventajas sobre el sistema discreto, por ser un sistema análogo es poco probable que la señal se vea distorsionada por ruidos en el ambiente, otra notable ventaja del sistema es que la señal resultante puede ser ampliada y restaurada a la vez.
  
- Mediante la fundamentación teórica de los diferentes conceptos que se detalló en el transcurso de estos capítulos, se observa que existen diferentes tipos de señales y que cada una de estas se puede calcular bajo varios parámetros de cálculos matemáticos, como lo son: Series de Fourier, Transformada de Fourier, Transformada de Laplace, etc.
  
- Al implementar la Instrumentación virtual utilizando LabView a través de prácticas en el software, ejercicios de refuerzos para la materia señales y sistemas, se puede observar de manera interactiva los diferentes tipos de funciones que se pueden dar como resultado, y en base a este grafico se puede comparar y ver qué tipo de filtro se puede aplicar a esta señal para que sea la más óptima a ser propagada en un medio de transmisión para cierto sistema determinado.

## **5.2. Recomendaciones.**

- Realizar cursos sobre el uso de la herramienta LabView, para así profundizar y poner en práctica todo lo aprendido dentro del aula de clases en la materia de señales y sistemas, y así ir adquiriendo experiencia para desarrollar prácticas que ayuden al estudiante en diferentes asignaturas dentro de la carrera de Ingeniería Telecomunicaciones.
  
- Promover que el uso de esta herramienta sea utilizada en el laboratorio de electrónica y telecomunicaciones de la Facultad de Educación Técnica para el Desarrollo, para así ver de forma digital la mayoría de fenómenos que ocurren dentro de una señal al ser aplicada a diferentes tipos de filtros.

## REFERENCIAS BIBLIOGRÁFICAS

Albertí, E. B., & Upc, U. E. (2009). *Procesado Digital de Seales - I Fundamentos Para Comunicaciones y Control*. Univ. Politèc. de Catalunya.

Alcaraz Domingo. (2017). Series y transformadas de Fourier. Recuperado el 13 de enero de 2017, a partir de [http://filemon.upct.es/~paredes/am\\_ti/apuntes/Tema%202.%20Series%20y%20transformadas%20de%20Fourier.pdf](http://filemon.upct.es/~paredes/am_ti/apuntes/Tema%202.%20Series%20y%20transformadas%20de%20Fourier.pdf)

Alvarado, P. (2008). *Señales y Sistemas. Fundamentos Matemáticos* (Primera). Costa Rica: Centro de Desarrollo de Material Bibliografico.

Antonio Alonso, & Federico Kuhlmann. (2017). INFORMACIÓN Y TELECOMUNICACIONES. Recuperado el 8 de enero de 2017, a partir de [http://www.cubaeduca.cu/medias/cienciatodos/Libros\\_3/ciencia3/149/htm/informac.htm](http://www.cubaeduca.cu/medias/cienciatodos/Libros_3/ciencia3/149/htm/informac.htm)

Briceño José E. (2012). Principio de las comunicaciones.

Bruzual Ramón, & Domínguez Marisela. (2003). Series de Fourier.

Cáceres Juan Pablo. (2007). Transformada de Fourier. Recuperado el 14 de enero de 2017, a partir de [https://ccrma.stanford.edu/workshops/cm2007/topics/clases/PDFs/05fourier\\_handout.pdf](https://ccrma.stanford.edu/workshops/cm2007/topics/clases/PDFs/05fourier_handout.pdf)

- Córdova Darwin. (2017). Radio Observatorio de Jicamarca. Recuperado el 4 de enero de 2017, a partir de <http://jro.igp.gob.pe/newsletter/200802/noticia3.php>
- Cornejo Serrano, M. del C., Villalobos Oliver, E. B., & Quintana Hernández, P. A. (2008). *Métodos de solución de ecuaciones diferenciales y aplicaciones*. México D.F.; Celaya [México: Reverté; Instituto Tecnológico de Celaya.
- EHU. (2017). LA TRANSFORMADA DISCRETA DE FOURIER. Recuperado el 16 de enero de 2017, a partir de <http://www.ehu.eus/Procesadodesenales/tema7/ty2.html>
- El pensante. (2017). La investigación exploratoria, qué es y en qué consiste – El pensante. Recuperado el 13 de febrero de 2017, a partir de <https://educacion.elpensante.com/la-investigacion-exploratoria-que-es-y-en-que-consiste/>
- EME. (2017). Introducción a filtros digitales. Recuperado el 17 de enero de 2017, a partir de <https://www.eumus.edu.uy/eme/ensenanza/electivas/dsp/presentaciones/clase10.pdf>
- Essick, J. (2016). *Hands-on introduction to LabVIEW for scientists and engineers* (Third edition). New York: Oxford University Press.
- Gómez Emilia. (2009). Introducción al filtrado digital. Recuperado a partir de <http://www.dtic.upf.edu/~egomez/teaching/sintesi/SPS1/Tema7-FiltrosDigitales.pdf>

Hernández Héctor. (2017). Series de Taylor. Recuperado el 13 de enero de 2017, a partir de

[http://webdelprofesor.ula.ve/ciencias/hector/prontuario/metodos2/S02\\_C05.pdf](http://webdelprofesor.ula.ve/ciencias/hector/prontuario/metodos2/S02_C05.pdf)

James, G., & Burley, D. (2002). *Matemáticas avanzadas para ingeniería*. Pearson Educación.

Kamen Edward W., & Heck Bonnie S. (2008). *Fundamentos de Señales y Sistemas usando la Web y MatLab* (Tercera). Mexico: Pearson Educación. Recuperado a partir de

<http://eva.sepyc.gob.mx:8383/greenstone3/sites/localsite/collect/ciencia1/index/assoc/HASHb7cd.dir/33040070.pdf>

Kulkarni S.R. (2012). Information Signal - Course Notes. Recuperado el 8 de enero de 2017, a partir de

[https://www.princeton.edu/~cuff/ele201/kulkarni\\_text/digitizn.pdf](https://www.princeton.edu/~cuff/ele201/kulkarni_text/digitizn.pdf)

Morales Luis. (2017). Sistemas en Tiempo Discreto. Recuperado a partir de <http://www.dicis.ugto.mx/profesores/ljavier/documentos/Lec03%20-%20Sistemas%20en%20Tiempo%20Discreto.pdf>

Moreno, L., Garrido, S., & Balaguer, C. (2003). *Ingeniería de control: modelado, análisis y control de sistemas dinámicos*. Barcelona: Editorial Ariel.

Morón Jose. (2011). *Señales y Sistemas*. Venezuela: Fondo Editorial Biblioteca. Recuperado a partir de

<http://www.uru.edu/fondoeditorial/libros/pdf/moron/Mor%C3%B3n%20completo.pdf>

Oppenheim, A. V., Willsky, A. S., & Nawab, S. H. (2015). *Señales y Sistemas*. Pearson Educación.

Pinoargote Edwin. (2016). *Implementación de filtros digitales FIR e IIR utilizando MatLab/Simulink*. Universidad Católica Santiago de Guayaquil, Ecuador. Recuperado a partir de <http://repositorio.ucsg.edu.ec/bitstream/3317/6407/1/T-UCSG-PRE-TEC-ITEL-150.pdf>

Stamps, D. (2015). *Learn LabVIEW 2013/2014 fast: a primer for automatic data acquisition*. Mission, Kan.: SDC Publ.

Velasco Ximena. (2017). *Métodos de investigación científica*. Recuperado a partir de [http://www.ecotec.edu.ec/documentacion%5Cinvestigaciones%5Cdocentes\\_y\\_directivos%5Carticulos/4937\\_Fcevallos\\_00004.pdf](http://www.ecotec.edu.ec/documentacion%5Cinvestigaciones%5Cdocentes_y_directivos%5Carticulos/4937_Fcevallos_00004.pdf)



Presidencia  
de la República  
del Ecuador



Plan Nacional  
de Ciencia, Tecnología,  
Innovación y Saberes



SENESCYT  
Secretaría Nacional de Educación Superior,  
Ciencia, Tecnología e Innovación

## DECLARACIÓN Y AUTORIZACIÓN

Yo, **Escalante Bacilio, Krissy Stefany** con C.C: # 0950632018 autor del Trabajo de Titulación: **Instrumentación virtual utilizando LabView para la asignatura de señales y sistemas en la carrera de ingeniería en telecomunicaciones** previo a la obtención del título de **INGENIERO EN TELECOMUNICACIONES** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 13 de Marzo de 2017

---

Nombre: Escalante Bacilio, Krissy Stefany

C.C: 0950632018



Presidencia  
de la República  
del Ecuador



Plan Nacional  
de Ciencia, Tecnología,  
Innovación y Saberes



SENESCYT  
Secretaría Nacional de Educación Superior,  
Ciencia, Tecnología e Innovación

## REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA

### FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN

<b>TÍTULO Y SUBTÍTULO:</b>	INSTRUMENTACIÓN VIRTUAL UTILIZANDO LABVIEW PARA LA ASIGNATURA DE SEÑALES Y SISTEMAS EN LA CARRERA DE INGENIERÍA EN TELECOMUNICACIONES		
<b>AUTOR(ES)</b>	ESCALANTE BACILIO, KRISSEY STEFANY		
<b>REVISOR(ES)/TUTOR(ES)</b>	M. Sc. NÉSTOR A. ZAMORA CEDEÑO		
<b>INSTITUCIÓN:</b>	Universidad Católica de Santiago de Guayaquil		
<b>FACULTAD:</b>	Facultad de Educación Técnica para el Desarrollo		
<b>CARRERA:</b>	Ingeniería en Telecomunicaciones		
<b>TÍTULO OBTENIDO:</b>	Ingeniero en Telecomunicaciones		
<b>FECHA DE PUBLICACIÓN:</b>	13 de Marzo de 2017	<b>No. DE PÁGINAS:</b>	87
<b>ÁREAS TEMÁTICAS:</b>	Análisis de Señales y Sistemas		
<b>PALABRAS CLAVES/ KEYWORDS:</b>	Señales y sistemas, LabView, procesamiento de señales, teorema de Parseval, Modulación FM, transformada de Fourier		
<b>RESUMEN/ABSTRACT</b> (150-250 palabras):	<p>El presente trabajo de titulación tiene como propósito dar solución a la complejidad que puede tener la materia de señales y sistemas para los alumnos de la facultad técnica de la Universidad Católica Santiago de Guayaquil, mediante el uso de un software que es ampliamente utilizado en áreas de ingeniería, se busca hacer que la asignatura se torne más sencilla, con la ayuda de este software se invita a interactuar, realizar simulaciones y tener una más amplia idea de donde aplicar conocimientos dictados en clase. Existen varias herramientas para simular señales y procesamiento de las señales como lo son MatLab/Simulink y LabView, ambas son herramientas útiles para el diseño de sistemas, en este trabajo de titulación la plataforma escogida fue LabView que dispone de diversos instrumentos virtuales. Mediante la fundamentación teórica se incentiva al lector a conocer principios básicos de señales y sistemas, se ha tomado como ejemplo el teorema de Parseval, la modulación FM y transformada de Fourier para mostrar al lector la manera en que LabView puede graficar tanto nociones básicas como algunas más estructuradas.</p>		
<b>ADJUNTO PDF:</b>	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO	
<b>CONTACTO CON AUTOR/ES:</b>	<b>Teléfono:</b> +593-9-81596369	<b>E-mail:</b> <a href="mailto:stefanny_escalante@hotmail.com">stefanny_escalante@hotmail.com</a>	
<b>CONTACTO CON LA INSTITUCIÓN: COORDINADOR DEL PROCESO DE UTE</b>	<b>Nombre:</b> Palacios Meléndez Edwin Fernando		
	<b>Teléfono:</b> +593-9-68366762		
	<b>E-mail:</b> <a href="mailto:edwin.palacios@cu.ucsg.edu.ec">edwin.palacios@cu.ucsg.edu.ec</a>		
<b>SECCIÓN PARA USO DE BIBLIOTECA</b>			
<b>Nº. DE REGISTRO (en base a datos):</b>			
<b>Nº. DE CLASIFICACIÓN:</b>			
<b>DIRECCIÓN URL (tesis en la web):</b>			