



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERIA ELÉCTRICA
DEPARTAMENTO DE SISTEMA Y AUTOMÁTICA
PROYECTO DE TRABAJO ESPECIAL DE GRADO



**LABORATORIO VIRTUAL DE SISTEMAS
DE CONTROL HACIENDO USO DE
SOFTWARE GRÁFICO.**

Tutor:
Prof. Liliana Villavicencio.

Alejandro José Morales Díaz
V- 17.024.124.
Nelson Javier Vivas Nieto
V- 18.142.656.

Naguanagua, Julio de 2012.



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERIA ELÉCTRICA
DEPARTAMENTO DE SISTEMA Y AUTOMÁTICA
PROYECTO DE TRABAJO ESPECIAL DE GRADO



**LABORATORIO VIRTUAL DE SISTEMAS
DE CONTROL HACIENDO USO DE
SOFTWARE GRÁFICO.**

**Trabajo Especial de Grado presentado ante la Ilustre Universidad de Carabobo para
optar al título de Ingeniero Electricista.**

Tutor:
Prof. Liliana Villavicencio.

Alejandro José Morales Díaz
V- 17.024.124.
Nelson Javier Vivas Nieto
V- 18.142.656.

Naguanagua, Julio de 2012.



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERIA ELÉCTRICA
DEPARTAMENTO DE SISTEMA Y AUTOMÁTICA
PROYECTO DE TRABAJO ESPECIAL DE GRADO



CERTIFICADO DE APROBACIÓN

Los abajo firmantes miembro del jurado asignado para evaluar el trabajo especial de grado titulado “**LABORATORIO VIRTUAL DE SISTEMAS DE CONTROL HACIENDO USO DE SOFTWARE GRÁFICO**”, realizado por los bachilleres **Morales D. Alejandro J, C. I.: 17.024.124** y **Vivas N. Nelson J, C. I.: 18.142.656**, hacemos constar que hemos revisado y aprobado dicho trabajo.

Prof. Liliana Villavicencio
TUTORA

Prof. Teddy Rojas
JURADO

Prof. Ander Miranda
JURADO

Naguanagua, Julio de 2.012



ANEXOS

Capacitive Type

Capacitive cylindrical type proximity sensor

■ Features

- Able to detect Iron, metal, plastic, water, stone, wood etc
- Long life cycle and High reliability
- Convenient to adjust the detecting distance by sensitivity adjustment potentiometer
- Applications to control level and position



Please read "Caution for your safety" in operation manual before using.

■ Type

○DC 3-wire type

Appearances	Model
	CR18-8DN
	CR18-8DP
	CR18-8DN2 *
	CR30-15DN
	CR30-15DP
	CR30-15DN2 *

○AC 2-wire type

Appearances	Model
	CR18-8AO
	CR18-8AC
	CR30-15AO
	CR30-15AC

▶ * Mark is optional.

■ Specifications

Mode □	CR18-8DN CR18-8DP CR18-8DN2	CR30-15DN CR30-15DP CR30-15DN2	CR18-8AO CR18-8AC	CR30-15AO CR30-15AC
Detecting distance	8mm ±10%	15mm ±10%	8mm ±10%	15mm ±10%
Hysteresis	Max. 20% of detecting distance			
Standard detecting target	50×50×1mm (Iron)			
Setting distance	0 - 5.6mm	0 - 10.5mm	0-5.6mm	0-10.5mm
Power supply (Operating voltage)	12-24VDC (1.0-30VDC)		100-240VAC (85-264VAC)	
Current consumption	Max. 15mA		—	
Leakage current	Max. 2.2mA			
Response frequency	50Hz		20Hz	
Residual voltage	Max. 1.5V		Max. 20V	
Affection by Temp.	±20% Max. of detecting distance at +20℃ within temperature range of -25 - +70℃			
Control output	200mA		5-200mA	
Insulation resistance	Min. 50MΩ (at 500VDC)			
Dielectric strength	1500VAC 50/60Hz for 1 minute			
Vibration	1mm amplitude at frequency of 10 - 55Hz in each of X, Y, Z directions for 2 hours			
Shock	500m/s ² (50G) X, Y, Z directions for 3 times			
Indicator □	Operation indicator (Red LED)			
Ambient temperature	-25 - +70℃ (at non-freezing status)			
Storage temperature	-30 - +80℃ (at non-freezing status)			
Ambient humidity	35 - 95%RH			
Protection circuit	Surge protection circuit, Reverse polarity protection		Surge protection circuit built-in	
Protection □	IP66 (IEC specification)	IP65 (IEC specification)	IP66 (IEC specification)	IP65 (IEC specification)
Weight	Approx. 72g	Approx. 212g	Approx. 63g	Approx. 220g



Precision Thermocouple Amplifiers with Cold Junction Compensation

AD8494/AD8495/AD8496/AD8497

FEATURES

- Low cost and easy to use
- Pretrimmed for J or K type thermocouples
- Internal cold junction compensation
- High impedance differential input
- Standalone 5 mV/°C thermometer
- Reference pin allows offset adjustment
- Thermocouple break detection
- Laser wafer trimmed to 1°C initial accuracy and 0.025°C/°C ambient temperature rejection
- Low power: <1 mW at $V_S = 5\text{ V}$
- Wide power supply range
 - Single supply: 2.7 V to 36 V
 - Dual supply: $\pm 2.7\text{ V}$ to $\pm 18\text{ V}$
- Small, 8-lead MSOP

APPLICATIONS

- J or K type thermocouple temperature measurement
- Setpoint controller
- Celsius thermometer
- Universal cold junction compensator
- White goods (oven, stove top) temperature measurements
- Exhaust gas temperature sensing
- Catalytic converter temperature sensing

GENERAL DESCRIPTION

The AD8494/AD8495/AD8496/AD8497 are precision instrumentation amplifiers with thermocouple cold junction compensators on an integrated circuit. They produce a high level (5 mV/°C) output directly from a thermocouple signal by combining an ice point reference with a precalibrated amplifier. They can be used as standalone thermometers or as switched output setpoint controllers using either a fixed or remote setpoint control.

The AD8494/AD8495/AD8496/AD8497 can be powered from a single-ended supply (less than 3 V) and can measure temperatures below 0°C by offsetting the reference input. To minimize self-heating, an unloaded AD849x typically operates with a total supply current of 180 μA , but it is also capable of delivering in excess of $\pm 5\text{ mA}$ to a load.

The AD8494 and AD8496 are precalibrated by laser wafer trimming to match the characteristics of J type (iron-constantan) thermocouples; the AD8495 and AD8497 are laser trimmed to match the characteristics of K type (chromel-alumel) thermocouples. See Table 1 for the optimized ambient temperature range of each part.

FUNCTIONAL BLOCK DIAGRAM

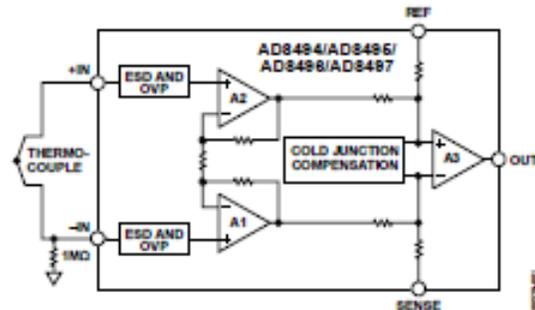


Figure 1.

Table 1. Device Temperature Ranges

Part No.	Thermo-Couple Type	Optimized Temperature Range	
		Ambient Temperature (Reference Junction)	Measurement Junction
AD8494	J	0°C to 50°C	Full J type range
AD8495	K	0°C to 50°C	Full K type range
AD8496	J	25°C to 100°C	Full J type range
AD8497	K	25°C to 100°C	Full K type range

The AD8494/AD8495/AD8496/AD8497 allow a wide variety of supply voltages. With a 5 V single supply, the 5 mV/°C output allows the devices to cover nearly 1000 degrees of a thermocouple's temperature range.

The AD8494/AD8495/AD8496/AD8497 work with 3 V supplies, allowing them to interface directly to lower supply ADCs. They can also work with supplies as large as 36 V in industrial systems that require a wide common-mode input range.

PRODUCT HIGHLIGHTS

1. Complete, precision laser wafer trimmed thermocouple signal conditioning system in a single IC package.
2. Flexible pinout provides for operation as a setpoint controller or as a standalone Celsius thermometer.
3. Rugged inputs withstand 4 kV ESD and provide over-voltage protection (OVP) up to $V_S \pm 25\text{ V}$.
4. Differential inputs reject common-mode noise on the thermocouple leads.
5. Reference pin voltage can be offset to measure 0°C on single supplies.
6. Available in a small, 8-lead MSOP that is fully RoHS compliant.

AD8494/AD8495/AD8496/AD8497

SETPOINT CONTROLLER

The AD849x can be used as a temperature setpoint controller, with a thermocouple input from a remote location or with the AD849x itself being used as a temperature sensor. When the measured temperature is below the setpoint temperature, the output voltage goes to $-V_S$. When the measured temperature is above the setpoint temperature, the output voltage goes to $+V_S$. For best accuracy and CMRR performance, the setpoint voltage should be created with a low impedance source. If the setpoint voltage is generated with a voltage divider, a buffer is recommended.

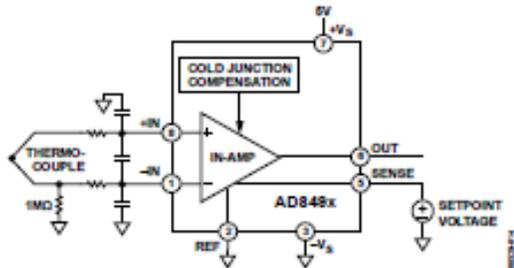


Figure 35. Setpoint Controller

Hysteresis can be added to the setpoint controller by using a resistor divider from the output to the reference pin, as shown in Figure 36. The hysteresis in $^{\circ}\text{C}$ is

$$T_{\text{HYS}} = \frac{V_S \times R1 / (R1 + R2)}{5 \text{ mV}/^{\circ}\text{C}}$$

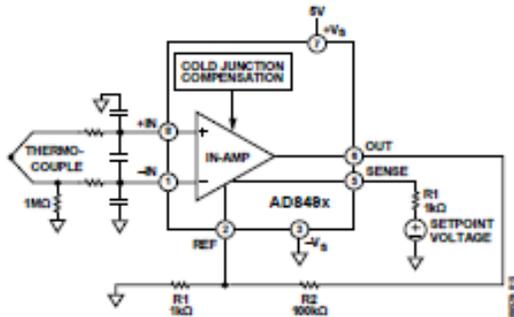


Figure 36. Adding 10 Degrees of Hysteresis

A resistor equivalent to the output resistance of the divider should be connected to the sense pin to ensure good CMRR.

MEASURING NEGATIVE TEMPERATURES

The AD849x can measure negative temperatures on dual supplies and on a single supply. When operating on dual supplies with the reference pin grounded, a negative output voltage indicates a negative temperature at the thermocouple measurement junction.

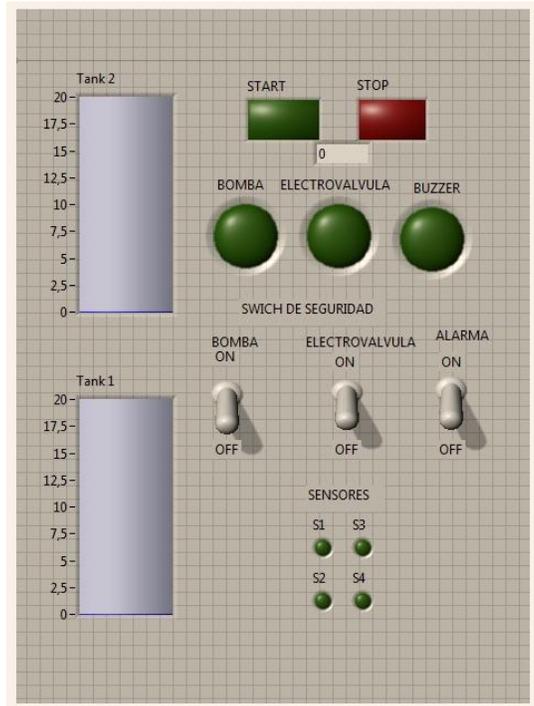
$$V_{\text{OUT}} = (T_M \times 5 \text{ mV}/^{\circ}\text{C}) + V_{\text{REF}}$$

When operating the AD849x on a single supply, level-shift the output by applying a positive voltage (less than $+V_S$) on the reference pin. An output voltage less than V_{REF} indicates a negative temperature at the thermocouple measurement junction.

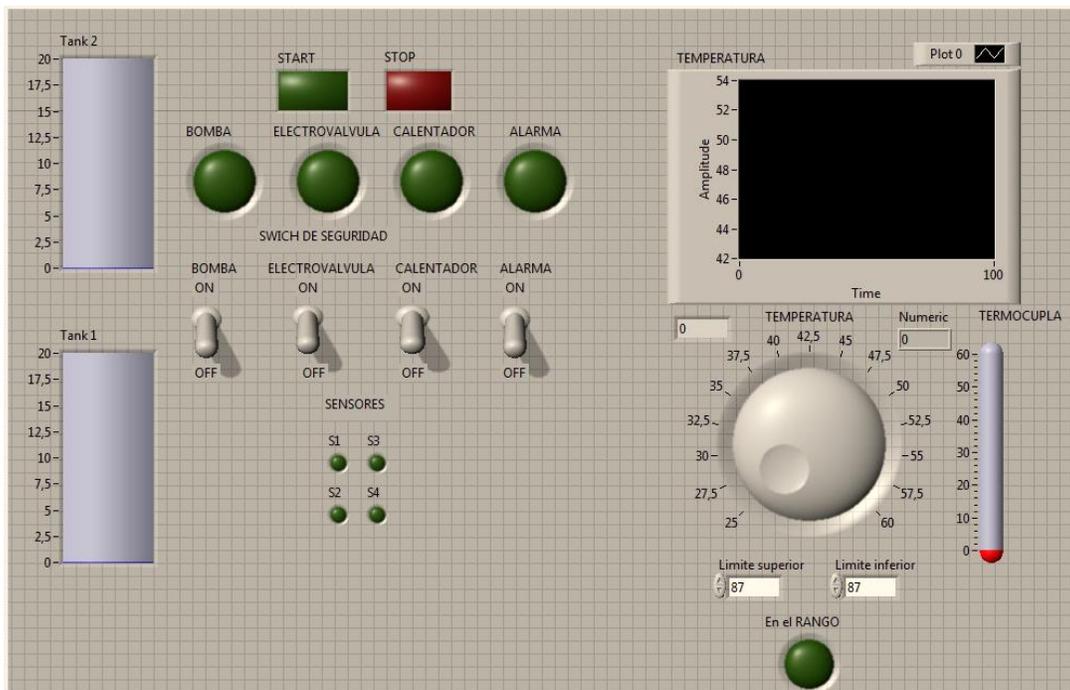
REFERENCE PIN ALLOWS OFFSET ADJUSTMENT

The reference pin can be used to level-shift the AD849x output voltage. This is useful for measuring negative temperatures on a single supply and to match the AD849x output voltage range to the input voltage range of the subsequent electronics in the signal chain.

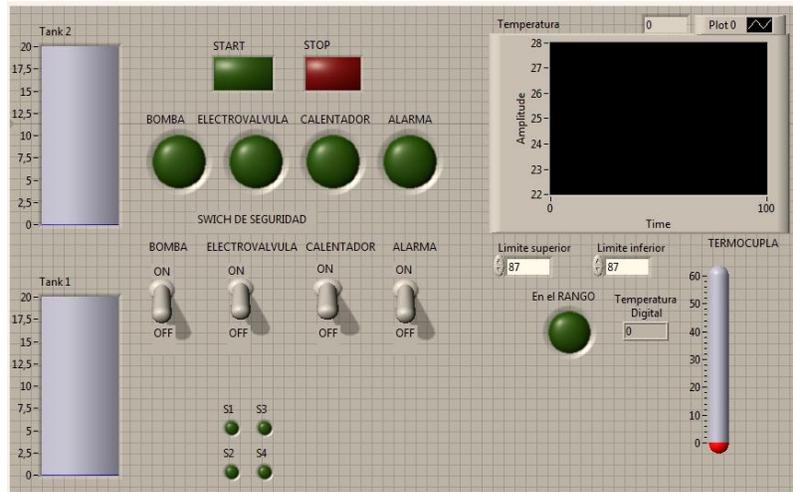
The reference pin can also be used to offset any initial calibration errors. Apply a small reference voltage proportional to the error to nullify the effect of the calibration error on the output.



Panel de Control de Nivel del Banco de Control de Procesos



Panel de control de temperatura por medio de control manual



Panel de control de temperatura por medio de ecuaciones simuladas



Dedicatoria

Este trabajo especial de grado va dedicado a mis padres, Amada Esther Díaz de Morales quien siempre me ha dado su apoyo incondicional en mis estudios, me ha enseñado buenos valores y principios para ir por el camino correcto y me ha dado todo su amor de madre, y Alejandro Raúl Morales Loyola (+) quien siempre me apoyó durante su estadía con nosotros, quien fue un buen padre ya que siempre me enseñó a andar por el buen camino en el trabajo y los estudios, a ser un buen hijo y a luchar siempre por lo que siempre he querido. Gracias a mi hermana Amalirey Joselin Morales Díaz por su apoyo y ayuda durante todo mi recorrido de vida como estudiante y hermano, a mis familiares, amigos, por sus apoyos y siempre mantenerme presente en lograr mi meta como profesional.

Alejandro J. Morales D.



Dedicatoria

El amor de la familia y más de una madre es solo comparable al amor que siente dios por sus hijos, por ello infinito respeto, admiración y reverencia a mi madre que con su tenacidad, constancia y a pesar de todo estar lo suficientemente mal, logra apoyarme cuidarme y mantenerme activo en lo que quiero. Por eso y más te amo madre con cada átomo de mi cuerpo físico, gracias por tu apoyo incondicional.

A mi padre que se esforzó por guiarme en el buen camino de la rectitud y las buenas costumbre, mis hermanos y amigos que estuvieron allí brindándome apoyo durante toda la carrera evitando que decayera en el intento. Muchas gracias a todos.

Nelson J. Vivas N.



Agradecimientos

El desarrollo de este trabajo especial de grado no habría sido posible sin el apoyo moral y económico de mi madre, a ella le debo todo el valor, la motivación y el cariño puesto en él.

Igualmente debo agradecer de manera especial:

- A mi hermana Amalirey Morales, y mi novia Marielvis Martínez, porque siempre estuvieron pendiente de cómo iba el desarrollo de este trabajo y por su gran apoyo.
- A los familiares de mi madre como tíos, tías, primos, primas, porque siempre me apoyaron en mis estudios y mantenerme siempre estable en ellos y a no desmallar.
- A los padres de mi novia, José Martínez y Melvis Castillo por su apoyo durante parte de la carrera, y por sus consejos en lo mismo.
- A Nelson Vivas, compañero de defensa de este trabajo porque siempre estuvimos en buena comunicación durante el desarrollo de este trabajo y el buen equipo que formamos en el desarrollo del mismo debido a que vivimos muy distantes.
- A mis amigos y ex - compañeros de estudios: Luis Rodríguez, Cesar Picón y otros por sus aportes que fueron claves para cumplir con algunas etapas de este proyecto.
- A mi tutora y amiga, Prof. Liliana Villavicencio por permitirnos realizar este proyecto, orientarnos y por sus aportes y conocimientos durante el desarrollo.
- A los profesores y amigos, Aída Pérez, Oriana Barrios, Frederick por orientarnos en parte del desarrollo de este proyecto.
- A todas aquellas personas, familiares, amigos y otros, que de alguna manera influyeron en la realización de este trabajo, aunque no los mencione, les estoy muy agradecidos por su aporte.

Y le agradezco a Dios, porque sé que a través de él fui conociendo a todas estas personas, y de todas las experiencias vividas en este tiempo. El es el gran responsable.

Alejandro J. Morales D.



Agradecimientos

El desarrollo de este proyecto sin el apoyo moral de mis padres, debo a ellos toda la dedicación y valor que hicieron posible mantenerme en pie durante mis estudios.

Igualmente debo agradecer de manera especial:

- Mis hermanos, por su apoyo incondicional, y aporte a la resolución de problemas
- Mis amigos y ex compañeros de estudios que estuvieron siempre allí apoyándome para culminar.
- A mi tutora y amiga Liliana Villavicencio, quien nos dió una ventana nueva de conocimiento que para mí es invaluable, así como también guiarnos con sus conocimientos de la manera más amable posible
- Y por último y no menos importante a mi compañero de tesis Alejandro Morales, el cual fue elemento necesario para lograr la meta, su apoyo y compromiso fueron de gran importancia, muchas gracias amigo mío.

Agradezco también a dios, que aunque sé que las personas llegan a nosotros por afinidad debo sentirme orgulloso de cada uno de ustedes y seguiré aprendiendo en todo lo que puedan brindarme.

Nelson J. Vivas N.



ÍNDICE GENERAL

Portada.....	i
Página de Título.....	ii
Certificado de Aprobación.....	iii
Dedicatorias.....	iv
Agradecimientos.....	vi
Índice General.....	viii
Índice de Imágenes.....	xi
Índice de Tablas.....	xv
Introducción.....	xvi
Capítulo I. El Problema.....	1
1.1. Planteamiento del Problema.....	1
1.2. Justificación.....	2
1.3. Objetivos.....	4
1.3.1. Objetivo General.....	4
1.3.2. Objetivo Específico.....	4
1.4. Alcance.....	5
Capítulo II. Marco Teórico.....	6
2.1. Antecedentes.....	6
2.2. Bases Teóricas.....	8
2.2.1. Control y Automatización.....	9
2.2.1.1. Ventajas.....	9
2.2.1.2. Desventajas.....	9
2.2.1.3. Partes de un Sistema Automatizado.....	10
2.2.2. Tarjeta de Adquisición de Datos y Tarjeta de Acople.....	11
2.2.2.1. Proceso de Adquisición de Datos.....	14
2.2.2.2. ¿Cómo se adquieren los Datos?.....	16
2.2.2.3. Tiempo de Conversión.....	18
2.2.2.4. La etapa de Accionamiento de la Señal.....	18
2.2.2.5. Ventajas de la Tarjeta de Adquisición de Datos.....	20



2.2.2.6.	Tarjeta de Adquisición de Datos USB – 6008.....	20
2.2.3.	¿Qué es LabVIEW®?.....	26
2.2.3.1.	Principales Usos.....	27
2.2.3.2.	Principales Características.....	28
2.2.3.3.	Ventajas de LabVIEW®.....	30
2.2.3.4.	¿Cómo trabaja LabVIEW®.....	32
2.2.3.5.	Programación en LabVIEW®.....	43
2.2.3.6.	Ejecución en un VI.....	44
2.2.3.7.	Tipos de Datos.....	48
2.2.3.8.	Sub rutina y SubVI.....	53
2.2.3.9.	Estructuras de Control.....	55
2.2.3.10.	Registro de Desplazamiento.....	65
2.2.3.11.	Arreglos y Clúster.....	67
2.2.3.12.	Indicadores Gráficos: Graph y Charts.....	70
2.2.3.13.	Archivos I/O.....	75
2.2.3.14.	Variables Locales y Globales.....	79
2.2.3.15.	Ejemplo: Construcción de un VI.....	84
2.2.3.16.	Programa de Ejemplo: Mezclador de Pintura.....	87
Capítulo III. Marco Metodológico.....		89
3.1.	Tipo de Investigación.....	89
3.2.	Diseño de la Investigación.....	89
3.3.	Procedimiento Metodológico.....	90
Capítulo IV. Desarrollo del Banco de Control de Procesos y Resultados.....		92
4.1.	Introducción.....	92
4.2.	Fases de Desarrollo.....	92
4.2.1.	Fase N° 1: Estudio, adiestramiento y manejo del software gráfico.....	92
4.2.2.	Fase N° 2: Selección de la Tarjeta de Adquisición de Datos y Tarjeta de Acople.....	94
4.2.3.	Fase N° 3: Diseño del Banco de Control de Procesos.....	96
4.2.3.1.	Diseño de la Estructura de Soporte.....	100
4.2.3.2.	Diseño de Tanques.....	103
4.2.3.3.	Elaboración e instalación del sistema de tuberías para los tanques.....	105
4.2.3.4.	Elaboración del sistema de sensores de nivel.....	109
4.2.3.5.	Elaboración de interfaz del tablero frontal.....	112
4.2.3.6.	Cableado del Banco de Control de Procesos.....	113
4.2.4.	Fase N° 4: Programación en el software gráfico adaptado al banco de control.....	119
4.2.5.	Fase N° 5: Verificación del sistema y software previamente Programado.....	124



4.2.6.	Fase N° 6: Puesta en marcha del banco de control.....	124
4.2.7.	Fase N° 7: Toma de Datos y Análisis.....	124
4.2.8.	Fase N° 8: Elaboración de compendio de prácticas.....	124
Capítulo V. Desarrollo de Prácticas de Laboratorio.....		126
5.1.	Introducción.....	126
5.2.	Práctica de Laboratorio N° 1.....	127
5.3.	Práctica de Laboratorio N° 2.....	129
5.4.	Práctica de Laboratorio N° 3.....	132
Conclusiones.....		136
Recomendaciones.....		139
Bibliografía.....		141
Apéndice A. Manual de Usuario.....		143
Apéndice B. Diagrama de las diferentes conexiones del Banco de Procesos.....		156
Apéndice C: Galería de imágenes del Banco de Control de Procesos.....		160
Apéndice D: Especificaciones de la DAQ NI USB 9211.....		165
Apéndice E: Manual para comunicación OPC SERVE con LabVIEW®.....		169
Apéndice F: Costo de elaboración de Banco de Control de Procesos.....		187
Anexos.....		190
Anexo 1.	Especificaciones de sensores capacitivos CR18 – 8DN.....	191
Anexo 2.	Especificaciones de dispositivo analógico AD8495 = AD595 (Parte 1).....	192
Anexo 3.	Especificaciones de dispositivo analógico AD8495 = AD595 (Parte 2).....	193
Anexo 4.	Panel de control de nivel.....	194
Anexo 5.	Panel de control de temperatura por control manual.....	194
Anexo 6.	Panel de control de temperatura por control de ecuación.....	195



ÍNDICE DE IMAGENES

Imagen 2.1.a. Diagrama general de la adquisición de datos.....	12
Imagen 2.1.b. Modelos de DAQ.....	12
Imagen 2.2. Tarjeta de Acople.....	13
Imagen 2.3. Dispositivo DAQ USB 6008.....	14
Imagen 2.4. Especificaciones y Medidas de la Tarjeta de Adquisición a usar.....	21
Imagen 2.5. Componentes funcionales.....	22
Imagen 2.6. Etiquetas de la DAQ.....	23
Imagen 2.7. Modo de Insertar los porta-pines de la DAQ.....	23
Imagen 2.8. Generación de Señales y Procesamiento.....	29
Imagen 2.9. Generación de Señal y Proceso. (Modo Gráfico).....	30
Imagen 2.10. Ventana Inicial al abrir LabVIEW®.....	32
Imagen 2.11. Ventanas de Panel Frontal y de Bloqueo.....	33
Imagen 2.12. Botones Perillas y otros dispositivos de entrada.....	34
Imagen 2.13. Indicadores Leds y de nivel.....	34
Imagen 2.14. Paletas de Control y Herramientas.....	35
Imagen 2.15. Paletas de Herramientas.....	36
Imagen 2.16. Panel Frontal de controles.....	38
Imagen 2.17. Paleta de Funciones.....	40
Imagen 2.18. Paletas de Funciones.....	41
Imagen 2.19. Ejecución de un VI.....	44
Imagen 2.20. Ejecutando el VI.....	45
Imagen 2.21. Ejecuciones continuas de un VI.....	45
Imagen 2.22. Botones de pausa y parada de un VI.....	46
Imagen 2.23. Panel Control. Compuertas Lógicas.....	46
Imagen 2.24. Diagrama de Bloques. Compuertas Lógicas.....	47
Imagen 2.25. Esquema de Tipos de Datos.....	48
Imagen 2.26. Controles e indicadores booleanos.....	48



Imagen 2.27. Diferentes representaciones de Datos Numéricos.....	50
Imagen 2.28. Arreglos n dimensionales.....	50
Imagen 2.29. Datos alfanuméricos.....	51
Imagen 2.30. Suma de dos números.....	52
Imagen 2.31. Concatenados de datos numéricos y alfanuméricos.....	52
Imagen 2.32. Panel de Conectores.....	54
Imagen 2.33. Terminal asignado Icono SC.....	54
Imagen 2.34. Edit Icono SC.....	55
Imagen 2.35. Selección de VI.....	55
Imagen 2.36. Estructuras de un VI.....	57
Imagen 2.37. Panel Frontal de un Control Numérico.....	64
Imagen 2.38. Ejemplo práctico nodo formula (diagrama de bloques).....	65
Imagen 2.39. Almacenamiento de Datos.....	66
Imagen 2.40. Ejemplo de un Panel Frontal.....	67
Imagen 2.41. Diagrama de Bloques (Ejemplo).....	67
Imagen 2.42. Arreglos de diagrama de bloques.....	69
Imagen 2.43. Cluster de diagrama de bloque.....	70
Imagen 2.44. Diagrama de bloques.....	71
Imagen 2.45. Diagrama de medición.....	72
Imagen 2.46. Muestra de una Grafica.....	72
Imagen 2.47. Digital Waveform Graph (panel frontal).....	73
Imagen 2.48. Gráfico Digital Waveform Graph (diagrama de bloques).....	73
Imagen 2.49. Grafica en plano XY.....	74
Imagen 2.50. Gráfica en 3D.....	74
Imagen 2.51. Ejemplo práctico con Waveform Graph.....	76
Imagen 2.52. Generación de números aleatorios.....	76
Imagen 2.53. Señal de una Onda.....	77



Imagen 2.54. Escritura de un archivo.....	77
Imagen 2.55. Resultado.....	78
Imagen 2.56. VI finalizado.....	79
Imagen 2.57. Pulsadores.....	81
Imagen 2.58. Acción Mecánica.....	81
Imagen 2.59. Diagrama de bloques.....	82
Imagen 2.60. Indicador Booleano.....	83
Imagen 2.61. Ejemplo con Booleanos.....	84
Imagen 2.62. Mezclador de Pintura.....	88
Imagen 3.1. Diagrama de comunicación Tarjeta – PC.....	91
Imagen 4.1. Esquema Básico del banco de Control de Procesos.....	92
Imagen 4.2. Vista principal del Software.....	93
Imagen 4.3. Panel frontal para la programación del Software.....	94
Imagen 4.4.1. Imágenes de la Tarjeta de Adquisición de Datos.....	95
Imagen 4.4.2. Tarjeta de Acople.....	96
Imagen 4.5. Diseño propuestos para los tanques del sistema.....	98
Imagen 4.6. Colocación de los tanques en el sistema.....	98
Imagen 4.7. Esquema de sistema de tuberías y sensores para el sistema de tanques.....	99
Imagen 4.8. Esquema de medidas del banco de proceso y ubicación de los tanques.....	101
Imagen 4.9. Estructura resultante.....	102
Imagen 4.10. Estructura finalizada.....	103
Imagen 4.11. Tanque elaborado y mesa para soporte.....	105
Imagen 4.12. Válvula Solenoide y Bomba.....	105
Imagen 4.13. Conexión de la bomba y válvula solenoide.....	107
Imagen 4.14. Imágenes de la instalación del sistema de tuberías del proceso.....	109
Imagen 4.15. Sensores capacitivos usados.....	110
Imagen 4.16. Imágenes para la elaboración del sistema de sensores.....	111



Imagen 4.17. Interfaz del tablero frontal.....	112
Imagen 4.18. Cableado del Banco de Control.....	114
Imagen 4.19. Tarjeta diseñada para conectar con la DAQ.....	115
Imagen 4.20. Circuito de la Tarjeta diseñada.....	115
Imagen 4.21. Optoacoplador para la tarjeta diseñada.....	116
Imagen 4.22. Puente Darlington.....	116
Imagen 4.23. Imágenes del calentador.....	119
Imagen 4.24. Representación del sistema de Temperatura.....	120
Imagen 4.25. Diagrama de flujo de proceso de temperatura.....	121
Imagen 4.26. Estructura FLIP-FLOP.....	122
Imagen 4.27. Estructura Timer TON.....	122
Imagen 4.28. Estructura Timer TOFF.....	122
Imagen 4.29. Programación del VI en LabVIEW®.....	123
Imagen 4.30. Programación gráfica del proceso de temperatura. (Panel de Trabajo).....	123
Imagen 5.1. Banco de Control de Procesos (Nivel).....	128
Imagen 5.2. Representación del sistema de vaciado y llenado de tanques.....	129
Imagen 5.3. Banco de Control de Procesos (Temperatura).....	131
Imagen 5.4. Representación del sistema de vaciado y llenado de tanques mediante variación de temperatura Manual.....	131
Imagen 5.5. Banco de Control de Procesos (Temperatura).....	134
Imagen 5.6. Representación del sistema de vaciado y llenado de tanques mediante variación de temperatura por medio de ecuaciones.....	135



ÍNDICE DE TABLAS

Tabla 2.1. Terminales Analógicos y Digitales.....	25
Tabla 2.2. Terminales Analógicos y Digitales.....	25
Tabla 2.3. Estados de Leds.....	24
Tabla 2.4. Estructuras de un VI.....	56
Tabla 4.1. Lista de materiales para la elaboración de la estructura de soporte.....	102
Tabla 4.2. Lista de materiales para la elaboración de los tanques.....	103
Tabla 4.3. Característica de la Válvula Solenoide.....	106
Tabla 4.4. Especificación del Producto.....	106
Tabla 4.5. Lista de materiales para las conexiones de tuberías.....	107
Tabla 4.6. Características técnicas de los sensores capacitivos.....	110
Tabla 4.7. Lista de materiales para la elaboración de interfaz del tablero.....	112
Tabla 4.8. Distribución de colores de cableado del tablero.....	116
Tabla 4.9. Nomenclatura de identificación del cableado del tablero.....	117
Tabla 4.10. Características del calentador.....	118
Tabla 4.11. Estudio de carga para la fuente de alimentación 12VDC.....	119



INTRODUCCIÓN

El presente trabajo de grado tiene como objeto incorporar software gráfico para la enseñanza de Sistemas de Control y Automatización con laboratorios virtuales en la Universidad de Carabobo. Dicho software representa un entorno de programación gráfico el cual usa sistemas de medidas, pruebas y control por medio de iconos, gráficos y cables, intuitivos, semejante a un diagrama de flujo. Ofreciendo una gran integración en cuanto a dispositivos de hardware para análisis y visualización de datos.

Por otra parte el uso de éste permite la apertura de manipulación hacia áreas concernientes al campo de control industrial como lo es la robótica, instrumentación de procesos entre otras, dando al estudiante la oportunidad de conocer nuevos sistemas, formando así ingenieros integrales con el valor agregado de conocer e introducir tecnología de punta a la universidad.

Estudiamos los aspectos fundamentales del software así como la transmisión de información al banco de proceso con el entorno gráfico mediante la recopilación y revisión de diversos manuales, tutoriales y fuentes bibliográficas, a fin de facilitar su implementación.

El objetivo de este trabajo de grado es la creación del Laboratorio virtual de control mediante un entorno gráfico con el resto de los componentes eléctricos y mecánicos necesarios, para que los estudiantes de la Universidad de Carabobo se familiaricen con las herramientas para su posterior desempeño industrial.

El presente trabajo de investigación está estructurado en cinco (5) capítulos, descritos brevemente a continuación:

- En el capítulo I, se realiza el planteamiento del problema y se mencionan las razones que justifica el desarrollo del Banco de Control de Procesos para el Laboratorio de Control, además de establecer los objetivos generales y específicos que se cumplieron durante la realización del proyecto.



- En el capítulo II, se mencionan antecedentes ligados al proyecto, se explican fundamentos teóricos relaciones con Sistemas de Control, el estudio de todo lo referente a la Tarjeta de Adquisición de Datos y su software de programación como lo es LabVIEW, y otros elementos de interés que pueden facilitar el entendimiento del proyecto.
- En el capítulo III, se describe brevemente la metodología de investigación adoptada para el presente trabajo de investigación, mencionando las distintas fases de desarrollo que llevaron a cabo del cumplimiento de los objetivos.
- En el capítulo IV, se describe en detalle el cumplimiento de cada una de las fases de desarrollo y el resultado de la puesta en marcha del Banco de Control de Procesos.
- En el capítulo V, se proponen prácticas de laboratorio diseñadas especialmente para el Banco de Control de Procesos.

Al final, se presentan las conclusiones y recomendaciones extraídas de la realización y puesta en marcha del Banco de Control de Procesos, y se anexan documentos que pueden ser de interés para el lector.



CAPÍTULO I

1.1. PLANTEAMIENTO DEL PROBLEMA

En la Facultad de Ingeniería / Escuela de Ingeniería Eléctrica en el departamento de Sistemas y Automática dentro del campus de la Universidad de Carabobo, se dictan materias tales como Teoría de Control I perteneciente al sexto semestre del pensum de estudios, Teoría de Control II al séptimo semestre, Teoría de Control III al octavo semestre, Automatización Industrial I al séptimo semestre y Automatización Industrial II al octavo semestre, materias las cuales encaminan al estudiante al conocimiento y comprensión de la naturaleza de los sistemas físicos de nuestro entorno, la manera en que se modela matemáticamente los mismos y la forma en que el humano ha sabido manipular e introducir variables de estados de control automático o manual a su conveniencia, además la forma de cómo esto se implementa en la industria en los procesos de automatización.

Por consiguiente para concretar y comprender estos conceptos teóricos ya aprendidos es de gran beneficio el uso de un laboratorio en el que se utilicen los conocimientos teóricos adquiridos y validarlos en un entorno práctico virtual; se habla de un software que incluye una herramienta gráfica para pruebas, control y diseño, que permite simular sistemas reales y modelarlos matemáticamente una vez analizado.

En la actualidad este tipo de estudios antes descrito solo se ha visualizado mediante ecuaciones que nos muestran su comportamiento, estado, y tipo de control también se ha llevado a cabo mediante simulaciones en otros entornos, esto en el campo de la automatización, logrando así un resultado bastante cercano a la realidad física estudiada.



1.2.JUSTIFICACIÓN

Día a día se incrementa la aplicación de sistemas de control en diferentes procesos, ya que de esta manera se aumenta la calidad de los bienes producidos así como también se disminuyen los costos de operación. Debido a esto cada vez se vuelven más complejas las áreas en las que se necesitan aplicar técnicas de sistemas de control, por este motivo el ingeniero electricista deberá estar altamente capacitado en esta área, habiendo experimentado el diseño de sistemas de control complejos que contengan múltiples variables, lo que es común en los sistemas industriales. De allí la necesidad de disponer de un laboratorio de sistemas de control, de equipos y dispositivos que les permitan a los futuros ingenieros realizar experimentos, en los que apliquen los conocimientos adquiridos, en condiciones muy similares a las que se presentan en un ambiente industrial real.

Con el desarrollo de este trabajo especial de grado, se logró que por medio del uso de Software gráfico se desarrollen maneras alternativas virtuales y de alta calidad para la emulación y simulación de sistemas físicos reales a los cuales se les pretende realizar un control de acuerdo a la necesidad.

La puesta en marcha de este diseño permite que el estudiante desarrolle habilidades prácticas de control en sistemas reales y complejos de múltiples variables, esto debido a que en él es imprescindible el uso de la teoría de control ya vista con anterioridad tal es el caso de los controladores PID que dentro del software se deben generar para así llevar a cabo el buen funcionamiento del diseño. Es preciso destacar que este diseño en cuestión consta de sensores, válvulas, tarjeta de adquisición de datos con su respectivo driver que permite la manipulación de la misma a nivel digital, bombas, alarma, agitador, entre otros, las cuales el estudiante tendrá que manipular para hacer efectivo el control del sistema de manera virtual.



Al llevar a cabo este proyecto que emplea tecnología de punta y de alta versatilidad, permitirá aprovechar de manera comprensible e intuitiva el uso de control por computadoras de sistemas físicos dando al estudiante herramientas las cuales amplían su visión de sistemas de control que los capacita para enfrentarse a los retos que hoy en día se presentan en la industria, además de ellos los conocimientos que contarán con la plataforma necesaria para el desarrollo futuro de nuevos sistemas de control. Por ello es menester el estudio de este software a nivel de pregrado con el fin de hacer uso y poner en práctica las capacidades y conocimientos adquiridos.



1.3.OBJETIVOS

1.3.1. OBJETIVO GENERAL:

Desarrollar el proyecto de implantación y componente didáctico (prácticas) de un laboratorio virtual de control usando Software Gráfico.

1.3.2. OBJETIVOS ESPECÍFICOS:

- Estudiar cómo transmitir información al banco de control de procesos de tanques y control de temperatura, con el entorno gráfico del software, mediante la recopilación y revisión de diversos manuales, tutoriales y fuentes bibliográficas, a fin de facilitar su implementación.
- Diseñar el control del banco de proceso y de temperatura mediante el entorno gráfico del software y el resto de los componentes eléctricos y mecánicos necesarios, cuyas dimensiones permitan su portabilidad en el laboratorio.
- Investigar el procedimiento para la conexión entre el software y el banco de control con la tarjeta de adquisición de datos.
- Seleccionar el sistema de tarjeta de adquisición de datos al banco de control de procesos de tanques y de temperatura para su implantación.
- Desarrollar prácticas de laboratorio que permitan simular el control de procesos en tanques y control de temperatura, donde también se puedan complementar los elementos del sistema instalado, sistemas electro – neumáticos del laboratorio, a fin de promover su utilidad en el Laboratorio virtual de Control.



1.4.ALCANCE

Se realizará la implantación del laboratorio virtual de control a través del software gráfico con una interfaz que permita una rápida conexión a la tarjeta de adquisición de datos para permitir al usuario el fácil conexionado de cualquier montaje, siempre que este cumpla con los requerimientos exigidos por el manual. Las variables involucradas en el control serán solo de tipo discreto. También se tendrá la posibilidad de desmontar fácilmente la tarjeta de adquisición de datos del Banco de Control, en caso de requerir su uso en otras aplicaciones.

Se implementará además sobre la interfaz externa del banco los elementos sensores y actuadores necesarios para presentar algún proceso de control automático sencillo, lo cual deberá ser programado y puesto en marcha por el usuario.



CAPÍTULO II

MARCO TEÓRICO

2.1 ANTECEDENTES

El presente trabajo especial de grado “Laboratorio Virtual de Sistemas De Control haciendo uso de Software Gráfico” tiene como investigación previa en cuanto a bases teóricas los trabajos especiales hecho en los últimos años, entre los cuales se menciona:

- **Hidalgo, Katherine (2008).**

“Desarrollo de un banco de procesos didáctico para el Laboratorio de Automatización Industrial I de la Escuela de Ingeniería Eléctrica de la Universidad de Carabobo”

Este proyecto presenta el desarrollo de un banco de procesos en cuyo tablero se simulan siete procesos industriales automatizados muy comunes. En el sistema que se elaboró, los procesos están representados gráficamente sobre láminas intercambiables que se superponen al tablero, lo cual permite reducir significativamente el tiempo de implementaciones dentro del laboratorio y también se incentiva el uso de los autómatas entre los estudiantes del Laboratorio de automatización industria I. La metodología empleada en este trabajo, para el uso del banco didáctico a fin de conseguir la implementación de labview para los alumnos dentro del laboratorio e incentivar nuevos métodos de control. [1]

- **Johan W. Janampa P. (2008)**

“Desarrollo de un Banco de Control de Procesos Portátil para el Laboratorio de Automatización Industrial I mediante un Micro PLC Simatic S7-200”

Este proyecto muestra la realización de un banco de procesos basado en el mezclado de líquidos de forma automatizado con la ayuda de un PLC Simatic S7-200. Este nos permite poner en marcha el proceso de automatizado del banco por medio de un sistema de control diferente, que ayuda a la interfaz entre el alumno y el banco como



lo es labview, además de la metodología a llevar a cabo para la implementación de la práctica. [2]

- **Rodrigo .González C, Rodrigo P. Pino**

“Análisis de software para desarrollo entorno gráfico LabVIEW® y propuesta de implementación para laboratorio en el instituto de electricidad y electrónica en universidad austral de chile” nos permite visualizar la manera de desarrollarse dentro del software LabVIEW® permitiendo la Investigación analítica del software herramienta de programación con el propósito de que las futuras generaciones lo utilicen con fines didácticos. [3]

- **Dra. Valentina, Ulver de Beluatti, Ing. Germán Pineda, Dr. Rodolfo Ávila, Dr. Hugo Juri**

“Aplicación de LabVIEW® como instrumento virtual en medición de la movilidad dental” Desarrollar una aplicación capaz de gobernar completamente el sistema de medición de la movilidad dental. Nos permitió verificar aplicaciones de implantación del software así como también maneras de conexión y adquisición de datos. [4]

- **Ing. María Paula Gómez Quintero, Ing. Gustavo Andrés Zabala Domínguez, Ing. Javier Andrés Dávila Rincón**

“Uso de LabVIEW® para sistemas de control en ingeniería química”, La construcción del modelo en Labview para la interfaz que se basa en la creación de un proyecto en el cual se integran todas las variables y los subprogramas que serán utilizados en el algoritmo principal. [5]



2.2. BASES TEÓRICAS:

La investigación que ayudará al desarrollo del proyecto son las siguientes:

- ✓ Control y Automatización.
- ✓ Ventajas.
- ✓ Desventajas.
- ✓ Partes de un sistema Automatizado.
- ✓ Tarjeta de Adquisición de datos y Tarjeta de Acople.
- ✓ Proceso de Adquisición de datos.
- ✓ Como se adquieren los datos.
- ✓ Tiempo de Conversión.
- ✓ La etapa de accionamiento de la señal.
- ✓ Ventajas de la tarjeta de adquisición de datos.
- ✓ Tarjeta de adquisición de datos USB 6008.
- ✓ ¿Qué es LabView?
- ✓ Principales usos.
- ✓ Principales Características.
- ✓ Ventajas de LabView.
- ✓ ¿Cómo trabaja LabView?
- ✓ Programación en LabView.
- ✓ Ejecución en un VI.
- ✓ Tipos de Datos.
- ✓ Sub rutina o SubVI.
- ✓ Estructuras de Control.
- ✓ Registros de desplazamientos.
- ✓ Arreglos y Clúster.
- ✓ Indicadores Gráficos.
- ✓ Archivos I/O.
- ✓ Variables Locales y Globales.



- ✓ Ejemplos de programación en un VI.
- ✓ Programa de Proyecto. “Mezclador de Productos”.

2.2.1. Control y Automatización.

El control del proceso consiste en la recepción de unas entradas, variables del proceso, su procesamiento y comparación con unos valores predeterminados por el usuario, y posterior corrección en caso de que se haya producido alguna desviación respecto al valor preestablecido de algún parámetro de proceso. La automatización consiste en emplear un sistema en donde la operación manual (realizada habitualmente por seres humanos) es sustituida por un conjunto de elementos tecnológicos a fin de mejorar las tareas de producción y simplificar las labores del operario.

2.2.1.1. Ventajas

- Aumento de la productividad.
- Reducción de costos laborales.
- Mejor calidad del producto.
- Reducción de costos por consumo de energía y mantenimiento.
- Aumento de la seguridad laboral.
- Integración de la operación y de la información para la recopilación de datos estadísticos del proceso.
- Monitoreo de los equipos y maquinas que intervienen en el proceso.
- Aumento en el rendimiento de los equipos.
- Disminución de contaminación y daños ambientales.

2.2.1.2. Desventajas

- Resistencia al cambio.



- Falta de conocimientos.
- Necesidad de capacitación personal.
- Inversión inicial elevada.
- Dependencia tecnológica.
- Obsolescencia.
- Escases de recurso humano capacitado.
- Limitación de maquinas frente al operador humano para censar ciertas variables: olor, sabor, acabado de piezas, etc.

2.2.1.3. Partes de un sistema Automatizado.

Un sistema automatizado consta de las siguientes partes:

- Interfaz Hombre – Máquina: es el medio por el cual el operador interactúa con los diferentes equipos que controlan el proceso. Esta interfaz, que puede ser una computadora personal (PC) o un tablero de control, permite al operador monitorear el estado del proceso.
- Controlador: este dispositivo genera las órdenes que obedecen los elementos actuadores del sistema en base a un conjunto de instrucciones preestablecidas y a un conjunto de señales provenientes del proceso. Para esta función se suelen emplear computadoras personales y sistemas electrónicos basados en microcontroladores, aunque hay otros equipos como el autómatas programable (PLC).
- Sensores: se encargan de convertir los valores de una cierta variable de proceso en señales que el controlador pueda interpretar correctamente. Entre los sensores más comunes se tienen los capacitivos, los inductivos, los fotoeléctricos, los finales de carreras, entre otros. Un sensor es un dispositivo que convierte una propiedad física



o fenómeno en una señal eléctrica correspondiente medible, tal como tensión, corriente, el cambio en los valores de resistencia o condensador, etc.

- Actuadores: son parte operativa del proceso quien llevan a cabo todas las órdenes y trabajos designados por la parte de control. Entre ellos están el accionamiento de maquinas como motores, cilindros, compresores, etc.
- Proceso: son los diferentes cambios a los que se somete una determinada combinación de variables para la obtención de un producto deseado, haciendo uso de maquinarias, equipos y del hombre.

2.2.2. Tarjeta de Adquisición de Datos y Tarjeta de acople.

La adquisición de Datos o adquisición de Señales consiste en la toma de muestras del mundo real (sistema analógico) para generar datos que puedan ser manipulados por un ordenador u otras electrónicas (sistema digital). Consiste, en tomar un conjunto de señales físicas, convertirlas en tensiones eléctricas y digitalizarlas de manera que se puedan procesar en una computadora o PAC. Se requiere una etapa de acondicionamiento, que adecua la señal a niveles compatibles con el elemento que hace la transformación a señal digital. El elemento que hace dicha transformación es el módulo de digitalización o tarjeta de Adquisición de Datos (DAQ).

National Instruments sostiene que:

“La adquisición de datos es el proceso de obtener o generar información de manera automatizada desde recursos de medidas analógicas y digitales como sensores y dispositivos bajo prueba. Utiliza una combinación de hardware y software basados en PC para brindar un sistema de medida flexible y definido por el usuario.”

Es por lo anterior que los dispositivos DAQ son instrumentos, ideales para una gran variedad de aplicaciones, desde registros de datos simples hasta sistemas integrados, ya que han sido diseñados con el propósito general de medir señales de voltaje.

La Imagen 2.1.a muestra el diagrama general de la adquisición de datos basada en PC.

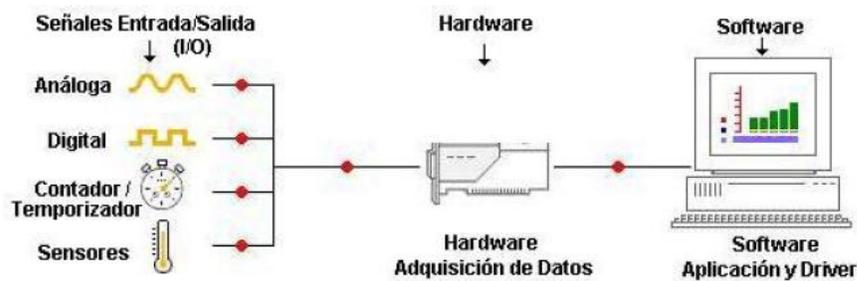


Imagen 2.1.a Diagrama general de la adquisición de datos

A la izquierda se tiene el modelo DAQ USB – 1208FS

A la derecha se tiene el modelo DAQ USB – 6008.



Imagen 2.1.b. Modelos de DAQ

Al igual que la tarjeta de acople, esta se diseño con el fin de elevar tensiones, ya que la DAQ trabaja con niveles de tensiones de 5V en entradas y salidas digitales, en entradas analógicas con 10V y salidas de 5V.



Imagen 2.2. Tarjeta de Acople

Fuente: Propia

National Instruments en el año 2006 se expandió al adquirir a Measurement Computing que es una empresa que desarrolla tarjetas de adquisición de datos de bajo coste para interfaces ISA, PCI y USB para computadoras personales. De este modo, como NI es una empresa que desarrolla tecnología en instrumentación virtual, el software NI LabVIEW ya funciona con los productos de esta compañía con sus respectivos Drivers.

Un dispositivo de este tipo, por ejemplo, el USB 1208FS posee 8 canales de entrada analógica. Tiene una resolución de 12-BIT, 50 kilo muestras/segundo (tasa de muestreo), dos salidas de D/A y 16 Digital I/O, en una presentación miniatura atractiva. Todas las señales son accesibles con tornillos en las terminales laterales. Conexión por USB

Mientras que el dispositivo DAQ USB 6008 (ver imagen 2.3) posee 8 canales de entrada analógica de 12 bits, 12 líneas DIO, 2 salidas analógicas, 1 contador.

Para dar funcionamiento a nuestro trabajo práctico es necesario instalar además de LabVIEW® 9.0 los drivers necesarios para la DAQ:



Imagen 2.3. Dispositivo DAQ USB 6008

Fuente: National Instruments

2.2.2.1. Proceso de Adquisición de Datos.

- Dato: Representación simbólica (numérica, alfabética...), atributo o característica de un valor. No tiene sentido en sí mismo, pero convenientemente tratado (procesado) se puede utilizar en la relación de cálculos o toma de decisiones.
- Adquisición: Recogida de un conjunto de variables físicas, conversión en voltaje y digitalización de manera que se puedan procesar en un ordenador.
- Sistema: Conjunto organizado de dispositivos que interactúan entre sí ofreciendo prestaciones más completas y de más alto nivel. Una vez que las señales eléctricas se transformaron en digitales, se envían a través del bus de datos a la memoria del



PC. Una vez los datos están en memoria pueden procesarse con una aplicación adecuada, archivarlas en el disco duro, visualizarlas en la pantalla, etc.

- Bit de Resolución: Número de bits que el convertidor analógico a digital (ADC) utiliza para representar una señal.
- Rango: Valores máximo y mínimo entre los que el sensor, instrumento o dispositivo funcionan bajo unas especificaciones.
- Teorema de Nyquist: Al muestrear una señal, la frecuencia de muestreo debe ser mayor que dos veces el ancho de banda de la señal de entrada, para poder reconstruir la señal original de forma exacta a partir de sus muestras. En caso contrario, aparecerá el fenómeno del aliasing (efecto que causa que señales continuas distintas se tornen indistinguibles cuando se muestrean digitalmente.) que se produce al infra-muestrear. Si la señal sufre aliasing, es imposible recuperar el original.
Velocidad de muestreo recomendada:

- $2 \times$ frecuencia mayor (medida de frecuencia).
- $10 \times$ frecuencia mayor (detalle de la forma de onda).

Los componentes de los sistemas de adquisición de datos, poseen sensores adecuados que convierten cualquier parámetro de medición de una señal eléctrica, que se adquiere por el hardware de adquisición de datos. Los datos adquiridos se visualizan, analizan, y almacenan en un ordenador, ya sea utilizando el proveedor de software suministrado u otro software. Los controles y visualizaciones se pueden desarrollar utilizando varios lenguajes de programación de propósito general como Visual BASIC, C++, Fortran, Java, Lisp, Pascal. Los lenguajes especializados de programación utilizados para la adquisición de datos incluyen EPICS, utilizada en la construcción de grandes sistemas de adquisición de datos, LabVIEW, que ofrece un entorno gráfico de programación optimizado para la adquisición de datos, y MATLAB. Estos entornos de



adquisición proporcionan un lenguaje de programación además de bibliotecas y herramientas para la adquisición de datos y posterior análisis.

De la misma manera que se toma una señal eléctrica y se transforma en una digital para enviarla al ordenador, se puede también tomar una señal digital o binaria y convertirla en una eléctrica. En este caso el elemento que hace la transformación es una tarjeta o módulo de Adquisición de Datos de salida, o tarjeta de control. La señal dentro de la memoria del PC la genera un programa adecuado a las aplicaciones que quiere el usuario y, luego de procesarla, es recibida por mecanismos que ejecutan movimientos mecánicos, a través de servomecanismos, que también son del tipo transductores.

Un sistema típico de adquisición utiliza sensores, transductores, amplificadores, convertidores analógico - digital (A/D) y digital - analógico (D/A), para procesar información acerca de un sistema físico de forma digitalizada.

2.2.2.2. ¿Cómo se adquieren los Datos?

La adquisición de datos se inicia con el fenómeno físico o la propiedad física de un objeto (objeto de la investigación) que se desea medir. Esta propiedad física o fenómeno podría ser el cambio de temperatura o la temperatura de una habitación, la intensidad o intensidad del cambio de una fuente de luz, la presión dentro de una cámara, la fuerza aplicada a un objeto, o muchas otras cosas. Un eficaz sistema de adquisición de datos puede medir todo esto diferentes propiedades o fenómenos. La capacidad de un sistema de adquisición de datos para medir los distintos fenómenos depende de los transductores para convertir las señales de los fenómenos físicos mensurables en la adquisición de datos por hardware. Transductores son sinónimo de sensores en sistemas de DAQ. Hay transductores específicos para diferentes aplicaciones, como la medición de la temperatura, la presión, o flujo de fluidos. DAQ también despliega diversas técnicas de acondicionamiento de Señales para modificar adecuadamente diferentes señales eléctricas en tensión, que luego pueden ser digitalizados usando CED.



Las señales pueden ser digitales (también llamada señales de la lógica) o analógicas en función del transductor utilizado.

El acondicionamiento de señales suele ser necesario si la señal desde el transductor no es adecuado para la DAQ hardware que se utiliza. La señal puede ser amplificada o desamplificada, o puede requerir de filtrado, o un cierre patronal, en el amplificador se incluye para realizar demodulación. Varios otros ejemplos de acondicionamiento de señales podría ser el puente de conclusión, la prestación actual de tensión o excitación al sensor, el aislamiento, linealización, etc. Este pre-tratamiento de señal normalmente lo realiza un pequeño módulo acoplado al transductor.

DAQ hardware son por lo general las interfaces entre la señal y un PC. Podría ser en forma de módulos que pueden ser conectados a la computadora de los puertos (paralelo, serie, USB, etc.) o ranuras de las tarjetas conectadas a (PCI, ISA) en la placa madre. Por lo general, el espacio en la parte posterior de una tarjeta PCI es demasiado pequeño para todas las conexiones necesarias, de modo que una ruptura de caja externa es obligatoria. El cable entre este recuadro y el PC es cara debido a los numerosos cables y el blindaje necesario y porque es exótico. Las tarjetas DAQ a menudo contienen múltiples componentes (multiplexores, ADC, DAC, TTL-IO, temporizadores de alta velocidad, memoria RAM). Estos son accesibles a través de un bus por un micro controlador, que puede ejecutar pequeños programas. El controlador es más flexible que una unidad lógica dura cableada, pero más barato que una CPU de modo que es correcto para bloquear con simples bucles de preguntas.

Driver software normalmente viene con el hardware DAQ o de otros proveedores, y permite que el sistema operativo pueda reconocer el hardware DAQ y dar así a los programas acceso a las señales de lectura por el hardware DAQ. Un buen conductor ofrece un alto y bajo nivel de acceso.



2.2.2.3. Tiempo de Conversión.

Es el tiempo que tarda en realizar una medida el convertidor en concreto, y dependerá de la tecnología de medida empleada. Evidentemente nos da una cota máxima de la frecuencia de la señal a medir.

Este tiempo se mide como el transcurrido desde que el convertidor recibe una señal de inicio de "conversión" (normalmente llamada SOC, Start of Conversión) hasta que en la salida aparece un dato válido. Para que tengamos constancia de un dato válido tenemos dos caminos:

- Esperar el tiempo de conversión máximo que aparece en la hoja de características.
- Esperar a que el convertidor nos envíe una señal de fin de conversión.

Si no respetamos el tiempo de conversión, en la salida tendremos un valor, que dependiendo de la constitución del convertidor será:

- Un valor aleatorio, como consecuencia de la conversión en curso
- El resultado de la última conversión

2.2.2.4. La etapa de Accionamiento de la señal.

Con más detalle, en una etapa de acondicionamiento podemos encontrar estas etapas, aunque no todas están siempre presentes:

- **Amplificación** Es el tipo más común de acondicionamiento. Para conseguir la mayor precisión posible la señal de entrada debe ser amplificada de modo que su máximo nivel coincida con la máxima tensión que el convertidor pueda leer.
- **Aislamiento** - Otra aplicación habitual en el acondicionamiento de la señal es el aislamiento eléctrico entre el transductor y el ordenador, para proteger al mismo de



transitorios de alta tensión que puedan dañarlo. Un motivo adicional para usar aislamiento es el garantizar que las lecturas del convertidor no son afectadas por diferencias en el potencial de masa o por tensiones en modo común. Cuando el sistema de adquisición y la señal a medir están ambas referidas a masa pueden aparecer problemas si hay una diferencia de potencial entre ambas masas, apareciendo un "bucle de masa", que puede devolver resultados erróneos.

- Multiplexado - El multiplexado es la conmutación de las entradas del convertidor, de modo que con un sólo convertidor podemos medir los datos de diferentes canales de entrada. Puesto que el mismo convertidor está midiendo diferentes canales, su frecuencia máxima de conversión será la original dividida por el número de canales muestreados. Se aconseja que los multiplexores se utilicen antes del conversor y después del condicionamiento de la señal, ya que de esta manera no molestará a los aislantes que podamos tener.
- Filtrado - El fin del filtro es eliminar las señales no deseadas de la señal que estamos observando. Por ejemplo, en las señales cuasi-continuas, (como la temperatura) se usa un filtro de ruido de unos 4 Hz, que eliminará interferencias, incluidos los 50/60 Hz de la red eléctrica. Las señales alternas, tales como la vibración, necesitan un tipo distinto de filtro, conocido como filtro antialiasing, que es un filtro pasabajo pero con un corte muy brusco, que elimina totalmente las señales de mayor frecuencia que la máxima a medir, ya que si no se eliminasen aparecerían superpuestas a la señal medida, con el consiguiente error.
- Excitación - La etapa de acondicionamiento de señal a veces genera excitación para algunos transductores, como por ejemplos las galgas "extesométricas", "termistores" o "RTD", que necesitan de la misma, bien por su constitución interna, (como el termistor, que es una resistencia variable con la temperatura) o bien por la



configuración en que se conectan (como el caso de las galgas, que se suelen montar en un puente de Wheatstone).

- Linealización - Muchos transductores, como los termopares, presentan una respuesta no lineal ante cambios lineales en los parámetros que están siendo medidos. Aunque la linealización puede realizarse mediante métodos numéricos en el sistema de adquisición de datos, suele ser una buena idea el hacer esta corrección mediante circuitería externa.

2.2.2.5. Ventajas de la tarjeta de adquisición de datos.

- Flexibilidad de procesamiento, posibilidad de realizar las tareas en tiempo real o en análisis posteriores (a fin de analizar los posibles errores), gran capacidad de almacenamiento, rápido acceso a la información y toma de decisión, se adquieren gran cantidad de datos para poder analizar, posibilidad de emular una gran cantidad de dispositivos de medición y activar varios instrumentos al mismo tiempo, facilidad de automatización, etc.
- Se utiliza en la industria, la investigación científica, el control de máquinas y de producción, la detección de fallas y el control de calidad entre otras aplicaciones.

2.2.2.6. Tarjeta de Adquisición de Datos USB 6008.

Este modelo de tarjeta que es con la que se va a implementar el proyecto consta de:

- 8 canales de entrada analógica de 12 bits, 12 líneas DIO, 2 salidas analógicas, 1 contador.
- Considere el NI USB-6210 y NI USB-6211 para un mayor rendimiento.
- Alimentación de energía por el bus para mayor comodidad y portabilidad.
- Obtenga los paquetes que tienen una tarjeta para aplicaciones OEM.
- Software controlador para Windows, Mac OS X, Linux, Pocket PC.

- El software de NI-DAQmx y software interactivo NI LabVIEW SignalExpress para registro de datos.

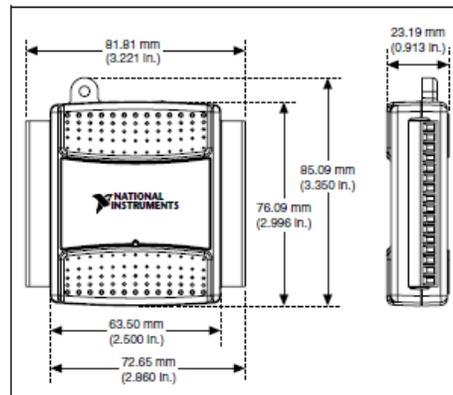
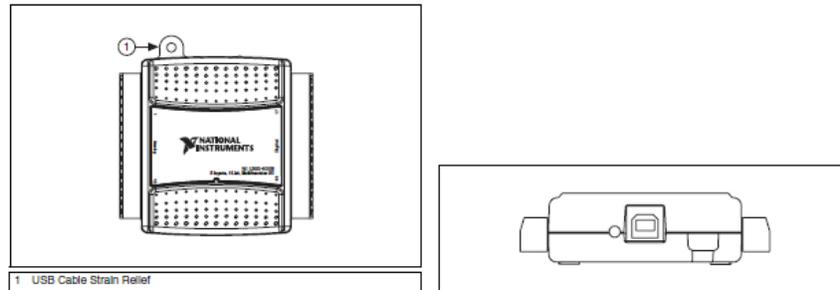


Imagen 2.4. Especificaciones y Medidas de la Tarjeta de Adquisición a usar

National Instruments

Algunos pasos de instalación de software para la interfaz tarjeta – maquina.

- Antes de instalar el dispositivo, debe instalar el software que va a utilizar con el dispositivo. Se refieren a la instalación de la sección Software de esta guía y la documentación incluida con el software para obtener más información.

La Imagen 2.5 muestra los principales componentes funcionales de la USB-6008/6009 NI.

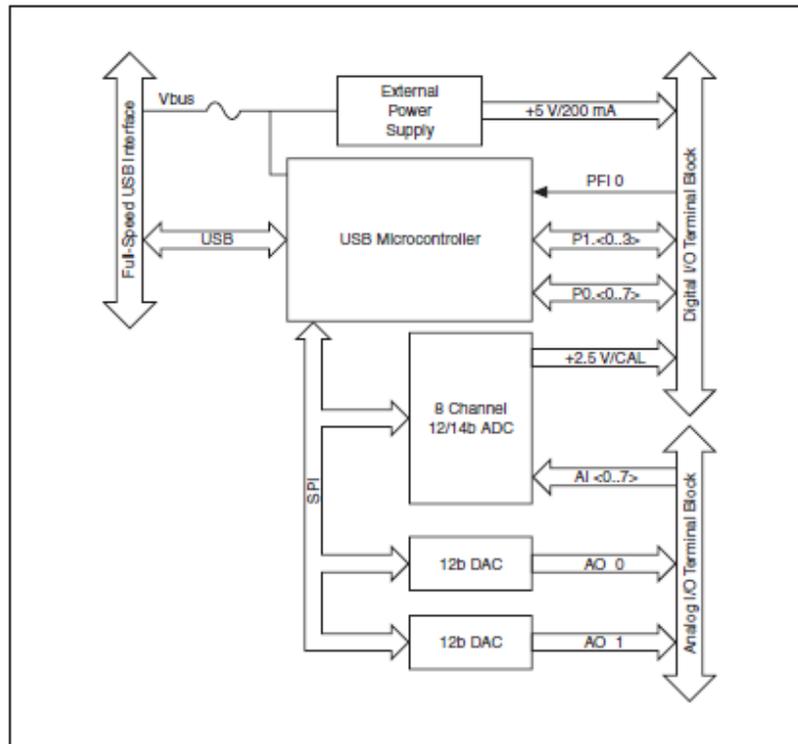


Imagen 2.5. Componentes funcionales

Fuente: National Instruments

- Complete los siguientes pasos para configurar el NI USB-6008/6009:
 1. Instalar los bloques de terminales de tornillo mediante la inserción de ellos.
 2. La Imagen 2.6 muestra las etiquetas de la señal que se incluyen en el USB-6008/6009 NI kit. Puede aplicar las etiquetas de la señal a los bloques de terminal de tornillo para identificación de la señal fácil.

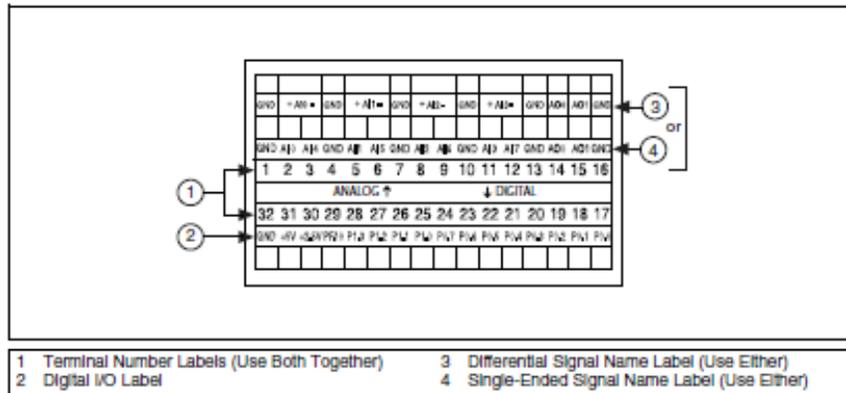


Imagen 2.6. Etiquetas de la DAQ.

Fuente: National Instruments

3. Consulte las Tablas 2.1 y 2.2, las Imagenes 2.6 y 2.7 para la orientación de la señal de la etiqueta y colocar las etiquetas de la señal proporcionada a los bloques de terminal de tornillo. Se pueden insertar los bloques de terminales de tornillo en cualquiera de las tomas.

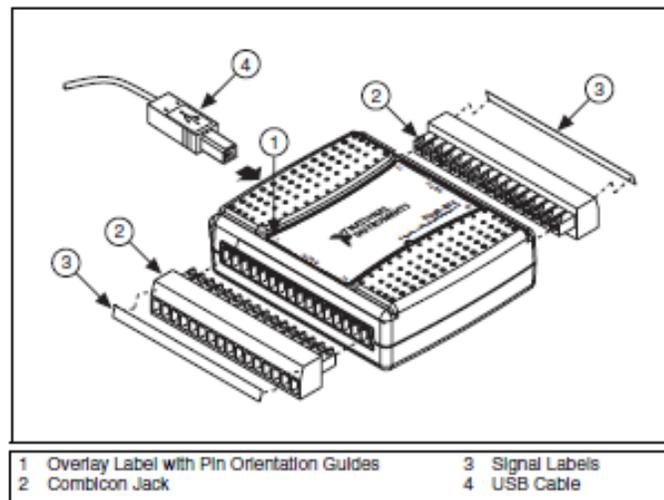


Imagen 2.7. Modo de Insertar los porta-pines de la DAQ

Fuente: National Instruments

4. Conecte los cables a los terminales de tornillo apropiado.



Indicador del LED:

El dispositivo NI USB-6008/6009 tiene un LED verde al lado del conector USB. El indicador LED indica el estado del dispositivo, como se indica en la Tabla 2.3. Cuando el dispositivo está conectado a un puerto USB, el LED parpadea cada vez que indica que el dispositivo se inicia y está recibiendo alimentación de la conexión. Si el LED no parpadea, puede significar que el dispositivo no se inicia o el ordenador está en modo de espera. Para que el dispositivo sea reconocido, el dispositivo debe estar conectado a un equipo que tiene instalado NI-DAQmx sobre el mismo. Si el dispositivo no parpadea, asegúrese de que su equipo tiene la última versión de NI-DAQmx instalado en él.

Tabla 2.3: Estados de Leds

LED State	Device Status
Not lit	Device not connected or in suspend.
On, not blinking	Device connected.
Single-blink	Operating normally.

La NI USB-6008/6009 viene con un Bloque de terminales para señales analógicas y un Bloque de terminales para las señales digitales. Estos bloques de terminales ofrecen 16 conexiones que utilizan 16 AWG para 28 AWG.

Tablas 2.1 y 2.2: Terminales Analógicos y Digitales

Module	Terminal	Signal, Single-Ended Mode	Signal, Differential Mode
	1	GND	GND
	2	AI 0	AI 0+
	3	AI 4	AI 0-
	4	GND	GND
	5	AI 1	AI 1+
	6	AI 5	AI 1-
	7	GND	GND
	8	AI 2	AI 2+
	9	AI 6	AI 2-
	10	GND	GND
	11	AI 3	AI 3+
	12	AI 7	AI 3-
	13	GND	GND
	14	AO 0	AO 0
	15	AO 1	AO 1
	16	GND	GND

Module	Terminal	Signal
	17	P0.0
	18	P0.1
	19	P0.2
	20	P0.3
	21	P0.4
	22	P0.5
	23	P0.6
	24	P0.7
	25	P1.0
	26	P1.1
	27	P1.2
	28	P1.3
	29	PFI 0
	30	+2.5 V
	31	+5 V
	32	GND



2.2.3. ¿Qué es LabView?

LabVIEW fue creado en 1976 por la empresa National Instruments (NI) con el propósito de funcionar sobre máquinas Apple Macintosh (MAC). Actualmente está disponible en los principales sistemas operativos utilizados en el mundo, como es Windows y Linux, entre otros. El 16 de abril de 2007, National Instruments anunció la disponibilidad de una nueva versión de LabVIEW, como es LabVIEW 9.0 éste software puede ser utilizado en la versión más nueva del sistema operativo de Microsoft (Windows 7).

La National Instruments señala que:

“LabVIEW es un revolucionario entorno de desarrollo gráfico con funciones integradas para realizar adquisición de datos, control de instrumentos, análisis de medida y presentaciones de datos. LabVIEW le da la flexibilidad de un potente ambiente de programación, pero mucho más sencillo que los entornos tradicionales”. Esta definición es bastante clara, el software LabVIEW tiene funciones específicas para acelerar el desarrollo de aplicaciones de medida, control y automatización, nos proporciona herramientas poderosas para que el usuario pueda crear aplicaciones sin líneas de código (lenguaje G) y nos permite colocar objetos ya construidos para lograr crear interfaces de usuario rápidamente. Después es uno mismo el que especifica las funciones del sistema construyendo diagramas de bloques. No podemos dejar de mencionar que LabVIEW se puede conectar con todo tipo de hardware incluyendo instrumentos de escritorio, tarjetas insertables, controladores de movimiento y controladores lógicos programables (PLCs).

LabVIEW® es una herramienta gráfica para pruebas, control y diseño mediante la programación. El lenguaje que usa se llama lenguaje G, donde la G simboliza que es lenguaje Gráfico. Ahora está disponible para las plataformas Windows, UNIX, MAC y GNU/Linux. La última versión es la 2010.



Los programas desarrollados con LabVIEW® se llaman Instrumentos Virtuales, o VIs, y su origen provenía del control de instrumentos, aunque hoy en día se ha expandido ampliamente no sólo al control de todo tipo de electrónica (Instrumentación electrónica) sino también a su programación embebida. Un lema tradicional de LabView es: "*La potencia está en el Software*", que con la aparición de los sistemas multinúcleo se ha hecho aún más patente. Entre sus objetivos están el reducir el tiempo de desarrollo de aplicaciones de todo tipo (no sólo en ámbitos de Pruebas, Control y Diseño) y el permitir la entrada a la informática a profesionales de cualquier otro campo. LabView consigue combinarse con todo tipo de software y hardware, tanto del propio fabricante -tarjetas de adquisición de datos, PAC, Visión, instrumentos y otro Hardware- como de otros fabricantes.

2.2.3.1. Principales Usos.

Es usado principalmente por ingenieros y científicos para tareas como:

- Adquisición de datos y análisis matemático
- Comunicación y control de instrumentos de cualquier fabricante
- Automatización industrial y programación de PACs (Controlador de Automatización Programable)
- Diseño de controladores: simulación, prototipaje rápido, hardware-en-el-ciclo (HIL) y validación
- Diseño embebido de micros y chips
- Control y supervisión de procesos
- Visión artificial y control de movimiento
- Robótica
- Domótica y redes de sensores inalámbricos
- En 2008 el programa fue utilizado para controlar el LHC, el acelerador de partículas más grande construido hasta la fecha.
- Pero también juguetes como el Lego Mindstorms o el WeDo lo utilizan, llevando la programación gráfica a niños de todas las edades.



2.2.3.2. Principales Características.

Su principal característica es la facilidad de uso, válido para programadores profesionales como para personas con pocos conocimientos en programación pueden hacer (programas) relativamente complejos, imposibles para ellos de hacer con lenguajes tradicionales. También es muy rápido hacer programas con LabView y cualquier programador, por experimentado que sea, puede beneficiarse de él. Los programas en LabView son llamados instrumentos virtuales (VIs) Para los amantes de lo complejo, con LabView pueden crearse programas de miles de VIs (equivalente a millones de páginas de código texto) para aplicaciones complejas, programas de automatizaciones de decenas de miles de puntos de entradas/salidas, proyectos para combinar nuevos VIs con VIs ya creados, etc. Incluso existen buenas prácticas de programación para optimizar el rendimiento y la calidad de la programación. El labView 7.0 introduce un nuevo tipo de subVI llamado VIs Expreso (Express VIs). Estos son VIs interactivos que tienen una configuración de caja de diálogo que permite al usuario personalizar la funcionalidad del VI Expreso. El VIs estándar son VIs modulares y personalizables mediante cableado y funciones que son elementos fundamentales de operación de LabView.

Presenta facilidades para el manejo de:

- Interfaces de comunicaciones:
 - Puerto serie
 - Puerto paralelo
 - GPIB
 - PXI
 - VXI
 - TCP/IP, UDP, DataSocket
 - Irda
 - Bluetooth
 - USB

- OPC.
- Capacidad de interactuar con otros lenguajes y aplicaciones:
 - DLL: librerías de funciones
 - .NET
 - ActiveX
 - Multisim
 - Matlab/Simulink
 - AutoCAD, SolidWorks, etc.
- Herramientas gráficas y textuales para el procesado digital de señales.
- Visualización y manejo de gráficas con datos dinámicos.
- Adquisición y tratamiento de imágenes.
- Control de movimiento (combinado incluso con todo lo anterior).
- Tiempo Real estrictamente hablando.
- Programación de FPGAs para control o validación.
- Sincronización entre dispositivos.

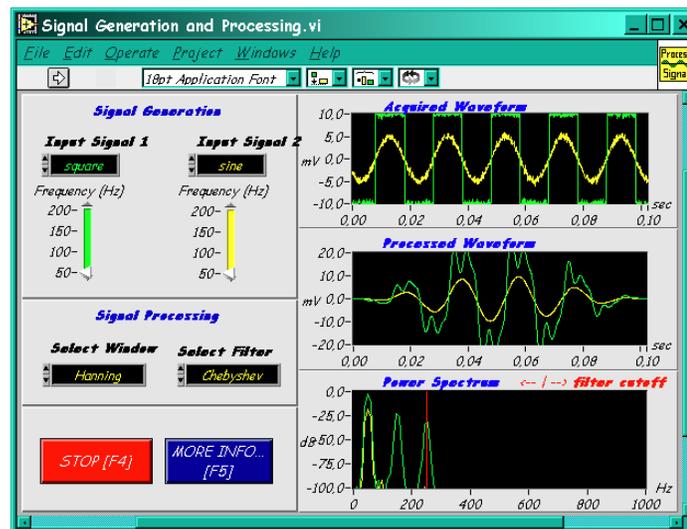


Imagen 2.8. Generación de Señales y Procesamiento

Fuente: Extraída de LabVIEW

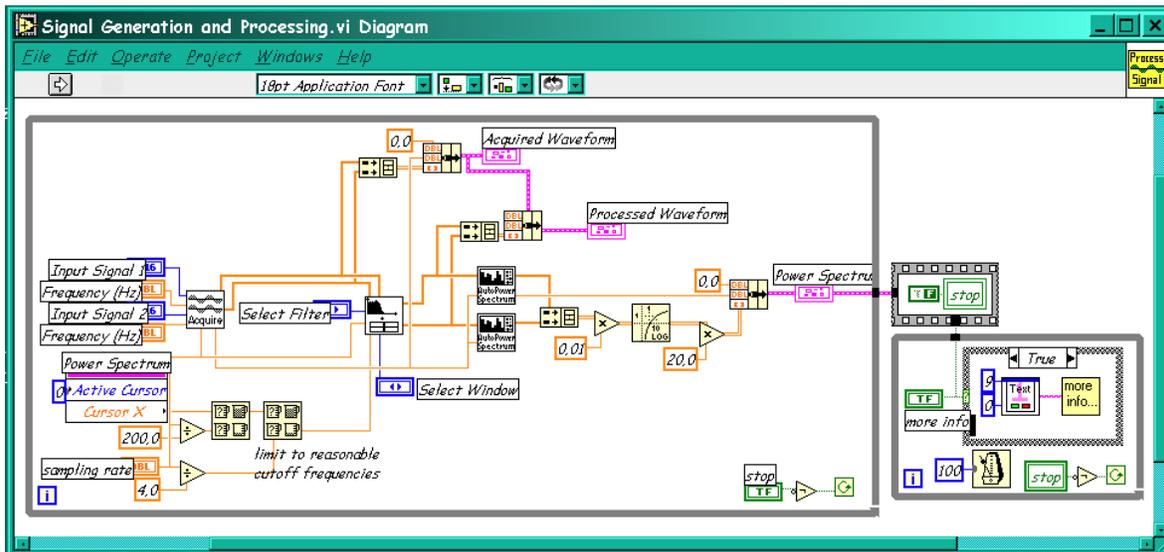


Imagen 2.9. Generación de Señal y Proceso. (Modo Gráfico)

Fuente: Extraída de LabVIEW

2.2.3.3. Ventajas de LabView.

- LabVIEW es ideal para la adquisición de datos y monitorización en la industria, es por ello que creamos y mejoramos una aplicación para la Pesquera el Golfo Corral de Valdivia.
- Se reduce el tiempo de desarrollo de las aplicaciones al menos de 4 a 10 veces, ya que es muy intuitivo y fácil de aprender.
- Dota de gran flexibilidad al sistema, permitiendo cambios y actualizaciones tanto del hardware como del software.
- Da la posibilidad a los usuarios de crear soluciones completas y complejas.



- Con un único sistema de desarrollo se integran las funciones de adquisición, análisis y presentación de datos.
- El sistema está dotado de un compilador gráfico para lograr la máxima velocidad de ejecución posible.
- Tiene la posibilidad de incorporar aplicaciones escritas en otros lenguajes.

LabVIEW es un entorno de programación destinado al desarrollo de aplicaciones, similar a los sistemas de desarrollo comerciales que utilizan el *lenguaje C* o *BASIC*. Sin embargo, LabVIEW se diferencia de dichos programas en un importante aspecto: los citados lenguajes de programación se basan en líneas de texto para crear el código fuente del programa, mientras que LabVIEW emplea la programación gráfica o *lenguaje G* para crear programas basados en diagramas de bloques. Para el empleo de LabVIEW no se requiere gran experiencia en programación, ya que se emplean iconos, términos e ideas familiares a científicos e ingenieros, y se apoya sobre símbolos gráficos en lugar de lenguaje escrito para construir las aplicaciones. Por ello resulta mucho más intuitivo que el resto de lenguajes de programación convencionales.

LabVIEW posee extensas librerías de funciones y subrutinas. Además de las funciones básicas de todo lenguaje de programación, LabVIEW incluye librerías específicas para la adquisición de datos, control de instrumentación VXI, GPIB y comunicación serie, análisis presentación y guardado de datos. LabVIEW también proporciona potentes herramientas que facilitan la depuración de los programas.

2.2.3.4. ¿Cómo trabaja LabVIEW?

Los programas desarrollados mediante LabVIEW se denominan *Instrumentos Virtuales (VIs)*, porque su apariencia y funcionamiento imitan los de un instrumento real. Sin embargo son análogos a las funciones creadas con los lenguajes de programación convencionales. Los *VIs* tienen una parte interactiva con el usuario y otra parte de código fuente, y aceptan parámetros procedentes de otros *VIs*. Todos los *VIs* tienen un *panel frontal* y un *diagrama de bloques*. Las *paletas* contienen las opciones que se emplean para crear y modificar los *VIs*. A continuación se procederá a realizar una somera descripción de estos conceptos.

Al arrancar el software a través del acceso directo o del menú de Inicio» Programas» National Instruments LabVIEW 9.0, se nos presenta la ventana (ver imagen 2.10) en donde se da inicio a la construcción del VI.

Blank VI = VI en blanco



Imagen 2.10. Ventana Inicial al abrir LabVIEW

Fuente: Extraída de LabVIEW

Al hacer clic en “Blank VI”, automáticamente se abren 2 ventanas. Una contiene al Panel Frontal (izquierda) y la otra al Diagrama de bloque (derecha). (Ver imagen 2.11)

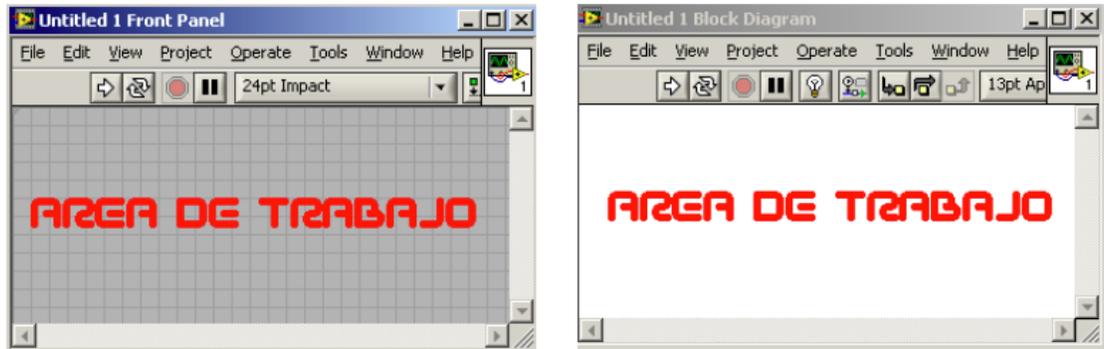


Imagen 2.11. Ventanas de Panel Frontal y de Bloque

Fuente: Extraída de LabVIEW

A) Panel Frontal (Front Panel)

Se utiliza como interface entre usuario/VI y que es donde los datos son manipulados, controlados y monitoreados. Se construye a partir de controles (entradas) e indicadores (salidas). Se trata de la interfaz gráfica del VI con el usuario. Esta interfaz recoge las entradas procedentes del usuario y representa las salidas proporcionadas por el programa. Un *panel frontal* está formado por una serie de botones, pulsadores, potenciómetros, gráficos, etc. Cada uno de ellos puede estar definido como un *control* (A.1) o un *indicador* (A.2). Los primeros sirven para introducir parámetros al VI, mientras que los indicadores se emplean para mostrar los resultados producidos, ya sean datos adquiridos o resultados de alguna operación.

A.1) Los **controles** simulan instrumentos de entrada y entregan los respectivos datos al diagrama de bloques del VI. Entre los controles tenemos perillas, pushbuttons y otros dispositivos de entrada. (Ver imagen 2.12)



Imagen 2.12. Botones Perillas y otros dispositivos de entrada

Fuente: Extraída de LabVIEW

A.2) Los *indicadores* simulan instrumentos de salida y muestran los datos que el diagrama de bloques genera o adquiere. Dentro de los indicadores podemos encontrar leds, gráficos y otros tipos de displays. (Ver imagen 2.13)



Imagen 2.13. Indicadores Leds y de nivel

Fuente: Extraída de LabVIEW

B) Diagrama de bloques (Block diagram)

Contiene el código gráfico G que define la funcionalidad del VI. Por ende, podemos ver la estructura del programa de una forma gráfica donde los datos fluyen a través de cables o líneas. Además contiene las librerías de LabVIEW como son las funciones y estructuras para conseguir nuestro programa.

El *diagrama de bloques* se construye conectando los distintos objetos entre sí, como si de un circuito se tratara. Los cables unen terminales de entrada y salida con los objetos correspondientes, y por ellos fluyen los datos. LabVIEW posee una extensa biblioteca de

funciones, entre ellas, aritméticas, comparaciones, conversiones, funciones de entrada/salida, de análisis, etc.

Las *estructuras*, similares a las declaraciones causales y a los bucles en lenguajes convencionales, ejecutan el código que contienen de forma condicional o repetitiva (bucle *for*, *while*, *case*,...). Los cables son las trayectorias que siguen los datos desde su origen hasta su destino, ya sea una función, una estructura, un terminal, etc. Cada cable tiene un color o un estilo diferente, lo que diferencia unos tipos de datos de otros.

En resumen, una vez construido el panel frontal, se crea automáticamente el código gráfico en el diagrama de bloques, representando las funciones de los controles que fueron puestos en el panel frontal y sólo bastaría unir correctamente los terminales de los controles e indicadores para el funcionamiento del VI creado.

Con la ayuda de las distintas **paletas** que aparecen dentro de los menús de LabVIEW, todo lo anterior es mucho más fácil. Dentro del panel frontal se desprenden dos tipos de paletas: Controls Palette (paleta de controles) y Tools Palette (paleta de herramientas).

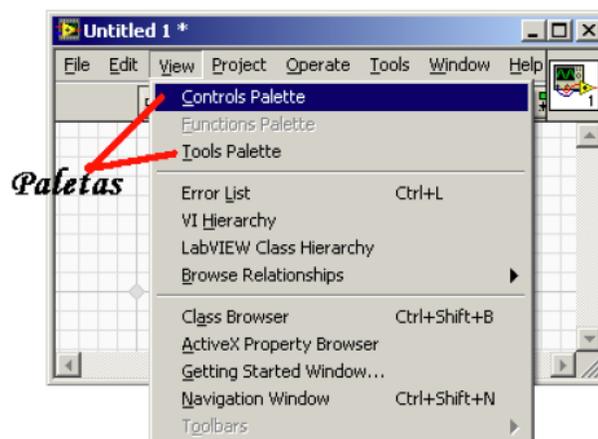


Imagen 2.14. Paletas de Control y Herramientas

Fuente: Extraída de LabVIEW

C) Paletas.

Las *paletas* de LabVIEW proporcionan las herramientas que se requieren para crear y modificar tanto el *panel frontal* como el *diagrama de bloques*. Existen las siguientes paletas:

- Paleta de herramientas (Tools Palette)

Se emplea tanto en el *panel frontal* como en el *diagrama de bloques*. Contiene las herramientas necesarias para editar y depurar los objetos tanto del *panel frontal* como del *diagrama de bloques*.

La **paleta de herramientas** está habilitada en el panel frontal y en el diagrama de bloques y es necesaria para trabajar o modificar los objetos. Una herramienta se refiere al modo especial del cursor del mouse, esto quiere decir que cuando uno selecciona una herramienta, el cursor cambia de icono al icono de la herramienta. Haciendo clic en el led de color verde uno habilita o deshabilita el cambio automático del cursor dependiendo de la función a realizar.

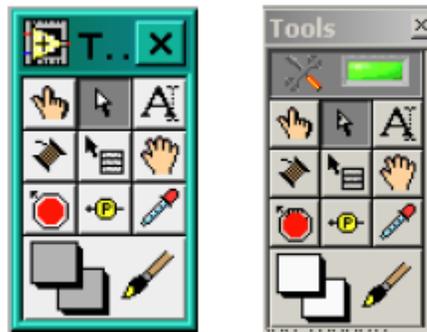


Imagen 2.15. Paletas de Herramientas

Fuente: Extraída de LabVIEW

Las opciones que presenta esta paleta son las siguiente:



Operating tool – Cambia el valor de los controles.



Positioning tool – Desplaza, cambia de tamaño y selecciona los objetos.



Labeling tool – Edita texto y crea etiquetas.



Wiring tool – Une los objetos en el *diagrama de bloques*.



Object Pop-up Menu tool – Abre el menú desplegable de un objeto.



Scroll tool – Desplaza la pantalla sin necesidad de emplear las barras de desplazamiento.



Breakpoint tool – Fija puntos de interrupción de la ejecución del programa en *VI*s, funciones y estructuras.



Probe tool – Crea puntos de prueba en los cables, en los que se puede visualizar el valor del dato que fluya por dicho cable en cada instante.



Color Copy tool – Copia el color para después establecerlo mediante la siguiente herramienta.



Color tool – Establece el color de fondo y el de los objetos

- Paleta de controles (Controls Palette)

La **paleta de controles** está habilitada solamente en el panel frontal y posee los controles e indicadores que se necesitan para crear el panel frontal. La paleta también aparece haciendo clic con el botón derecho del mouse sobre el área de trabajo del panel frontal.

Se utiliza únicamente en el *panel frontal*. Contiene todos los *controles e indicadores* que se emplearán para crear la interfaz del VI con el usuario.

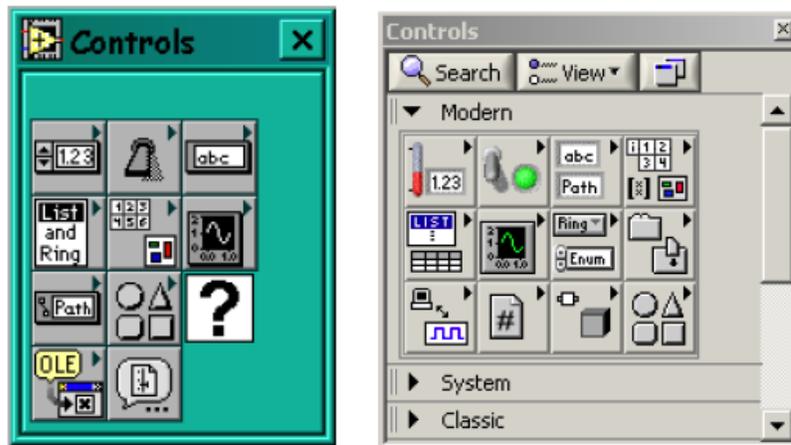


Imagen 4.16. Panel Frontal de controles

Fuente: Extraída de LabVIEW

Nota: manteniendo la tecla “shif” presionada y haciendo clic con el botón derecho del mouse sobre el área de trabajo, aparece la paleta de herramientas. Dentro del diagrama de bloque está la Functions Palette (paleta de funciones) y la paleta de herramientas como mencionamos anteriormente.

El menú *Controls* de la ventana correspondiente al panel frontal contiene las siguientes opciones:



Numeric – Para la introducción y visualización de cantidades numéricas.



Boolean – Para la entrada y visualización de valores booleanos.



String & Table – Para la entrada y visualización de texto.



List & Ring – Para visualizar y/o seleccionar una lista de opciones.



Array & Cluster – Para agrupar elementos.



Graph – Para representar gráficamente los datos.



Path & RefNum – Para gestión de archivos.



Decorations – Para introducir decoraciones en el *panel frontal*. No visualizan datos.



User Controls – Para elegir un *control* creado por el propio usuario.



ActiveX – Para transferir datos y programas de unas aplicaciones a otras dentro de Windows.



Select a Control – Para seleccionar cualquier *control*.

Al seleccionar objetos desde el menú *Controls* estos aparecen sobre el *panel frontal*, pueden colocarse donde convenga, y además tienen su propio menú desplegable que permite la configuración de algunos parámetros específicos de cada tipo de *control*.

- Paleta de funciones (Functions Palette)

Se emplea en el diseño del *diagrama de bloques*. La *paleta de funciones* contiene todos los objetos que se emplean en la implementación del programa del VI, ya sean *funciones* aritméticas, de entrada/salida de señales, entrada/salida de datos a fichero, adquisición de señales, temporización de la ejecución del programa.

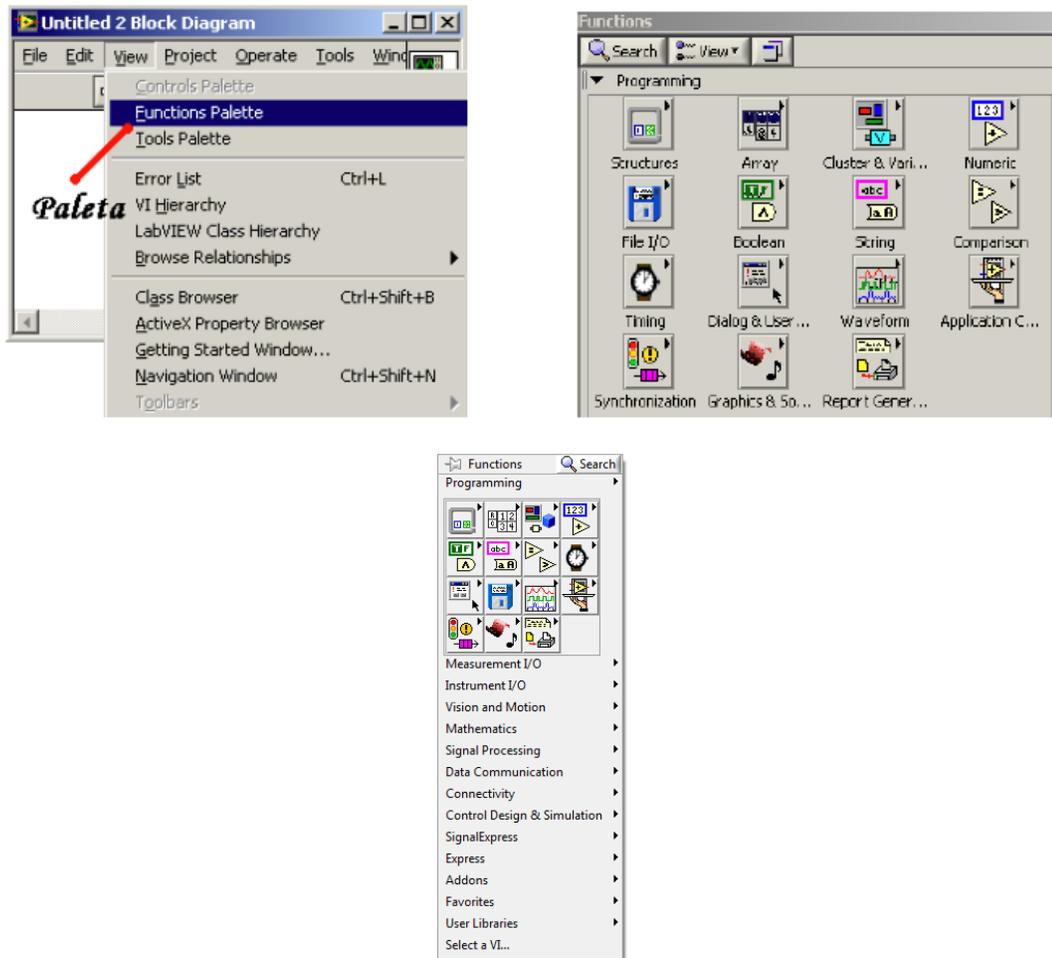


Imagen 2.17. Paleta de Funciones.

Fuente: Extraída de LabVIEW

Para seleccionar una *función* o *estructura* concretas, se debe desplegar el menú *Functions* y elegir entre las opciones que aparecen.

Existen 6 opciones para visualizar las paletas, ya sea de funciones o controles.

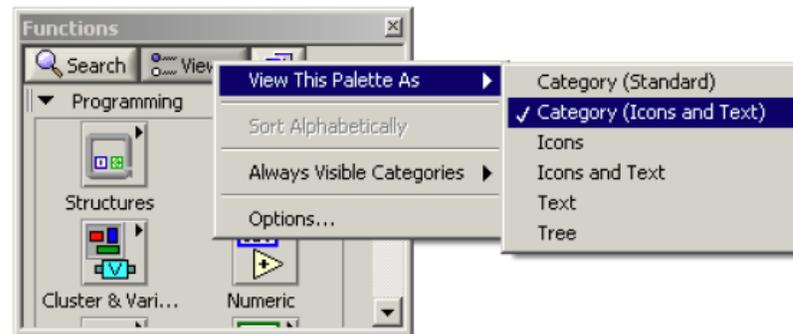


Imagen 2.18. Paletas de Funciones

Fuente: Extraída de LabVIEW



Structures – Muestra las *estructuras* de control del programa, junto con las variables locales y globales.



Numeric – Muestra *funciones* aritméticas y constantes numéricas.



Boolean – Muestra *funciones* y constantes lógicas.



String – Muestra *funciones* para manipular cadenas de caracteres, así como constantes de caracteres.



Array – Contiene *funciones* útiles para procesar datos en forma de vectores, así como constantes de vectores.



Cluster – Contiene *funciones* útiles para procesar datos procedentes de gráficas y destinados a ser representados en ellas, así como las correspondientes constantes.



Comparison – Muestra *funciones* que sirven para comparar números, valores booleanos o cadenas de caracteres.



Time & Dialog – Contiene *funciones* para trabajar con cuadros de diálogo, introducir contadores y retardos, etc.



File I/O – Muestra *funciones* para operar con ficheros.



Communication – Muestra diversas *funciones* que sirven para comunicar varios ordenadores entre sí, o para permitir la comunicación entre distintos programas.



Instrument I/O – Muestra un submenú de *VI*s, que facilita la comunicación con instrumentos periféricos que siguen la norma ANSI/IEEE 488.2-1987, y el control del puerto serie.



Data Acquisition – Contiene a su vez un submenú donde puede elegirse entre distintas librerías referentes a la adquisición de datos.



Analysis – Contiene un submenú en el que se puede elegir entre una amplia gama de *funciones* matemáticas de análisis.



Tutorial – Incluye un menú de *VI*s que se utilizan en el manual LabVIEW Tutorial.



Advanced – Contiene diversos submenús que permiten el control de la ayuda, de los *VI*s, manipulación de datos, procesamiento de eventos, control de la memoria, empleo de programas ejecutables o incluidos en librerías DLL, etc.



Instrument drivers – En él se muestran los drivers disponibles de distintos instrumentos.



User Libraries – Muestra as librerías definidas por el usuario. En este caso, la librería mostrada contiene los drivers de la tarjeta de adquisición de datos de Advantech.



Application control – Contiene varias *funciones* que regulan el funcionamiento de la propia aplicación en ejecución.



Select a VI – Permite seleccionar cualquier *VI* para emplearlo como *subVI*.

2.2.3.5. Programación en LabView.

Con el entorno gráfico de programación de LabVIEW se comienza a programar a partir del *panel frontal*. En primer lugar se definirán y seleccionarán de la *paleta de controles* todos los *controles* (entradas que dará el usuario) e *indicadores* (salidas que presentará en pantalla el *VI*) que se emplearán para introducir los datos por parte del usuario y presentar en pantalla los resultados.

Una vez colocados en la ventana correspondiente al *panel frontal* todos los objetos necesarios, debe pasarse a la ventana *Diagram* (menú *Windows > Show Diagram*), que es donde se realiza la programación propiamente dicha (*diagrama de bloques*). Al abrir esta

ventana, en ella se encuentran los terminales correspondientes a los objetos situados en el *panel frontal*, dispuestos automáticamente por LabVIEW.

Se deben ir situando las *funciones*, *estructuras*, etc. que se requieran para el desarrollo del programa, las cuales se unen a los terminales mediante cables. Para facilitar la tarea de conexión de todos los terminales, en el menú “*Help*” puede elegirse la opción “*Show Help*”, con lo que al colocar el cursor del ratón sobre un elemento aparece una ventana con información relativa a éste (parámetros de entrada y salida). Además, si se tiene seleccionado el cursor de cableado, al situar éste sobre un elemento se muestran los terminales de forma intermitente.

2.2.3.6. Ejecución en un VI.

Una vez se ha concluido la programación del VI sin errores se debe proceder a su ejecución. Para ello la ventana activa debe ser el *panel frontal* (si se está en la ventana del *diagrama de bloques*, se debe seleccionar la opción *Show Panel* del menú *Windows*). Una vez situados en el *panel frontal*, se pulsará el botón de *Run*, situado en la barra de herramientas.

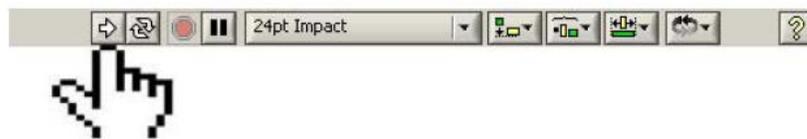
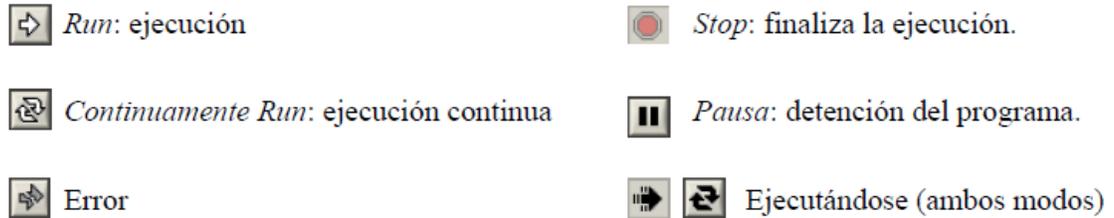


Imagen 2.19. Ejecución de un VI

Fuente: Extraída de LabVIEW



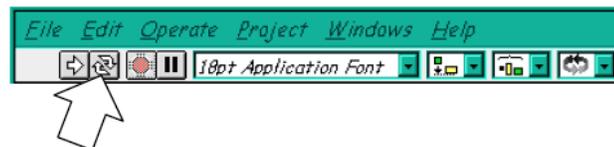
El programa comenzará a ejecutarse. Mientras dura la ejecución del mismo, la apariencia del botón de *Run* es la que se muestra a continuación:



Imagen 2.20. Ejecutando el VI.

Fuente: Extraída de LabVIEW

De este modo el programa se ejecutará una sola vez. Si se desea una ejecución continua, se pulsará el botón situado a la derecha del de *Run* (*Continuous Run*). Si durante el funcionamiento continuo del programa se vuelve a pulsar el citado botón, se finalizará la última ejecución del mismo, tras lo cual el programa se parará.



Continuous Run

Imagen 2.21. Ejecuciones continuas de un VI.

Fuente: Extraída de LabVIEW

Para finalizar la ejecución de un programa se puede operar de dos formas. La primera, y la más aconsejable, es emplear un botón en la *panel frontal* del VI, cuya pulsación produzca la interrupción del bucle de ejecución de la aplicación. La segunda

forma de detener la ejecución del VI es pulsando el botón de *pausa* o el de *stop*. La diferencia entre ambos es que si se pulsa *stop*, la ejecución del programa finaliza inmediatamente, mientras que si se pulsa *pausa*, se produce una detención en el funcionamiento del programa, retomándose su ejecución una vez se vuelve a pulsar el mismo botón.

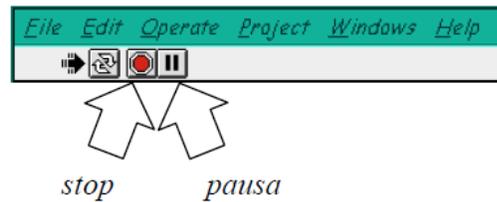


Imagen 2.22. Botones de pausa y parada de un VI

Fuente: Extraída de LabVIEW

Si existe error, se puede pulsar el mismo botón de Run y nos muestra el tipo de error que se está cometiendo (líneas rotas, conexiones malas, etc.). Según la necesidad del programador, este puede colocar un botón de stop en el panel frontal de su VI, cuya pulsación provoque una interrupción en el bucle de ejecución de la aplicación.

Con todo lo anterior se procede a dar un pequeño ejemplo sobre la operabilidad de funciones lógicas.

Ejemplo:

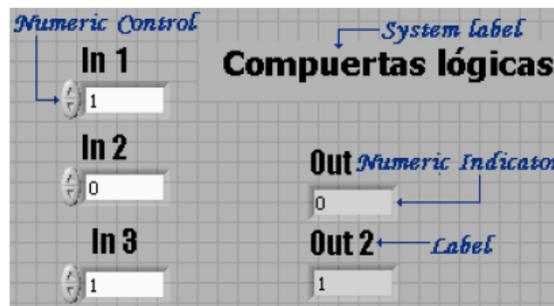


Imagen 2.23. Panel Control. Compuertas Lógicas

Fuente: Extraída de LabVIEW

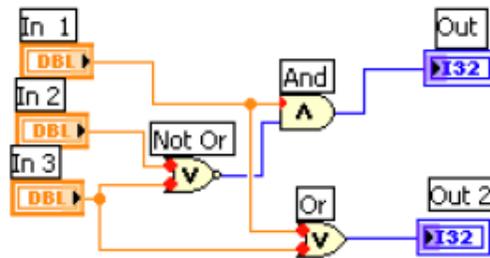


Imagen 2.24. Diagrama de Bloques. Compuertas Lógicas

Fuente: Extraída de LabVIEW

Observación 1: Los controles e indicadores numéricos y etiqueta del sistema (ver imagen a continuación) los encontramos en la paleta de controles. Las compuertas lógicas (ver imagen anterior) las encontramos en la paleta de funciones. Para insertar los controles e indicadores en el panel frontal o diagrama de bloques se selecciona el elemento y se arrastra con el mouse hasta su posición final. Dependiendo de la aplicación procedemos a unir los elementos con ayuda del mouse.

Observación 2: Al presionar el botón de *Run Continuously*  el programa se ejecuta continuamente, y así el usuario puede cambiar los valores de los controles para realizar otras combinaciones lógicas sin parar el programa. Para detenerlo se debe volver a presionar el botón *Run Continuously* o presionando el botón de stop .

Observación 3: Para guardar el programa, nos dirigimos a la barra de herramientas **File»Save** aparecerá un cuadro de diálogo donde se debe especificar la dirección donde se desea guardar el programa, el nombre del programa y la extensión (si no se especifica la extensión LabVIEW automáticamente le agrega la extensión .VI).

2.2.3.7. Tipos de datos.

Existen 3 tipos o clases de datos en LabVIEW:



Imagen 2.25. Esquema de Tipos de Datos

Fuente: Propia

Dependiendo de nuestra aplicación, utilizamos estos datos en el diagrama de bloques existiendo un color para cada uno. (Booleanos: verde claro, Numéricos: azules-naranjos y los Alfanuméricos: rosados). Esto con la finalidad de familiarizarse tempranamente con ellos.

- Boolean (Booleano):

Definidos por enteros de 16 bits. El bit más significativo contiene al valor booleano. Si el bit 15 se pone a 1, el valor del control o indicador es *true* (verdadero) y si se pone a 0, toma el valor *false* (falso). Al dar un clic con el botón derecho del mouse sobre el área de trabajo del panel frontal nos aparecerá la paleta de controles (ver imagen) allí podemos hallar los diferentes controles e indicadores booleanos que posee LabVIEW.

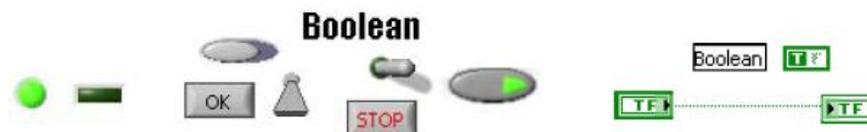


Imagen 2.26. Controles e indicadores booleanos

Fuente Extraída de LabVIEW



Observación: Recordar que al colocar en el panel frontal un control o un indicador automáticamente se agrega el código de estos objetos en el diagrama de bloques.

- Numeric (Numérico):

Los datos numéricos se clasifican en 12 representaciones para los controles e indicadores, señalados por su respectivo color:

- a) Números de tipo entero (**I, Integer**) de 8, 16 y 32 bits
- b) Números de tipo sin signo (**U, Unsigned**) de 8, 16 y 32 bits.
- c) Números de punto flotante 32 (**SGL**), 64 (**DBL**) y 80 (**EXT**) bits
- d) Números de tipo complejos simples (**CSG**), dobles (**CDB**) y extendidos (**CXT**).

El tipo de número se aprecia en la terminal de conexión de los controles o indicadores, pues aparece inscrito, y el color de las conexiones de punto flotante es anaranjadas, mientras que en los enteros y sin signo son azules. Teniendo puesto un control o indicador de tipo numérico en el Diagrama de bloques, hacemos clic sobre él con el botón derecho del mouse y nos dirigimos a “Representation” donde podemos configurar el tipo de dato.

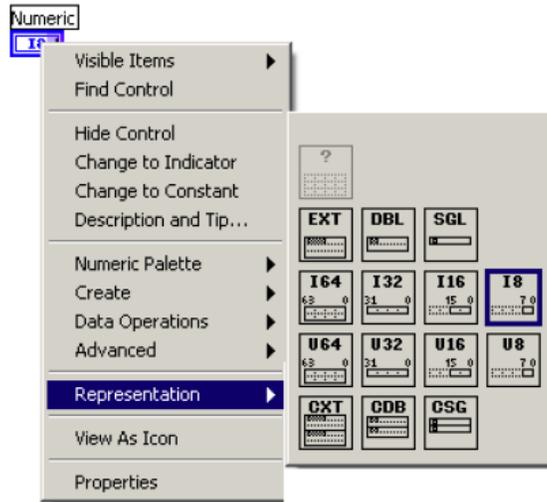


Imagen 2.27. Diferentes representaciones de Datos Numéricos

Fuente: Extraída de LabVIEW

- Strings (Alfanuméricos):

LabVIEW almacena los *strings* como si fuera un array uni-dimensional de bytes enteros (caracteres de 8 bits). Sus principales aplicaciones van definidas a crear mensajes de texto, pasar los datos numéricos a caracteres de datos para instrumentos y luego convertirlos nuevamente de strings a datos numéricos, guardar datos numéricos en el disco, etc. Como mencionamos anteriormente, los datos alfanuméricos se visualizan en el diagrama de bloques con el color rosado (ver imagen derecha) mientras que en el panel frontal, estos datos aparecen como etiquetas, tablas y entradas de texto (ver Imagen izquierda).

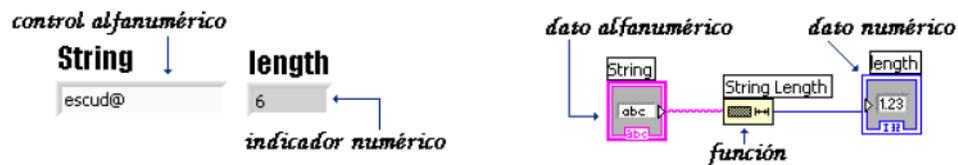


Imagen 2.28. Arreglos n dimensionales

Fuente: Extraída de LabVIEW

En el diagrama de bloques existen muchas funciones donde se ven involucrados los datos alfanuméricos, para verlas hacemos clic con el botón derecho del mouse sobre el área de trabajo para ver la paleta de funciones, nos dirigimos al icono string y se desplegará una ventana con las distintas funciones, como por ejemplo la más simple, *string length* que indica la longitud de la cadena de datos alfanuméricos.

Por ejemplo si en el panel frontal, colocamos un control alfanumérico y escribimos alguna “palabra”, al arrancar el VI (Run) el indicador numérico nos da la respuesta a la función *length*.



Dato alfanumérico Panel de Control.

Dato alfanumérico Diagrama de Bloques.

Imagen 2.29. Datos alfanuméricos

Fuente: Extraída de LabVIEW

Es posible realizar atractivos VI donde participen los tres tipos de datos para la utilización de una aplicación determinada. Por ejemplo, en la imagen se muestra un sencillo sistema donde se suman dos números, en donde si se sobrepasa el valor 50 la aplicación nos entrega una señal de alerta mediante el encendido de un led (*boolean*) y una palabra de advertencia (*string*).

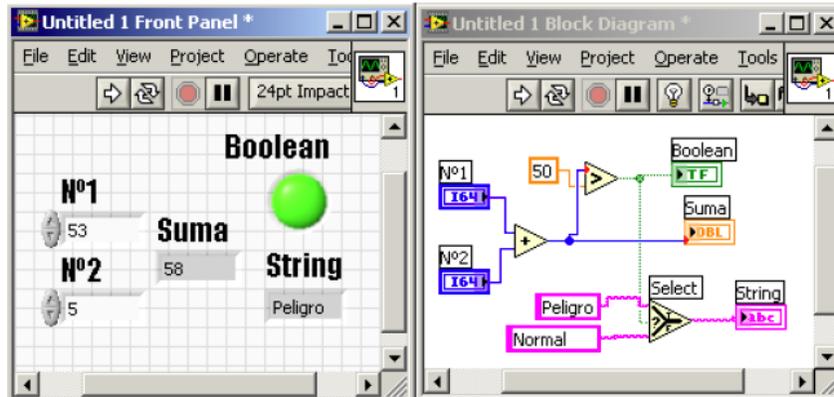


Imagen 2.30. Suma de dos números

Fuente: Extraída de LabVIEW

Observación: El selector, el sumador y el comparador que se visualizan en el ejemplo anterior se encuentran en la paleta de funciones del diagrama de bloques. Cabe mencionar que se pueden combinar éstos tipos de datos en un mismo indicador (cadena resultante), por ejemplo con el elemento *Concatenate Strings*  es posible concatenar datos numéricos y alfanuméricos.

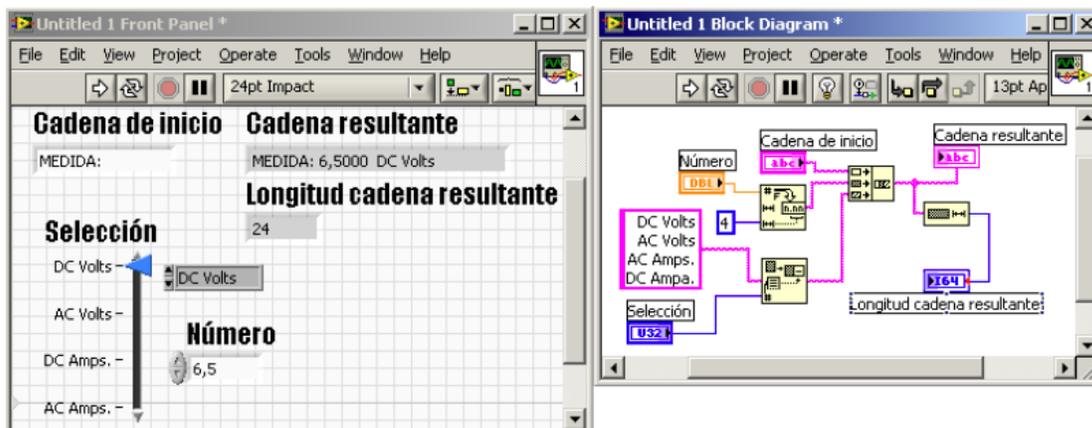


Imagen 2.31. Concatenados de datos numéricos y alfanuméricos

Fuente: Extraída de LabVIEW



Línea de selección (*Pick line*), nos permite visualizar el dato seleccionado en el panel frontal del instrumento.

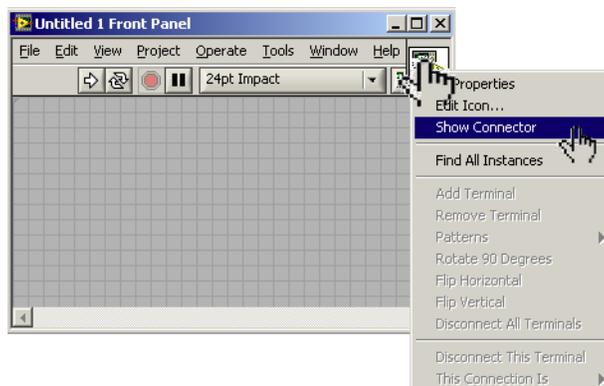


Número a String Fraccionario (*Number to Fractional String*), nos permite convertir el dato numérico en un datos alfanumérico para poderlo unir a la concatenación.

2.2.3.8. Sub rutina o SubVI.

Dentro de la programación de LabVIEW existe una utilización de los VI bastante interesante de conocer, que facilita el desarrollo de aplicaciones. Cuando uno finaliza y guarda un archivo con extensión *.vi*, éste puede ser llamado en el diagrama de bloques desde otro VI conociéndose también como *SubVI* que corresponde a una subrutina como en los demás lenguajes tradicionales. Para usar un VI como subVI, se requiere construir un panel de conectores. Este panel, determina las entradas y salidas de los controles e indicadores.

Para construir un panel de conectores, nos dirigimos al icono en la parte superior izquierda del panel frontal y hacemos clic con el botón derecho del mouse, enseguida se muestra un menú y hacemos clic en *Show Connector* y el icono cambiará estéticamente como se muestra en la imagen 2.32.



Show Connector



Icono SC

Imagen 2.32. Panel de Conectores

Fuente: Extraída de LabVIEW

Cada pequeño cuadrado o rectángulo representa un terminal (salida o entrada) siendo el número máximo de 28 terminales posibles de añadir. Para asignar un control a uno de los terminales del bloque de conectores hacemos clic sobre el terminal que se desea configurar, el puntero cambia a la forma de un carrete y el terminal tiene el color negro, ahora si nos posicionamos sobre el control que se desea asignar el terminal, al ser asignado, el color del terminal cambiará de color dependiendo del tipo de dato asignado.

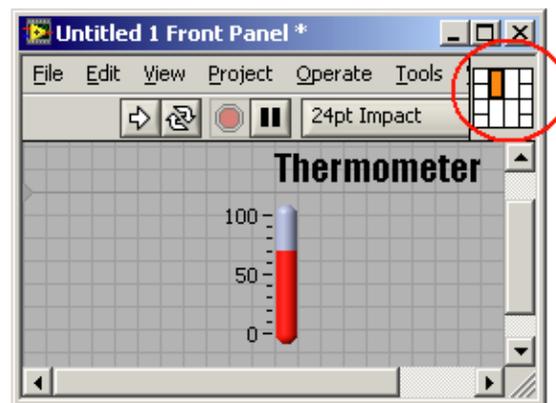


Imagen 2.33. Terminal asignado Icono SC

Fuente: Extraída de LabVIEW

Este mismo icono se puede editar en la parte superior izquierda del diagrama de bloques (*Edit Icon*) a gusto del usuario. Una vez configurado nuestro control de terminales dependiendo de los requerimientos de la aplicación, guardamos el VI en nuestro PC.

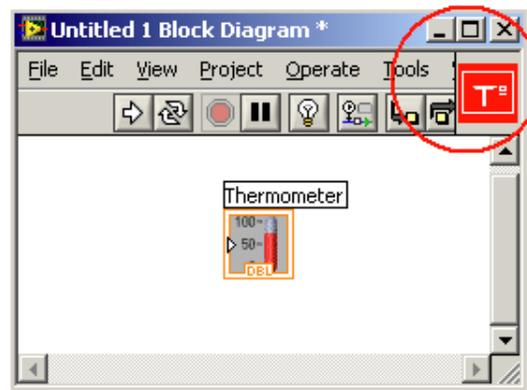


Imagen 2.34. Edit Icono SC

Fuente: Extraída de LabVIEW

Este VI puede ser llamado desde la paleta de funciones del diagrama de bloques pulsando donde diga “*Select VI*” y mostrando el icono con sus respectivos terminales (pasando el mouse por encima del icono) como se muestra en la imagen 2.35.

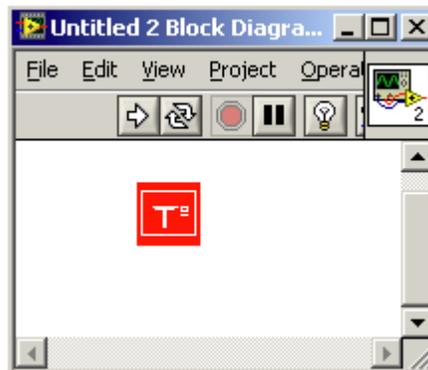


Imagen 2.35. Selección de VI.

Fuente: Extraída de LabVIEW

2.2.3.9. Estructuras de control.

A la hora de programar, muchas veces es necesario ejecutar un mismo conjunto de sentencias con un número determinado de veces, o que éstas se repitan mientras se cumplan



ciertas condiciones. LabVIEW dispone principalmente de cinco estructuras fácilmente diferenciables por su apariencia y disponibles en la opción *Structures* que aparece en la paleta de funciones del diagrama de bloques. Estas estructuras van ejecutando automáticamente lo que está programado en su interior, sin embargo, cada estructura ejecuta su sub diagrama de acuerdo a las características que gobiernan su funcionamiento.

En la *paleta de funciones* la primera opción es la de las *estructuras*. Éstas controlan el flujo del programa, bien sea mediante la secuenciación de acciones, ejecución de bucles, etc.

Las estructuras se comportan como cualquier otro nodo en el diagrama de bloques, ejecutando automáticamente lo que está programada en su interior una vez tiene disponibles los datos de entrada, y una vez ejecutadas las instrucciones requeridas, suministran los correspondientes valores a los cables unidos a sus salidas. Sin embargo, cada estructura ejecuta su *subdiagrama* de acuerdo con las reglas específicas que rigen su comportamiento, y que se especifican a continuación.

Un *subdiagrama* es una colección de nodos, cables y terminales situados en el anterior del rectángulo que constituye la estructura.

Tabla 2.4. Estructuras de un VI

Estructura (Structure)	Nº sub diagramas
While Loop	1
For Loop	1
Case Structure	Múltiples
Sequence Structure	Múltiples
Event Structure	Múltiples

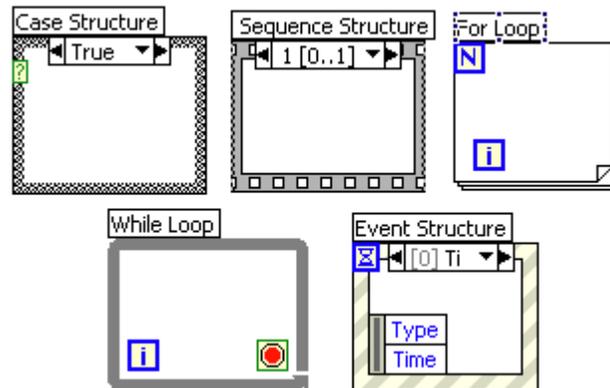


Imagen 2.36. Estructuras de un VI.

Fuente: Extraída de LabVIEW

El *For Loop* y el *While Loop* únicamente tienen un subdiagrama. El *Case Structure* y el *Sequence Structure*, sin embargo, pueden tener múltiples subdiagramas, superpuestos como si se tratara de cartas en una baraja, por lo que en el diagrama de bloques únicamente será posible visualizar al tiempo uno de ellos. Los subdiagramas se construyen del mismo modo que el resto del programa. Las siguientes estructuras se hallan disponibles en el lenguaje G.

Case Structure

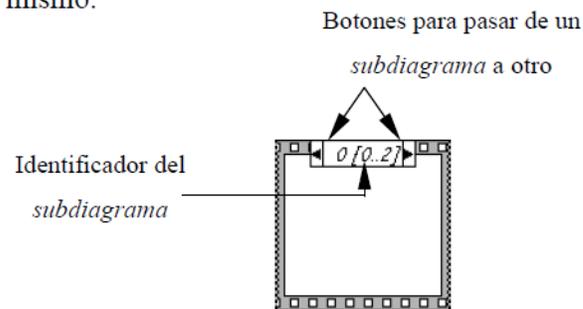
 Al igual que otras estructuras posee varios *subdiagramas*, que se superponen como si de una baraja de cartas se tratara. En la parte superior del subdiagrama aparece el identificador del que se está representando en pantalla. A ambos lados de este identificador aparecen unas flechas que permiten pasar de un *subdiagrama* a otro.

En este caso el identificador es un valor que selecciona el subdiagrama que se debe ejecutar en cada momento.

La estructura *Case* tiene al menos dos *subdiagramas* (*True* y *False*). Únicamente se ejecutará el contenido de uno de ellos, dependiendo del valor de lo que se conecte al *selector*.

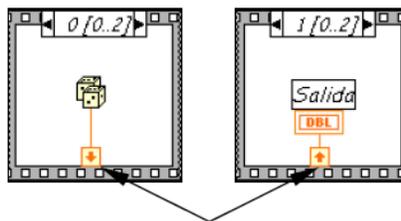
Sequence Structure

 De nuevo, este tipo de estructuras presenta varios *subdiagramas*, superpuestos como en una baraja de cartas, de modo que únicamente se puede visualizar una en pantalla. También poseen un identificador del *subdiagrama* mostrado en su parte superior, con posibilidad de avanzar o retroceder a otros *subdiagramas* gracias a las flechas situadas a ambos lados del mismo.



Esta estructura secuencia la ejecución del programa. Primero ejecutará el *subdiagrama* de la hoja (*frame*) nº0, después el de la nº 1, y así sucesivamente.

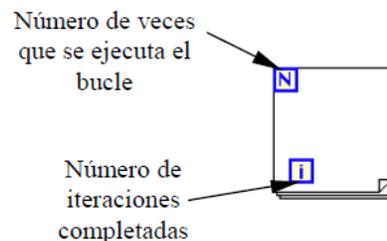
Para pasar datos de una hoja a otra se pulsará el botón derecho del ratón sobre el borde de la estructura, seleccionando la opción *Add sequence local*.



Sequence local: paso de un dato de la *frame* 0 a la 1

For Loop

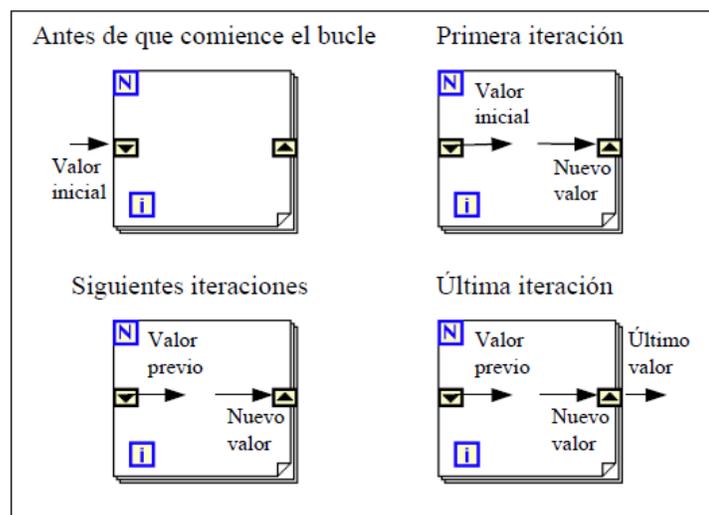
 Es el equivalente al bucle *for* en los lenguajes de programación convencionales. Ejecuta el código dispuesto en su interior un número determinado de veces.



Ejecutar el bucle *for* es equivalente al siguiente fragmento de código:

```
For i = 0 to N - 1  
Ejecutar el subdiagrama del interior del Bucle
```

Para pasar valores de una iteración a otra se emplean los llamados *shift registers*. Para crear uno, se pulsará el botón derecho del ratón mientras éste se halla situado sobre el borde del bucle, seleccionando la opción *Add Shift Register*. El *shift register* consta de dos terminales, situados en los bordes laterales del bloque. El terminal izquierdo almacena el valor obtenido en la iteración anterior. El terminal derecho guardará el dato correspondiente a la iteración en ejecución. dicho dato aparecerá, por tanto, en el terminal izquierdo durante la iteración posterior.

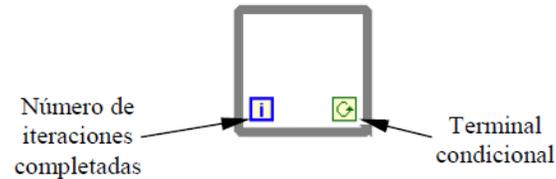


Se puede configurar un *shift register* para memorizar valores de varias iteraciones previas. Para ello, con el ratón situado sobre el terminal izquierdo del *shift register*, se pulsará el botón derecho, seleccionando a continuación la opción *Add Element*.

While Loop



Es el equivalente al bucle *while* empleado en los lenguajes convencionales de programación. Su funcionamiento es similar al del bucle *for*.



El bucle *while* es equivalente al código siguiente:

```
Do
  Se ejecuta lo que hay en el interior del bloque
while terminal condicional is true
```

El programa comprueba el valor de lo que se halle conectado al terminal condicional al finalizar el bucle. Por lo tanto, el bucle siempre se ejecuta al menos una vez.

Con esta estructura también se pueden emplear los *shift registers* para tener disponibles los datos obtenidos en iteraciones anteriores (es decir, para memorizar valores obtenidos). su empleo es análogo al de los bucles *for*, por lo que omitirá su explicación.

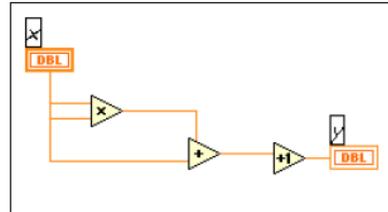
Formula Node



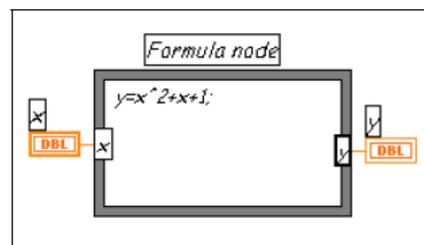
La estructura denominada *Formula Node* se emplea para introducir en el diagrama de bloques fórmulas de un modo directo. Resulta de gran utilidad cuando la ecuación tiene muchas variables o es relativamente compleja. Por ejemplo, se desea implementar la ecuación:

$$y = x^2 + x + 1$$

Empleando bloques pertenecientes al *lenguaje G* quedaría:



Si se utiliza la *formula node*, se obtiene:



Para definir una fórmula mediante esta estructura, se actuará del siguiente modo:

- En primer lugar, se deben definir las variables de entrada y las de salida. Para ello, se pulsa con el botón derecho del ratón sobre el borde de la *formula node*. A continuación se seleccionará *Add Input* o *Add Output*, según se trate de una entrada o una salida, respectivamente. Aparecerá un rectángulo, en el que se debe escribir el nombre de la variable (se distingue entre mayúsculas y minúsculas). Todas las variables que se empleen deben estar declaradas como entradas o salidas. Las que se empleen como variables intermedias se declararán como salidas, aunque posteriormente no se unan a ningún bloque posterior.
- Una vez definidas las variables a emplear, se escribirán la o las fórmulas en el interior del recuadro (para ello se emplea la *labeling tool*). Cada fórmula debe finalizar con un “;”.
- Los operadores y funciones que se pueden emplear se explican en la ayuda de LabVIEW, y son los que se muestran a continuación:



Operadores:

asignación =

condicional ?:

OR lógico ||

AND lógico &&

relacionales == != > < >= <=

aritméticos + - * / ^

Funciones:

abs acos acosh asin asinh atan atanh ceil cos cosh

cot csc exp expm1 floor getexp getman int intrz ln

lnp1 log log2 max min mod rand rem sec sgn sin

sinc sinh sqrt tan tanh

La sintaxis de una expresión incondicional es la siguiente:

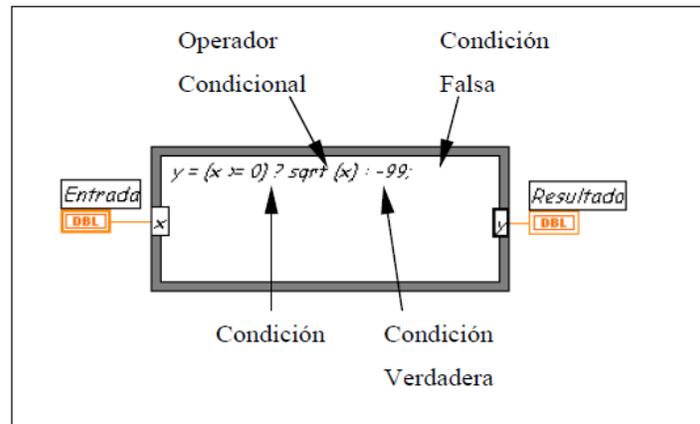
$$\langle \text{expresión condicional} \rangle ? \langle \text{texpresión} \rangle : \langle \text{fexpresión} \rangle$$

Si el valor lógico de la *expresión condicional* es *true* se ejecutará *texpresión*. Si, por el contrario, fuese *false*, lo que se aplicará será *fexpresión*

Como ejemplo considérese el siguiente fragmento de código:

```
if (x >= 0) then
y = sqrt (x)
else
y = -99
end if
```

Se puede implementar este fragmento de código empleando un *formula node*, tal y como se muestra en la siguiente figura:



Observación: En la misma paleta donde encontramos el *Formule Node*, se encuentra otro icono llamado *MathScript Node* que es similar al anterior, sólo que este trabaja más con la sintaxis de MATLAB®

Ejemplo:

En el siguiente ejemplo, realizamos un VI donde obtendremos el valor de la caída de voltaje y la potencia demandada por un resistor. Los valores de entrada son la corriente y el valor de la resistencia.

En nuestro panel frontal vamos a colocar dos controles numéricos y dos indicadores numéricos. Dentro del Nodo de Fórmula vamos a escribir la fórmula para calcular la caída de voltaje conociendo la corriente (I) y el valor de la resistencia (R) y la fórmula de consumo de potencia (W), separamos las dos fórmulas por medio de un punto y coma tal como se muestra en la imagen.



Imagen 2.37. Panel Frontal de un Control Numérico

Fuente: Extraída de LabVIEW

Observación: Después de cada formula debes colocar un punto y coma.

Para colocar las entradas y salidas del nodo de fórmula, uno debe posicionarse sobre el marco de la estructura y dar un clic derecho con el mouse, se desplegará un sub menú, se elige la opción **Add Input** en el recuadro que aparece, posteriormente se coloca el nombre de la variable tal como hemos querido llamarla en la fórmula. (Se hace lo mismo para las salidas, claro la opción será **Add Output**). Dos entradas una para corriente (I) y otra para el valor de la resistencia (R) y dos salidas una para el valor de caída de voltaje (V) y la otra para el valor de potencia (P), ta como se aprecia en la imagen.

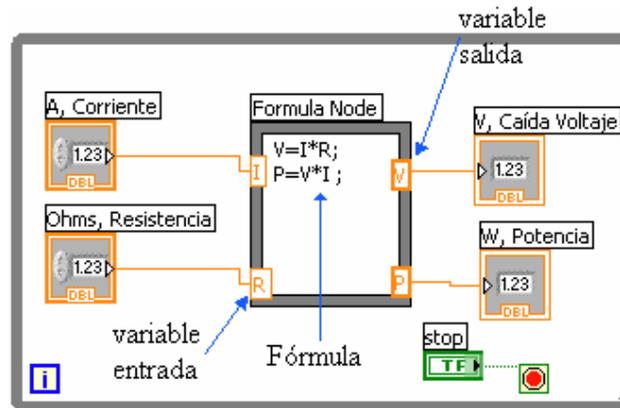


Imagen 2.38. Ejemplo práctico nodo fórmula (diagrama de bloques)

Fuente: Extraída de LabVIEW

Observación: No existe límite para el número de variables o de fórmulas y nunca podrá haber 2 entradas o 2 salidas con el mismo nombre, aunque una salida si podrá tener el mismo nombre que una entrada. Todas las variables de salida deberán estar asignadas a una fórmula por lo menos una vez.

Event Structure: Este tipo de estructuras presenta varios sub diagramas, los cuales se ejecutan exactamente cuando la estructura se ejecuta. La estructura de Evento espera hasta que sucede algún evento. Se pueden agregar nuevos eventos haciendo clic derecho en el borde de la estructura evento y posicionarse en *Add Event Case*. La implementación del sistema monitoreo para la Pesquera el Golfo Corral está programada bajo *Event Structure*.

2.2.3.10. Registro de Desplazamiento.

Se les conoce también como *Shift Register* y son variables locales (En las hojas posteriores se explicará lo que es una variable local). Al hacer clic con el botón derecho del mouse en el borde de la estructura *While* o *For Loop* aparece un menú con la opción *Add Shift Register*.

El propósito del Shift Register es el almacenamiento de datos, por ejemplo, el terminal de la derecha  va almacenando el valor final de la repetición o iteración. Cuando sucede una nueva repetición, este valor se mueve al terminal izquierdo  y el nuevo valor queda en el terminal derecho.

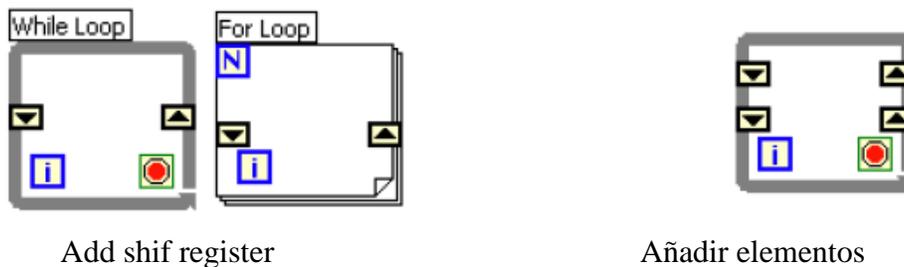


Imagen 2.39. Almacenamiento de Datos

Fuente: Extraída de LabVIEW

Con la opción *Add Shift Register* se pueden ir añadiendo elementos, que quiere decir esto, que cuanto más terminales tengamos en el lado izquierdo más valores de iteraciones anteriores podremos almacenar. (Ver imagen anterior).

Ejemplo:

En el siguiente ejemplo, se diseña un programa con la ayuda de un *For While* que cuenta hasta 10 segundos, al transcurrir los 10 segundos este se detiene y muestra un mensaje que indica que se han cumplido los 10 segundos.

Panel Frontal:

- a) For While (Izquierda)
- b) Mensaje (derecha)



Imagen 2.40. Ejemplo de un Panel Frontal

Fuente: Extraída de LabVIEW

Diagrama de Bloques

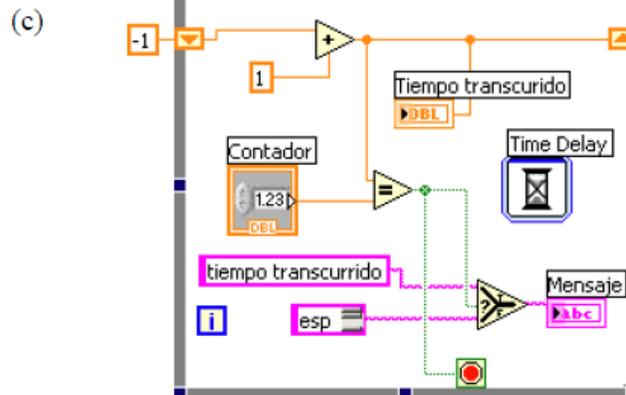


Imagen 2.41. Diagrama de Bloques (Ejemplo)

Fuente: Extraída de LabVIEW

2.2.3.11. Arreglos y Clúster.

- **Arrays:** Arreglos o *Arrays* es un grupo de elementos de datos del mismo tipo, se compone de elementos y dimensiones. Se entiende como elemento a los datos que



conforman un arreglo y dimensión como la longitud, altura o profundidad del mismo.

- **Indexado:** Para colocar un elemento en particular en un arreglo este requiere un índice por dimensión. En LabVIEW®, el indexado permite navegar a través de un arreglo y además recuperar elementos, filas, columnas, y páginas de un arreglo en el diagrama de bloques. Un ejemplo de un arreglo es una señal de onda representada como un arreglo numérico en el cual cada elemento es el valor de voltaje de sucesivos intervalos de tiempo, pero este ejemplo es de 1D (una dimensión). Un arreglo de 2D guarda los elementos en un tipo de rejilla (tabla). Esto requiere un índice de columna y un índice de fila para localizar cada elemento, ambos índices de localización comienzan en cero. Por ejemplo, un tablero de ajedrez tiene ocho columnas y ocho filas para formar un total de 64 posiciones. Cada posición está vacía o tiene una pieza de ajedrez. Se puede representar un tablero de ajedrez con un arreglo de texto de 2D.

- **Restricciones de los arreglos:** Se puede crear un arreglo casi de cualquier tipo de datos, con las siguientes excepciones:
 - No se pueden crear arreglos de arreglos. Independientemente, se pueden utilizar un arreglo multidimensional o usar la función *Build Cluster Array* para crear un arreglo de cluster donde cada cluster contiene uno o más arreglos.

 - No se puede crear un arreglo de gráficos, porque un gráfico es un tipo de datos del arreglo y un arreglo no puede contener otro arreglo. Sin embargo, se puede tener un arreglo de gráficos si el gráfico está dentro de un cluster.

 - No se puede crear arreglos de charts.

En la paleta de funciones del diagrama de bloques encontramos diferentes tipos de arreglos para crear y manipular (ver Imagen) como por ejemplo, extraer un elemento individual, dividir, insertar, eliminar o reemplazar un elemento dentro de un arreglo.

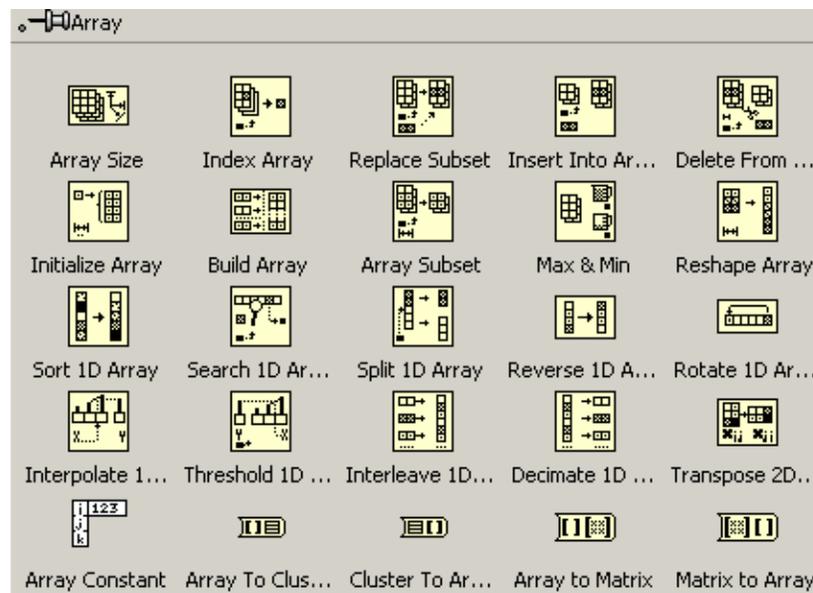


Imagen 2.42. Arreglos de diagrama de bloques

Fuente: Extraída de LabVIEW

- **Clúster:** Los clusters agrupan diferentes tipos de elementos o datos, como un bulto de alambres, es decir, como en un cable del teléfono dónde cada alambre en el cable representa un elemento diferente del cluster. Un cluster es similar a un registro o una estructura en los lenguajes de programación basados en texto. La agrupación de los datos en los clusters elimina el desorden de los alambres en el diagrama de bloques y reducen el número de conectores o terminales que un subVI necesita. El panel de conectores tiene la capacidad, a lo sumo, de 28 términos. Si su panel frontal contiene más de 28 controles e indicadores que se quieren usar en su programa, lo que se hace es agrupar algunos de ellos en un cluster y asignar el cluster a uno de las terminales del panel de conectores.

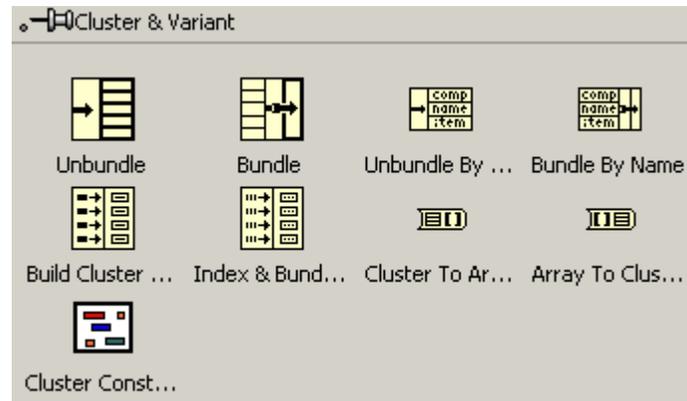


Imagen 2.43. Cluster de diagrama de bloque

Fuente: Extraída de LabVIEW

Aunque un cluster y un arreglo de elementos en ambos existe un ordenamiento, se debe desagrupar todos los elementos del cluster en lugar de escribir un índice de un elemento a la vez. Se puede también usar la función *Unbundle By name* y así acceder a los elementos del cluster específicos.

Los cluster también difieren de los arreglos en que ellos son de un tamaño fijo. Como un arreglo, un cluster puede ser un control o un indicador. Un cluster no puede contener una mezcla de controles e indicadores. La mayoría de los cluster en el diagrama de bloques se caracterizan por que el alambre y el icono de los datos son de color rosado.

Los cluster de números, se caracterizan por un alambre y un icono de datos de color café. Se pueden alambrear los cluster numéricos de color café a las funciones numéricas, como a la función *Add*, *Square Root*, para realizar la misma función simultáneamente en todos los elementos del cluster.

2.2.3.12. Indicadores Gráficos: Graph y Charts.

Existen variados tipos de indicadores gráficos en LabVIEW, se destacan principalmente: Waveform Graph, Waveform Chart, Intensity Graph, Intensity Chart, XY Graph, 3D Graphs y Digital Waveform Graph.

- **Waveform Graph – Chart:** Después de adquirir o generar datos, utilizar un Graph (gráfico) o un Chart es una muy idea para desplegar datos de una forma gráfica.

Los *Graph* y los *Chart* se diferencian entre sí en la manera de mostrar los datos. Un VI con un *Graph* recoge generalmente los datos en un *Array* (arreglo) y después traza los datos al gráfico. Cuando se trazan los datos, desecha los datos previamente trazados y muestra solamente los nuevos datos. Se utiliza un *Graph* en los procesos rápidos que están continuamente recogiendo datos. En cambio, un *Chart* añade nuevos puntos de referencias a esos puntos en su pantalla para crear una especie de historial. Se utiliza un *Chart* en los procesos lentos en los cuales solamente algunos puntos de referencias por segundo, se agregan al diagrama.

A pesar que el *Waveform Graph* y el *Waveform Chart* presentan diferencias, también poseen muchas opciones iguales en donde se accede mediante un menú contextual. Por ejemplo, ambos soportan múltiples escalas en x e y, se puede personalizar su apariencia.

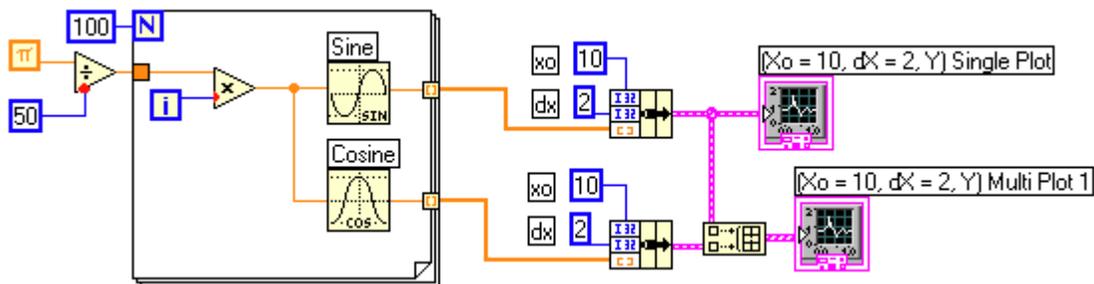


Imagen 2.44. Diagrama de bloques

Fuente: Extraída de LabVIEW

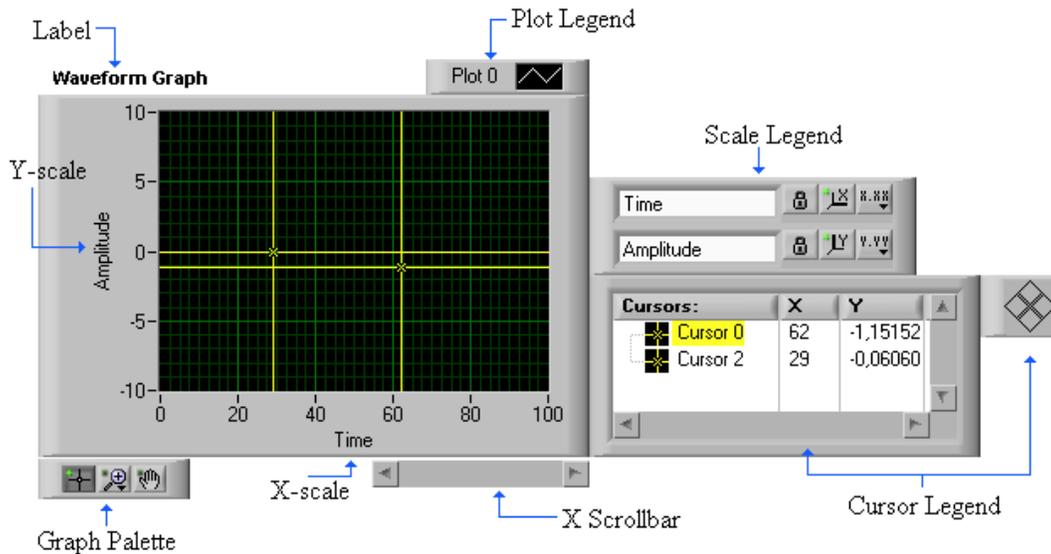


Imagen 2.45. Diagrama de medición

Fuente: Extraída de LabVIEW

El Waveform Graph muestra unos o más diagramas de medición uniformemente muestreados. El *Waveform Graph* traza solamente funciones valoradas, como por ejemplo, $y = f(x)$, con los puntos distribuidos uniformemente a lo largo del eje x, tal como formas de onda de tiempo. La Imagen muestra un ejemplo de un *Waveform Graph* simulando ondas seno y coseno respectivamente.

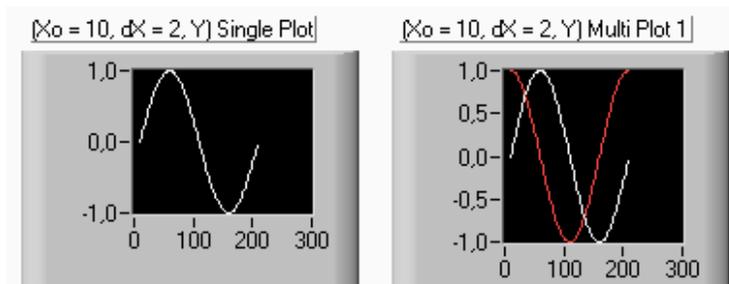


Imagen 2.46. Muestra de una Grafica

Fuente: Extraída de LabVIEW

- **Intensity Graph – Chart:** Despliega la información en 3D en una gráfica 2D mediante colores para desplegar valores de tercera dimensión. Son útiles para representar datos de superficie, donde el color puede representar altitud, humedad, temperatura, entre otras variables.
- **Digital Waveform Graph:** Despliega datos como pulsos o grupos de líneas digitales, es decir, señales digitales. Son útiles para diagramas de tiempo.

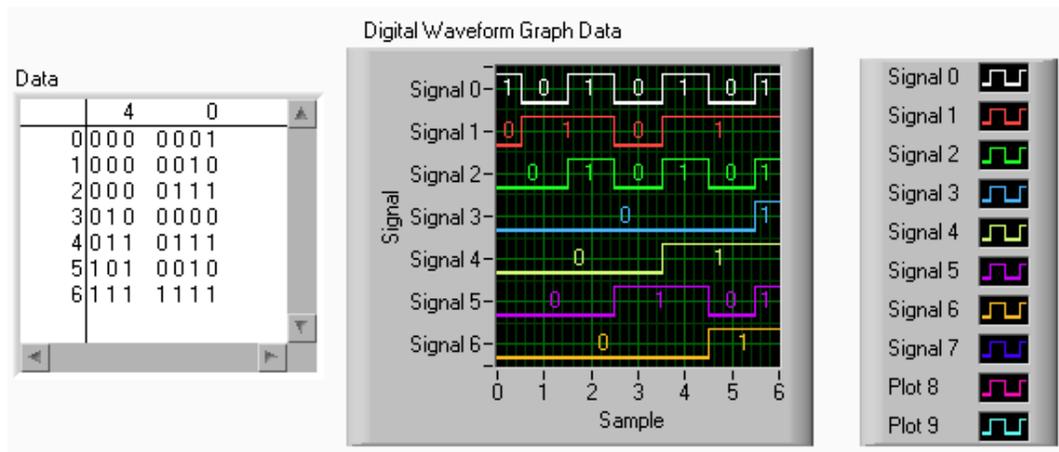


Imagen 2.47 Digital Waveform Graph (panel frontal)

Fuente: Propia

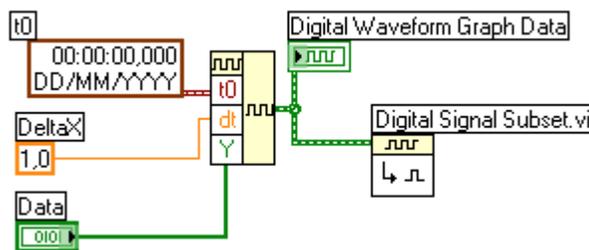


Imagen 2.48. Gráfico Digital Waveform Graph (diagrama de bloques)

Fuente: Extraída de LabVIEW

- **XY Graphs:** Despliega los datos adquiridos en una tasa no constante, es decir, el mismo usuario puede determinar la separación.

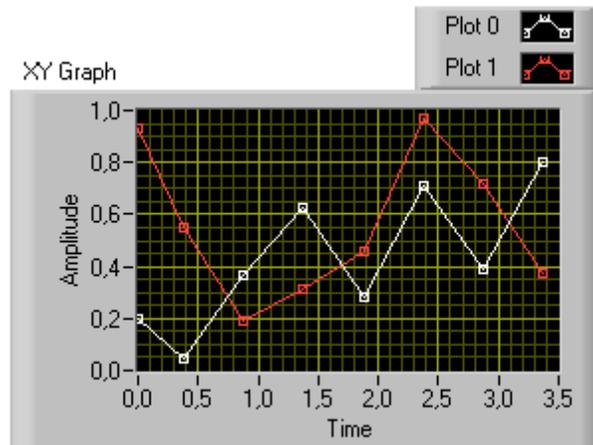


Imagen 2.49. Grafica en plano XY

Fuente: Extraída de LabVIEW

- **3D Graph:** Despliega datos 3D en un gráfico 3D mediante un objeto ActiveX en el panel frontal.

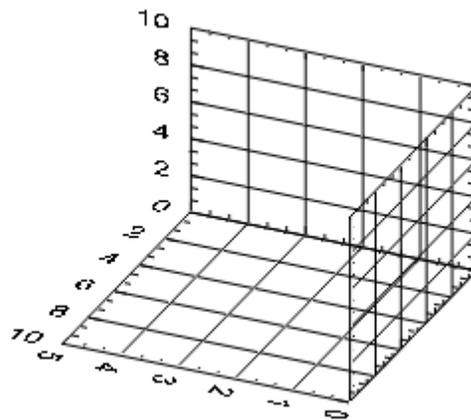


Imagen 2.50. Gráfica en 3D

Fuente: Extraída de LabVIEW



2.2.3.13. Archivos I/O.

Los ficheros de entrada y salida (I/O) transfieren los datos de LabVIEW a archivos y viceversa. En la paleta de funciones del diagrama de bloques *File I/O* podemos encontrar bloques que nos permiten, abrir y cerrar archivos de datos, leer y escribir datos a archivos, leer y escribir archivos de hojas de cálculo, mover y renombrar archivos y directorios, cambiar las características de los archivos, crear, modificar y leer una configuración de archivo. Se pueden leer o escribir datos de tres tipos de archivos diferentes: texto, binarios y datalog. Por ejemplo, si se quiere guardar un archivo de datos en Microsoft Excel usaremos el formato de texto debido a que este es el más común en este tipo de trabajo.

Guardando un Archivo

Para entender rápidamente este proceso, crearemos un ejemplo donde analizaremos la amplitud de una señal senoidal, de tal manera que si la amplitud sobrepasa o está por debajo de un porcentaje establecido por el usuario guarde dicho valor en un archivo de texto.

El programa muestra la amplitud actual de la señal senoidal, y dice cuantos datos se han guardado en el archivo.

Ejemplo

Nuestro Panel Frontal lo dejamos de la siguiente manera, claro que es opcional, pues dependerá del usuario establecer una estética que cumpla con sus necesidades.

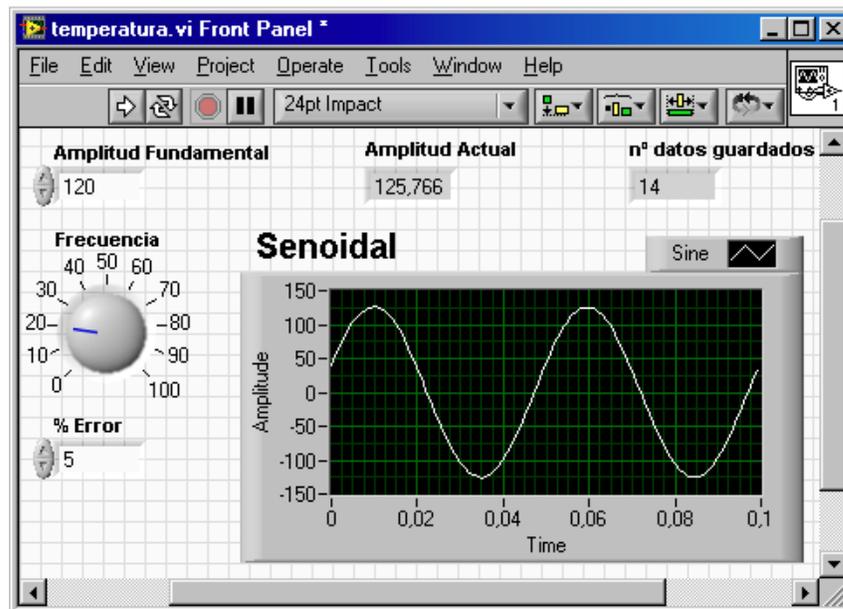


Imagen 2.51. Ejemplo práctico con Waveform Graph

Fuente: Propia

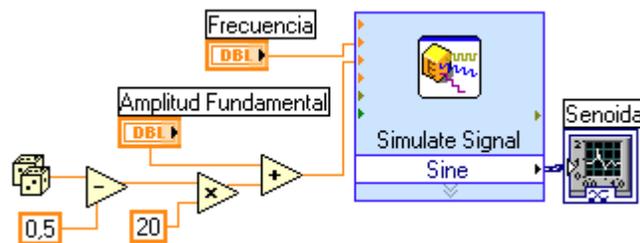


Imagen 2.52. Generación de números aleatorios

Fuente: Extraída de LabVIEW

Para analizar y conocer el nuevo valor de amplitud que se está generando utilizamos la función de análisis que se encuentra en la paleta de funciones, *Express* » *Signal Analysis* » *Tone*, al colocarla se genera una ventana en la cual configuramos el VI, en nuestro caso elegimos únicamente la opción *Amplitude*.

Ahora, como ya conocemos el valor de la amplitud instantánea, de la señal senoidal verificamos si se encuentra dentro del rango establecido (ver imagen).

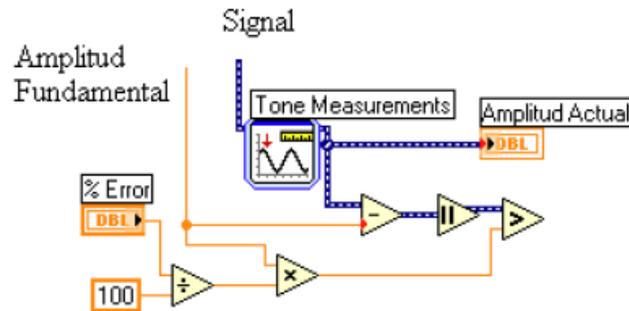


Imagen 2.53. Señal de una Onda

Fuente: Extraída de LabVIEW

Restamos el valor actual menos el valor de amplitud fundamental, y el valor absoluto de esta operación lo comparamos con el porcentaje establecido. En el caso en que se sobrepase o esté por debajo del porcentaje de error establecido vamos a guardar el valor (ver imagen 2.54).

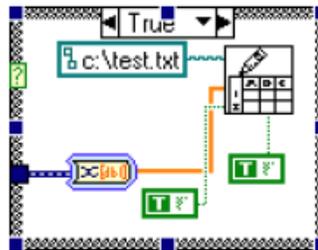


Imagen 2.54

Fuente: Extraída de LabVIEW

Para guardar el archivo vamos a utilizar el VI, *Write to Spreadsheet File.vi*, que aparece en la paleta de funciones *File I/O*, para nuestro caso únicamente hemos colocado

una constante en la terminal *file path*, esta constante es una constante de ruta de archivo, aquí colocamos la dirección del lugar donde queremos que el archivo se guarde.

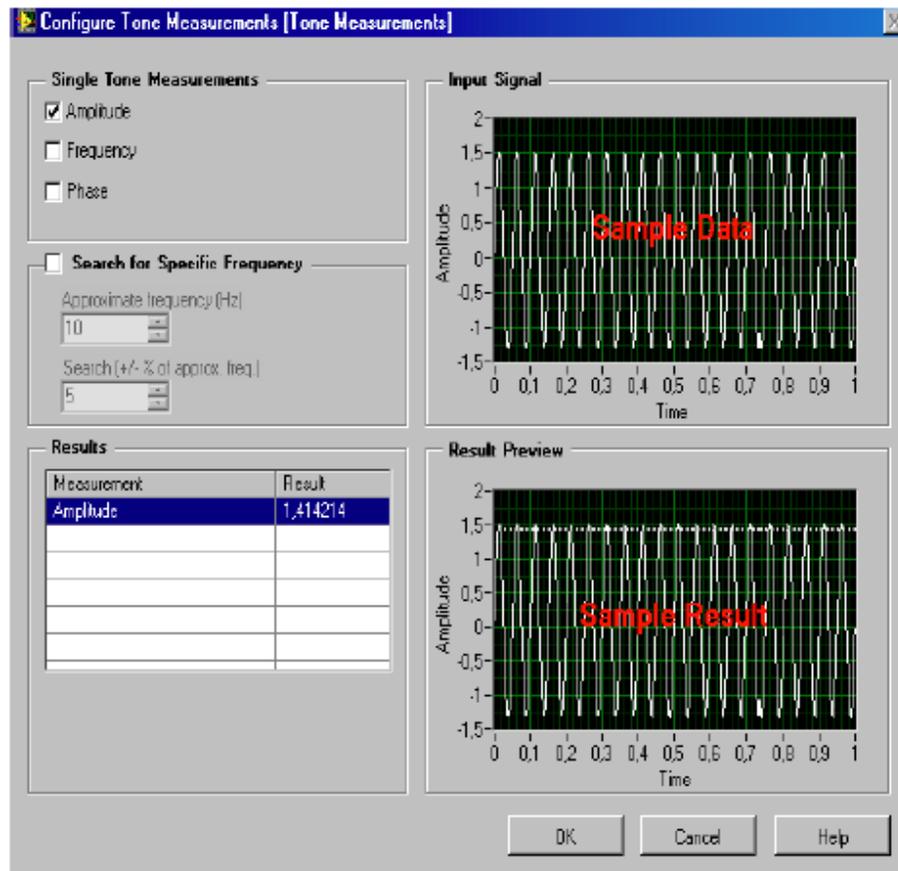


Imagen 2.55. Resultado

Fuente: Extraída de LabVIEW

Como queremos que cada vez que se genere un dato que este fuera del rango se agregue al archivo que ya hemos creado el terminal *append to file*, tiene que tener una constante *True* para que vaya agregando los valores, La terminal *Transpose* se utiliza para generar el arreglo como fila (default) o columna, colocamos una constante *True* para que se despliegue en forma de columna.

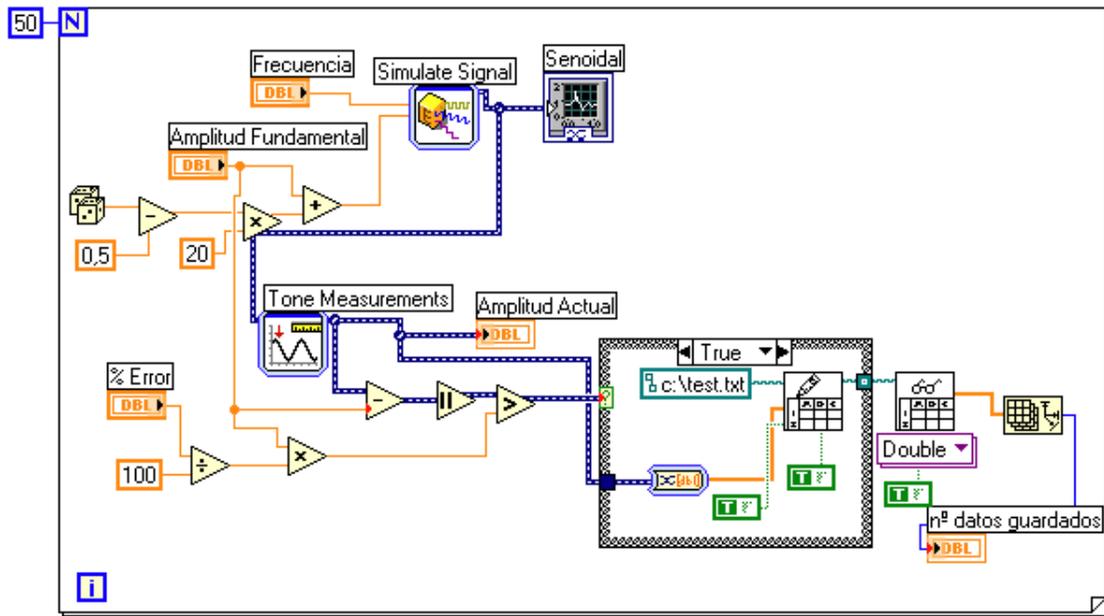


Imagen 2.56.

Fuente: Extraída de LabVIEW

2.2.3.14. Variables Locales y Globales.

En LabVIEW se pueden leer o escribir datos en el panel frontal utilizando terminales en el diagrama de bloques. Sin embargo, un objeto en el panel frontal tiene sólo un terminal en el diagrama de bloques y su aplicación podría necesitar acceder a los datos del terminal en más de una vez.

Hablar de variables locales o globales en LabVIEW es algo no tan fácil de entender, pues son conceptos avanzados y una mala utilización de estas puede provocar un comportamiento inesperado en los VI.

¿Cuándo utilizar una Variable Local?



Cuando se pretende acceder a objetos del panel frontal en más de un lugar dentro del mismo VI y pasar la información (datos) entre las estructuras de bloque donde **No** se puede conectar un cable. Trabajando con una variable local, se puede leer o escribir desde un indicador o control en el panel frontal. En efecto, se puede acceder a un objeto del panel frontal como entrada y como salida.

¿Cuándo utilizar una Variable Global?

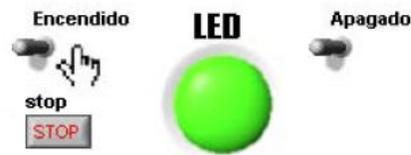
Cuando se pretende acceder y pasar datos entre varios VI que se ejecutan al mismo tiempo. Cuando en LabVIEW se crea una variable global, automáticamente este crea un VI especial, el cual contiene un panel frontal pero **No** un diagrama de bloques, es decir, el panel frontal es un contenedor de varios VIs que pueden acceder datos.

En resumen, la diferencia entre una variable local y una global, es que con la variable local, solo puedes intercambiar datos dentro de un mismo VI. En cambio con la variable global puedes intercambiar datos entre diferentes VIs. A continuación daremos solución a un sencillo ejemplo utilizando variables locales y globales, que consiste en el encendido de un LED por medio de un pulsador y sea apagado por otro pulsador.

Ejemplo

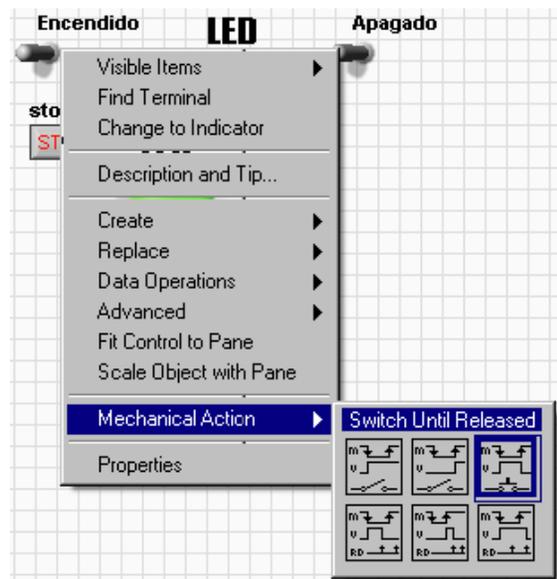
Como mencionábamos anteriormente, este problema lo vamos a resolver de dos maneras diferentes, la primera utilizando variables locales, y la otra utilizando variables globales.

- A) Utilizando variables Locales: Lo primero a realizar es ir a nuestro panel frontal y colocar 2 pulsadores y un led, asegurándonos que los botones (pulsadores) tengan una acción mecánica al igual que un pulsador real (ver imagen 2.57).

Panel Frontal**Imagen 2.57.** Pulsadores

Fuente: Extraída de LabVIEW

Para seleccionar una acción mecánica del pulsador, hacemos clic derecho sobre el pulsador y nos dirigimos a *Mechanical Action*

**Imagen 2.58.** Acción Mecánica

Fuente Extraída de LabVIEW

En el diagrama de bloques, vamos a crear una variable local del indicador tipo LED, para crear la variable local, hacemos clic derecho sobre el LED, se desplegará un menú, elegimos la opción, *Create » Local Variable*. Posteriormente, colocamos 2 variables locales del indicador LED, una dentro de cada *Case*, tal como se muestra en la Imagen 5.62, ahora conectamos un pulsador a cada *Case*, en el

Case en el que se ha conectado el pulsador de encendido se conecta una constante verdadera a la variable local en el caso True, en el caso falso no se hace nada. En el otro Case donde hemos conectado el pulsador de apagado conectamos una constante falsa a la variable local, en el caso verdadero del case, en el caso falso no hacemos nada.

Diagrama de Bloques

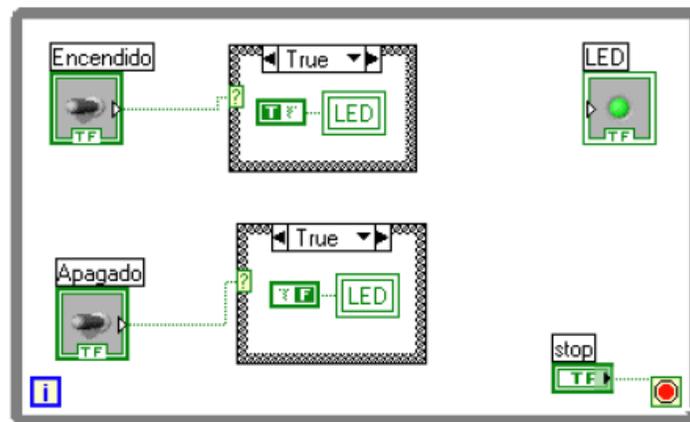


Imagen 2.59. Diagrama de bloques

Fuente: Extraída de LabVIEW

De manera que cuando se presiona el pulsador Encendido se ejecutará el caso verdadero en el cual se le asigna el valor de verdadero al LED, por lo tanto enciende, cuando se deja de presionar el botón de encendido el caso que se ejecuta el falso pero como no ha programado nada en ese caso, el valor en la variable local queda guardado. Para apagarlo debe ejecutarse el caso verdadero del Case al que se ha conectado el pulsador de apagado.

B) Usando variables Globales: El panel frontal queda igual al ya visto anteriormente. Ahora, en lugar de crear variables locales, vamos a crear una

variable  global, se encuentran en, *Functions* » *All Functions* » *Structures* » *Global Variable*, ahora hacemos doble clic sobre el icono de la variable global aparecerá una ventana de panel frontal aquí coloca cualquier control o indicador booleano como se muestra en la imagen 2.60, guardamos y cerramos este panel frontal.

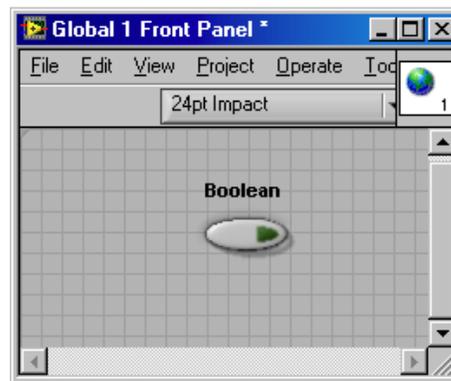


Imagen 2.60. Indicador Booleano

Fuente: Extraída de LabVIEW

Ahora en el diagrama de bloques, hacemos clic sobre el icono de la variable global, aparecerá la opción *Boolean*, la seleccionamos y automáticamente cambiará, ahora podemos guardar datos booleanos en dicha variable global, hacemos 3 copias de esta variable, tal como se muestra en la Imagen.

El funcionamiento de este programa es el mismo que en el caso anterior, lo único que ahora debemos leer de la variable global para encender el LED.

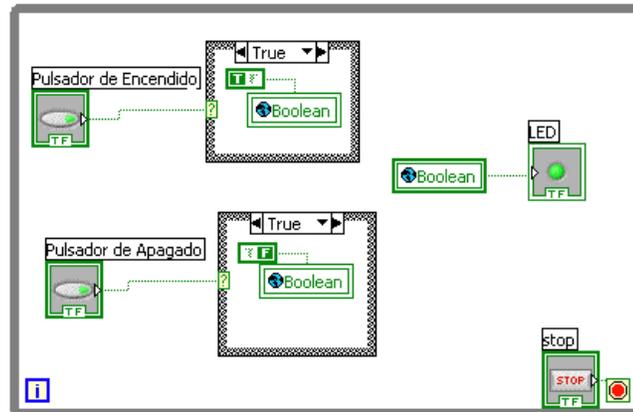


Imagen 2.61. Ejemplo con Booleanos

Fuente: Extraída de LabVIEW

2.2.3.15. Ejemplo: Construcción de un VI.

En este apartado se mostrará cómo construir una aplicación mediante el empleo del entorno de programación que proporciona LabVIEW®.

- Panel Frontal

En primer lugar, se debe construir el panel frontal deseado, que en este ejemplo debe tener el siguiente aspecto:

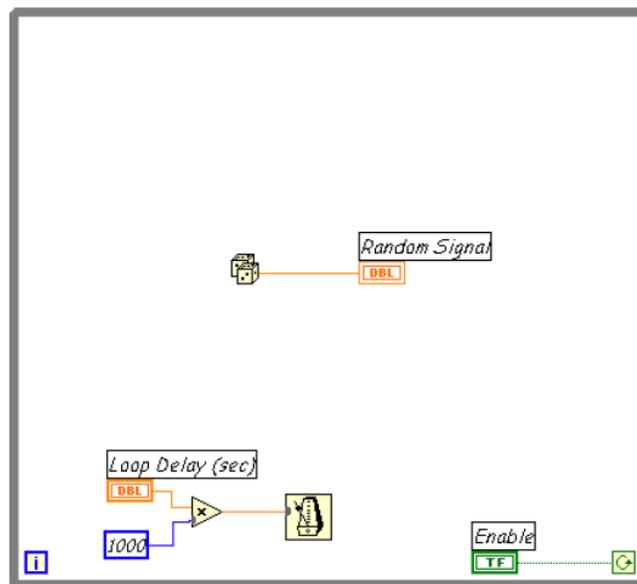


Proceso a seguir:

1. Abrir un *panel frontal* nuevo.
2. Colocar un "*vertical switch*" (paleta *Boolean*), cuyo nombre será *Enable*. Su finalidad será finalizar la adquisición.
3. Emplear la *Labeling Tool* para crear una etiqueta libre para *ON* y *OFF*.
Utilizar la *Coloring Tool* para hacer que el borde de dicha etiqueta sea transparente. La *T* en el borde inferior izquierdo de la paleta de colores hace transparente un objeto.
4. Colocar el gráfico (*waveform chart*), situado en la paleta *Graph*. Su nombre será *Random Signal*. El gráfico representará valores aleatorios en tiempo real.
5. El gráfico tiene un display digital que muestra el último dato. Pulsar el botón derecho del ratón situado sobre el gráfico, y seleccionar *Digital Display* del submenú *Show*. Asimismo se deberá deseleccionar *Legend* y *Palette* del mismo submenú .
6. Empleando la *Labeling Tool*, pulsar dos veces con el botón izquierdo del ratón sobre el 10.0 en el eje *Y* del gráfico, introducir 1.0 y pulsar fuera del gráfico. Así se habrá cambiado el fondo de escala.

7. Colocar un *knob* (paleta *Numeric*), cuyo nombre será *Loop Delay (sec)*. Este control determinará la velocidad de ejecución del bucle. Pulsar sobre él con el botón derecho del ratón y deseleccionar *Digital Display* del submenú *Show*.
 8. Empleando la *Labeling Tool*, pulsar dos veces con el botón izquierdo del ratón sobre el 10.0 de la escala, introducir 2.0 y pulsar fuera del control para introducir el nuevo valor.
- Diagrama de Bloques.

El siguiente es el aspecto que presentará el diagrama de bloques una vez finalizada su construcción:



1. Abrir el diagrama de bloques (menú *Window*, opción *Show Diagram*).
2. Colocar el *While Loop* (subpaleta *Structures* de la paleta de funciones). Dicha estructura, como todas las demás es de tamaño ajustable.
3. Seleccionar la función *Random Number (0-1)* de la subpaleta *Numeric* del menú de funciones.
4. Seleccionar la función *Wait until Next ms Multiple* de la subpaleta *Time & Dialog* del menú de funciones.

 5. Seleccionar la función de multiplicación de la subpaleta *Numeric*, del menú de funciones, así como una constante numérica, introduciendo el valor 1000 en lugar de 0, que es el que aparece por defecto.



 6. Colocar los cables tal y como se muestra en la figura anterior, empleando para ello la *Wiring Tool*.

 7. Volver al *panel frontal*. Con la *Operating Tool* poner el interruptor en su posición *ON*. Ejecutar el programa pulsando el botón *run*. 

La frecuencia de ejecución de las iteraciones del bucle *While* es la indicada en el *panel frontal* con el control *Loop Delay (sec)*. Es decir, se generará y representará un valor aleatorio cada periodo de tiempo (en segundos) seleccionado.

8. Para finalizar la ejecución del bucle, colocar el interruptor en la posición de *OFF*. De ese modo la condición de ejecución del bucle *While* será falsa, por lo que se detendrá a la siguiente iteración.

2.2.3.16. Programa de Ejemplo: Mezclador de Pintura.

“La imagen 2.63 muestra un mezclador de pintura. En la parte superior del mezclador hay dos tuberías por donde se suministran dos componentes diferentes. En la parte inferior se aprecia una sola tubería que transporta la mezcla de pintura preparada. Por una tubería en la parte inferior del tanque se conduce la mezcla ya lista. El programa de ejemplo controla la operación de llenado, supervisa el nivel del tanque y controla un ciclo de mezcla y de calefacción como describe a continuación:

Paso 1: Llenar el mezclador con el componente 1.

Paso 2: Llenar el mezclador con el componente 2.

Paso 3: Supervisar el nivel del mezclador para c

Paso 4: Mantener el estado de la bomba si se abre el interruptor de arranque.

Paso 5: Arrancar el ciclo de mezcla y calefacción.

Paso 6: Poner en marcha el motor del mezclador y la válvula de vapor.

Paso 7: Vaciar el mezclador.

Paso 8: Contar cada ciclo”.

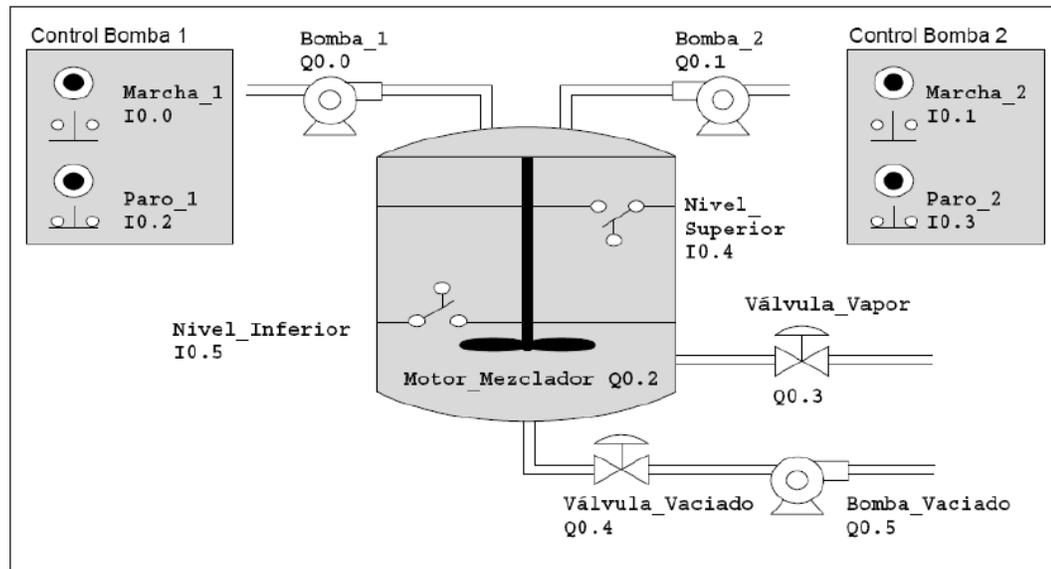


Imagen 2.63. Mezclador de Pintura



CAPITULO III

MARCO METODOLOGICO

3.1. Tipo de Investigación.

Según la definición dada por Fidias G. Arias en su Guía para la Elaboración de un Proyecto de Investigación, el presente trabajo corresponde a un tipo de investigación tecnicista, el cual cumple con etapas de planificación, producción y función, y cuenta con una posible aplicación inmediata.

Debido a que la investigación intenta satisfacer una necesidad de la Institución, mediante el desarrollo de una herramienta didáctica viable para el Laboratorio virtual de Control, este trabajo también corresponde a un proyecto de tipo factible.

3.2. Diseño de la Investigación.

El desarrollo del presente trabajo conlleva una serie de etapas que van desde la revisión documental para conocer el manejo de ciertas herramientas (hardware y software), hasta el diseño y aplicación de dichos instrumentos de programación y la elaboración de lo mismo, proceso que llevará al investigador a la realización de un trabajo de campo. Por lo tanto el diseño de la investigación puede ajustarse a las etapas de una *investigación experimental*, la cual según su definición, consiste en la intervención directa del investigador sobre el objeto de estudio durante todo el desarrollo de la misma.

Por otro lado, como el proyecto que se plantea busca caracterizar físicamente un concepto teórico, como lo es laboratorio virtual de control, se debe diseñar un procedimiento para fines descriptivos caracterizando el objeto de estudio a través de una descripción exacta de su comportamiento y de los resultados.



3.3. Procedimiento Metodológico.

A fin de cumplir con cada uno de los objetivos trazados, y en base al diseño de investigación elegido, se propone seguir un orden para el desarrollo del trabajo según las siguientes fases:

Fase N 1: Estudio, adiestramiento y manejo del software gráfico.

Se conoce todo lo relativo al software gráfico a usarse en el desarrollo del laboratorio virtual de control, para lo cual se acude a manuales de usuario, hojas de especificaciones técnicas, documentos en línea y otros trabajos anteriores relacionados.

Fase N 2: Selección de la tarjeta de adquisición de datos y creación tarjeta de acople.

Para la aplicación deseada en el banco de control de proceso y control de temperatura se define el tipo de tarjeta de adquisición de datos a usarse que se ajuste de manera eficiente al control. Para ellos se revisaron diferentes tipos de fuentes electrónicas y experimentos previos que permiten un punto de comparación con nuestro sistema a simular. Además de la creación de una tarjeta de acople de voltajes para la comunicación entre los periféricos del banco y la DAQ.

Fase N 3: Diseño del Banco de Control de Procesos.

En esta fase se fabricara el banco de Control de Procesos según sus medidas, materiales a utilizar y componente del mismo.

Fase N 4: Programación en el software gráfico adaptado al banco de control.

Se simulará y programará el banco de control a implementar dentro del software a fin de dar factibilidad al trabajo antes de implementarlo directamente sobre el banco. Tomando en cuenta los elementos que va a soportar y controlar.

Fase N 5: Verificación del sistema y software previamente programado

Al hacer el montaje de la programación se verificara su correcta funcionalidad, tanto de todas sus partes como los resultados que arroja el software.

Fase N 6: Puesta en marcha del banco de control.

Para la puesta en marcha se vacía la lógica programada en la tarjeta de adquisición de datos la cual se encuentra conectada a la PC y a su vez conectada a periféricos y dispositivos a manipular dentro del banco, con esto comprobado, se pone en marcha el proceso programado verificando que todo opere según lo esperado. Ver imagen 3.1

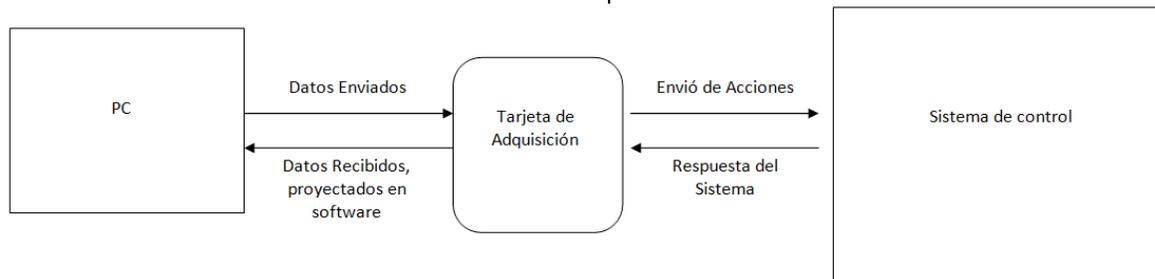


Imagen 3.1. Diagrama de comunicación Tarjeta - PC

Fase N 7: Toma de datos y análisis

Al tener el sistema en su funcionamiento óptimo se tomaran los datos arrojados para hacer la comprobación, además de analizar el tipo de sistema estudiado y su comportamiento.

Fase N 8: Elaboración de compendio de prácticas.

Se elaborará un compendio de prácticas haciendo uso de los bancos de control de proceso y de temperatura, con el fin de dar al estudiante ciertas condiciones de adiestramiento y estudie en la realidad los conocimientos obtenidos en el área de teoría de control.

CAPITULO IV

DESARROLLO DEL BANCO DE CONTROL DE PROCESOS Y RESULTADOS

4.1. Introducción.

El objetivo principal del presente proyecto consiste en el desarrollo de un Laboratorio Virtual de Sistemas de Control haciendo uso de Software Gráfico que permita al estudiante aplicar sus conceptos aprendidos en forma práctica. Esto implica que dicho proyecto debe estar compuesto por una Tarjeta de Adquisición de Datos (DAQ), capaz de comunicarse a través de un computador, recibir y ejecutar órdenes programados por el mismo a través de un Software Gráfico y poner en marcha el sistema, es decir, ejecutar el proceso en el Modelo de Aplicación. Este proceso se representa por el esquema presentado a continuación:

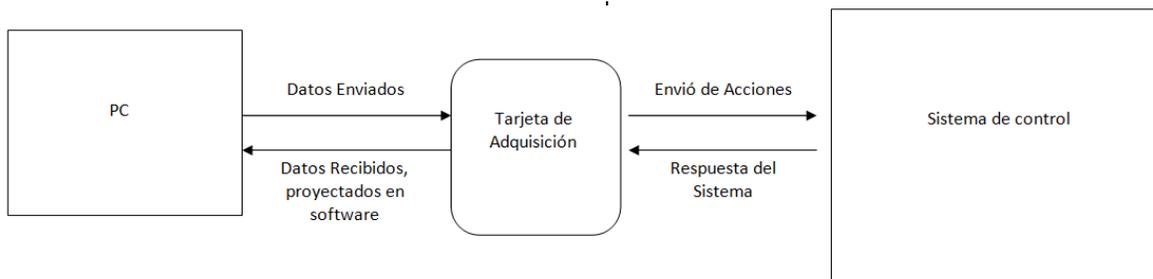


Imagen 4.1. Esquema Básico del banco de Control de Procesos.

En el presente capítulo se describen las fases que llevaron el cumplimiento de los objetivos para la elaboración del banco de Control de Procesos, y se señalan los resultados que se obtuvieron luego de la puesta en marcha.

4.2. Fases de Desarrollo.

4.2.1. Fase N° 1: Estudio, adiestramiento y manejo del software gráfico.

Como se ha mencionado previamente, el sistema del banco de Control de Procesos a implantar se desarrolla a través de un Software Gráfico que permite la comunicación entre

un computador y la Tarjeta de Adquisición de Datos (DAQ). Según la página web y fuentes bibliográficas consultadas de la National Instruments, este Software Gráfico a programar que permite la comunicación con la tarjeta es LabVIEW® ya que posee una rápida programación y comunicación con la tarjeta.

Para nuestra programación usaremos el Software LabVIEW® versión 9.0 de la National Instruments. En la Imagen 4.2 se observan las imágenes del Software mencionado.



Imagen 4.2. Vista principal del Software.

Fuente: Software LabVIEW® de la pantalla inicial

Se hizo el estudio de este Software donde se conocieron aspectos acerca del manejo de la interfaz de usuario (VI: Instrumento Virtual), sus funciones, su programación, la comunicación con la tarjeta de Adquisición de Datos, entre otros. La información recopilada sobre el manejo de este Software se obtuvo de diversos documentos en línea y de la ayuda del mismo programa la cual se muestra explicada en el apartado 2.2.3 del capítulo II. En la Imagen 4.3 se aprecia la interfaz de programación al iniciarse.

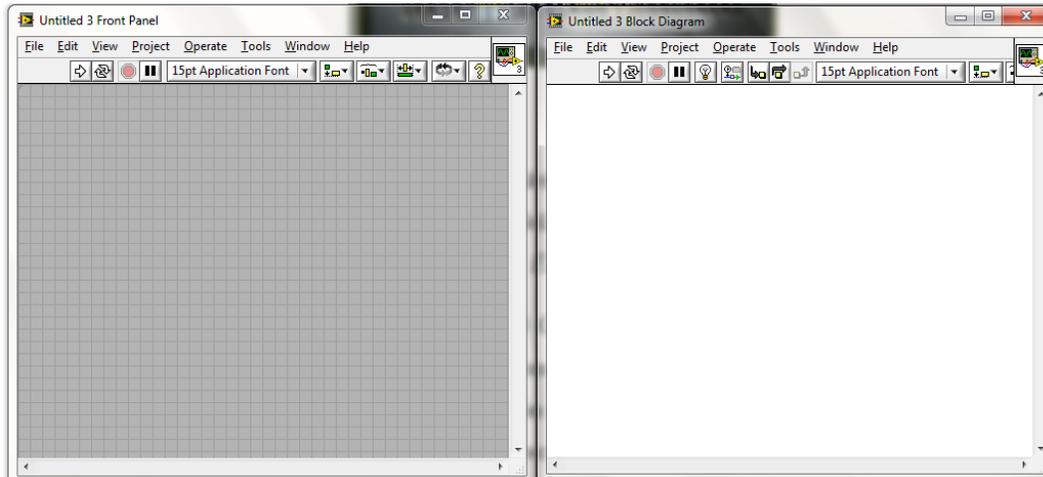


Imagen 4.3. Panel frontal para la programación del Software.

Como se menciona en su estudio, el software posee dos editores distintos para la programación, donde cada cual tiene su ventaja y desventajas. Ambos editores se programan gráficamente.

4.2.2. Fase N° 2: Selección de la Tarjeta de Adquisición de Datos y creación de tarjeta de acople.

Para nuestro Banco de Control de Procesos, con la cantidad de entradas y salidas a utilizar, además de las variables analógicas que se pretenden manipular y su temporización, usaremos la Tarjeta de Adquisición de Datos NI USB-6008, DAQ Multifunción de Bajo Costo de 12 Bits, 10 kS/s, la cual integra entradas y salidas analógicas, entradas y salidas digitales, entre otros. En National Instruments existen varios modelos de Tarjetas de Adquisición de Datos lo cuales se diferencian por la cantidad de entradas y salidas que poseen, además de las diferentes funciones que ellas delegan entre otras.

Esta Tarjeta de Adquisición de Datos según su hoja de especificaciones tiene las siguientes características:

- Alimentación de 5V por vía USB.

- 8 canales de entradas analógicas de 10V, 12 bits, 10 kS/s.
- 2 canales de salidas analógicas de 5V, 12 bits a 150 S/s.
- 12 Entradas/Salidas digitales de 5V.
- 1 Contador/Temporizador de 5V de 32 bits.
- Compatible con LabVIEW, LabWindows/CVI y Measurement Studio para Visual studio .NET
- Software controlador NI-DAQmx y software interactivo NI LabVIEW SignalExpress LE para registro de datos.

En la Imagen 4.4.1 se observan imágenes del dispositivo mencionado.



Imagen 4.4.1. Imágenes de la Tarjeta de Adquisición de Datos. A) Vista Frontal. B) Vista Frontal donde van conectados los cables. C) Vista del conector USB.

Luego se creó la tarjeta de acople de voltaje (ver apartado 4.2.3.6 de este capítulo), esto en vista de la necesidad que existía en adquirir las señales de los periféricos del banco de proceso, las cuales trabajan en su mayoría con 12VDC, mientras que la DAQ solo emite

y recibe señales de 5VDC, voltajes mayores a estos podrían causar daños graves a la tarjeta.
(Ver imagen 4.4.2)



Imagen 4.4.2. Tarjeta de Acople

4.2.3. Fase N° 3: Diseño del Banco de Control de Procesos.

Antes de comenzar con el diseño del Banco de Control de Procesos se hizo necesario definir claramente el modelo de aplicación práctica que se va a implementar, para luego en base a éste, realizar un diseño adecuado de la estructura. Se tomaron en cuenta ciertos requerimientos al elegir esta aplicación, entre los cuales se pueden mencionar:

- ✓ Portabilidad: una aplicación cuyas dimensiones permita hacer un diseño portátil del banco.
- ✓ Costo: un sistema cuya elaboración resulte accesible económicamente.
- ✓ Simplicidad: una aplicación con una gama de complejidad ajustando a los requerimientos de la cátedra de Teoría de Control I, Teoría de Control II y Teoría de Control III, a cuyos cursantes irá destinado el proyecto.
- ✓ Operabilidad: un sistema cerrado funcionalmente que permita ejecutar diferentes tipos de prácticas que se puedan ejecutar en un mismo lugar.
- ✓ Originalidad: un proceso distinto a los que tienen elaborados los otros laboratorios de Instrumentación y Automatización.



Se estudiaron diversas opciones, entre las cuales se pueden nombrar.

- Un sistema de un brazo robótico.
- Un sistema de control de velocidades de motores.
- Un sistema de un horno industrial.
- Un sistema de control de nivel de tanque sensando temperaturas

La última opción fue el modelo de aplicación más atractivo adaptado debido a la dinámica de fluidos que involucra y a la posibilidad de incorporar la variable temperatura, cuya dinámica resulta interesante analizar por la familiaridad que se tiene con los sistemas de tanques a través de diversas asignaturas del área de Control. Posteriormente, de un sistema de control de nivel de tanques y temperatura se derivó la idea de un modelo más específico, denominado Procesos de Calentamientos.

El modelo de aplicación Procesos de Calentamientos básicamente consiste en un sistema de dos tanques: **A** y **B**, donde el tanque **B** por medio de una resistencia térmica se calienta el líquido una vez se encuentre lleno el tanque al nivel deseado. Sin embargo, para representar el fluido del sistema se propuso utilizar agua.

Para este sistema se diseñaron tanques de sección rectangular con las siguientes dimensiones, tal como se muestra en la Imagen 4.5.

- Tanque A: 20 cm (Ancho) x 20 cm (Largo) x 25 cm (Alto).
- Tanque B: 20 cm (Ancho) x 20 cm (Largo) x 30 cm (Alto).

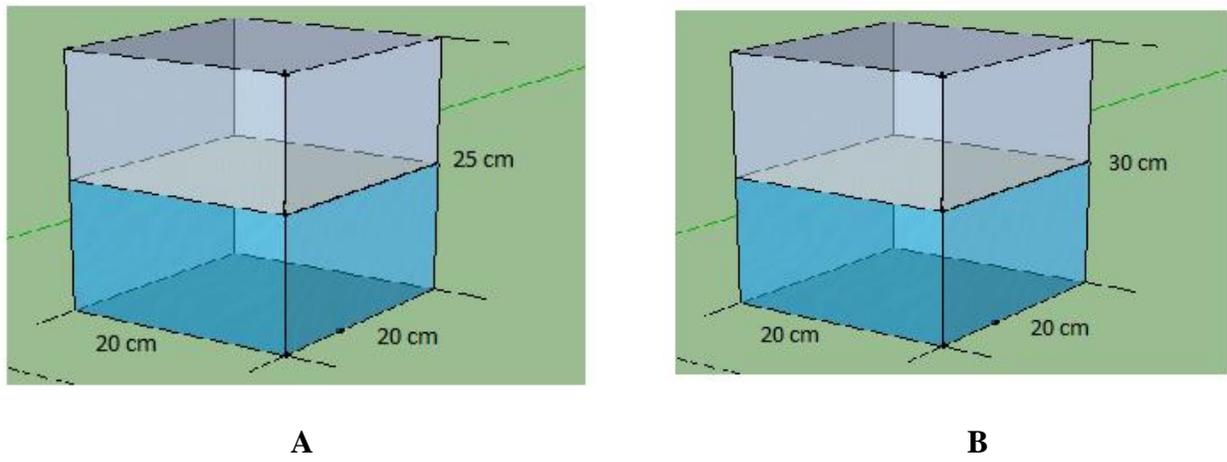


Imagen 4.5. Diseño propuestos para los tanques del sistema.

La colocación de los tanques en el sistema se estableció de la siguiente manera: el tanque A reposa sobre una superficie y el tanque B sobre otra superficie por encima de A, comportándose como un tanque aéreo. En la siguiente Imagen 4.6 se ilustra la colocación y las distancias de separación.

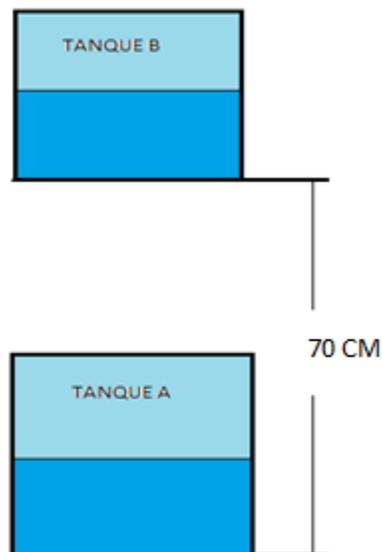


Imagen 4.6. Colocación de los tanques en el sistema.

Para el intercambio de fluidos entre los tanques se definió un sistema de tubería de la siguiente forma:

- La salida del tanque A ubicada en la parte baja se conecta directamente a la entrada del tanque B, ubicada en la parte alta del mismo, donde el transporte de fluido se hará por medio de bomba de agua. B va ubicada en la salida. (Ver Imagen 4.7).
- La salida del tanque B ubicada en el fondo, se conecta directamente a la entrada del tanque A en la parte más alta del mismo, donde el paso del fluido se controlará con una electroválvula EV ubicada en la entrada. (Ver Imagen 4.7).

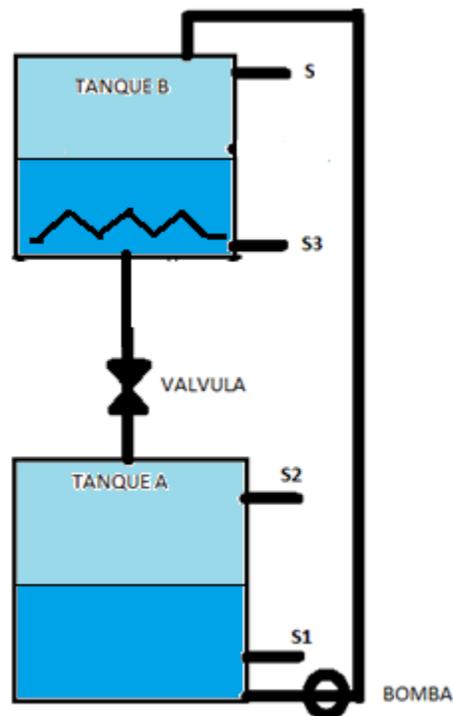


Imagen 4.7. Esquema de sistema de tuberías y sensores para el sistema de tanques en la aplicación “Procesos de Temperaturas)

Para realizar el control de llenado, temperatura y vaciado se propuso la colocación de sensores de proximidad capacitivos en cada tanque, debido a la eficiencia que tiene este tipo de sensores en la detección de líquidos como el agua y además que el material que se usó fue el acrílico no presentaba problemas de detección. En el caso de los sensores



inductivos, estos no se usaron ya que se requiere sensor líquidos y no metales. En el caso de los sensores ópticos, estos no se usaron ya que son muy sensibles y la mayoría de ellos no duran mucho tiempo. Se requerirán 4 sensores de proximidad capacitivos lo cuales estarán distribuidos de la siguiente forma. (Ver Imagen 4.7).

Para el caso de la temperatura tendrá una resistencia térmica ubicada en el tanque B para que realice su calentamiento según el nivel de líquido. (Ver Imagen 4.7).

- El tanque A tendrá sensores para la detección de dos niveles de altura del fluido: nivel alto (S2) y nivel bajo (S1).
- El tanque B por poseer resistencia térmica, tendrá sensores para la detección de dos niveles de altura de fluidos al igual que el tanque A: nivel alto (S4) y nivel bajo (S3).

4.2.3.1. Diseño de la Estructura de soporte.

Para realizar el diseño de la estructura de soporte se tomaron en cuenta los siguientes aspectos básicos:

- Se requiere contar con espacio suficiente para la colocación de los sistemas de tanques de la aplicación. Los soportes deben permitir ubicar los tanques A y B, con las respectivas distancias de separación del diseño.
- Se requiere contar con un soporte con suficiente área para la colocación de tableros de control de la Tarjeta de Adquisición de Datos, modelo de aplicación y un computador que manipulara el proceso.
- Debe tener buena resistencia mecánica para poder soportar el peso de todos los componentes.
- Debe ser una estructura portátil.

En base a lo anterior, se diseñó el modelo de banco como el que se muestra en la Imagen 4.8a donde señala una posible manera de distribuir los componentes

principales. La imagen 4.8b muestra la estructura de soporte necesaria para elaborar este modelo de banco. Como se observa en el diseño, se cuenta con área rectangular de 123cm (ancho) x 70 cm (alto) x 48 cm (largo) destinado para la colocación del tanque A con un computador y otros elementos a usar, una repisa de 123 cm (ancho) x 30 cm (alto) x 25 cm (largo) para la colocación del tanque B con otro sistema.

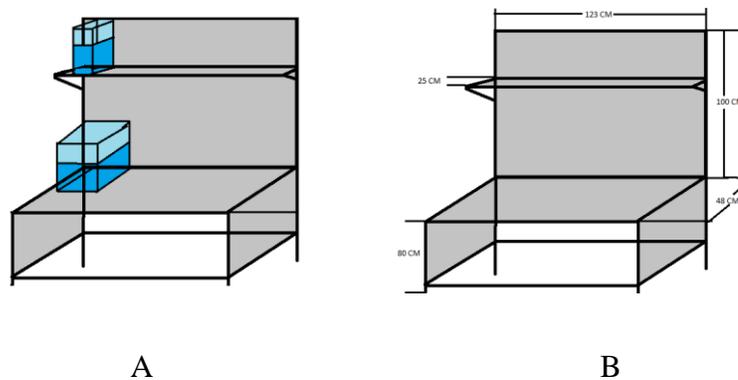


Imagen 4.8. Esquema de medidas del banco de proceso (A) y ubicación de los tanques (B)

Por otro lado, para que la estructura soporte cuente con una resistencia mecánica adecuada para portar todos los elementos del Banco de Control de Procesos, y a su vez generar el menor de costos posibles, se asentó fabricarla utilizando tubos cuadrados de hierro de 1x1 pulgadas. El procedimiento de la fabricación se describe de la siguiente manera:

- Elaboración de la estructura de soporte. La estructura de soporte diseñada se implementó con piezas de hierro unidas con soldaduras. La estructura resultante (Ver Imagen 4.9) posee buena resistencia para soportar el peso de los componentes, y el costo de elaboración fue relativamente bajo.



Imagen 4.9. Estructura resultante.

Para la elaboración de la estructura de soporte se utilizaron los siguientes materiales de la Tabla 4.1.

Tabla 4.1. Lista de materiales para la elaboración de la estructura de soporte.

MATERIAL	CANTIDAD
Tubo cuadrado de hierro 1x1 pulgada.	12 Mts
Electrodos para soldar.	½ kilo.
Maquina de soldar.	1
Taladro.	1
Segueta	1

La elaboración se llevo a cabo de la siguiente forma:

- Se hizo cortes de los tubos cuadrados de hierro a las medidas del diseño usando la segueta.
- Se soldaron las piezas según el diseño hasta obtener la estructura deseada.
- Por último, se le aplico anticorrosivo y se pintó con esmalte brillante en color negro como en la Imagen 4.10.



Imagen 4.10. Estructura finalizada.

4.2.3.2. Diseño de los tanques.

Para elaborar los tanques se hicieron 2 recipientes rectangulares en acrílico transparente de 3 mm de forma manual. Los recipientes tienen las siguientes dimensiones:

- Tanque A: 20 cm (Ancho) x 20 cm (Largo) x 25 cm (Alto).
- Tanque B: 20 cm (Ancho) x 20 cm (Largo) x 30 cm (Alto).

Los materiales utilizados para la fabricación de los tanques se especifican en la tabla 4.2.

Tabla 4.2. Lista de materiales para la elaboración de los tanques.

MATERIAL	CANTIDAD
Lamina de acrílico transparente de 3 mm de 122 cm x 183 cm	1 pieza
Cortador estándar para plástico.	1 pieza
Sellador de silicón Silipex	1 tubo
Tirro blanco	1 rollo



Pegamento universal (Ester de Cianocrilato)	2 tubos
Angulo de aluminio de ½	3 Mts
Taladro	1
Bisagras de plástico	2 piezas
Papel contact negro	½ Mt

- Se hizo la unión de las piezas para la base y las paredes de cada recipiente mediante sellador de silicón. Luego se le hicieron pruebas para verificar de que no existieran fugas de líquido.
- Se hicieron orificios en las paredes de los recipientes para las distintas conexiones de tuberías.
- Para poder atornillar los recipientes a la superficie se le colocó un marco de aluminio en forma de U a la altura de la base, con perforaciones para los tornillos.
- Se hicieron tapas para los tanques con piezas de acrílico transparente y bisagras, y se fijaron a los recipientes mediante pegamento universal. Estas tienen perforaciones para el paso de tuberías y algunos cables.
- A cada tanque se le coloco letras adhesivas en la cara externa.
- Para el acabado final se le colocaron marcos de aluminio sobre cada una de las esquinas superiores y traseras.

Al culminar con la elaboración de los tanques, se colocaron las superficies de apoyo en la estructura de soporte, utilizando láminas de MDF de 15 mm. Posteriormente se procedió a colocar los tanques en su posición exacta sobre estas superficies. Ver Imagen 4.11.



Imagen 4.11. Tanque elaborado y mesa para soporte.

4.2.3.3. Elaboración e instalación del sistema de tuberías para los tanques.

Este sistema está elaborado con tuberías flexibles de cobre de 3/8 pulgadas de diámetro, cuya elección se hizo a la buena resistencia del material ante la corrosión y la durabilidad en las distintas uniones. Se utilizó una válvula solenoide “normalmente cerrada” de 3/8 pulgadas con conexiones tipo flare para tuberías de cobre ya que se necesita que el líquido se mantenga en el tanque b y sin necesidad de estar enviando una señal para que se mantenga cerrada, de usar una válvula solenoide “normalmente abierta” el líquido caería en todo momento y habría que estar enviando una señal para que esta se cierre. Se utilizo también una bomba de pecera para el transporte del liquido al tanque B (ver Imagen 4.12). Las características técnicas de estos elementos se muestran en las tablas 4.3 y 4.4.



Imagen 4.12. Válvula Solenoide y Bomba.

**Tabla 4.3.** Característica de la Válvula Solenoide.

CARACTERISTICA	DESCRIPCION
Marca	Danfoss
Serial Parte	Solenoide: N° 018F7396 Válvula: N° 032U3606
Voltaje del Solenoide	12V DC
Potencia del Solenoide	18 W
Tipo de Válvula	4.50B G 38F, normalmente cerrada, conexión 3/8” ISO 228-1
Temperatura de trabajo	-30 °C a +120 °C
Presión de trabajo	30 bar

Fuente: Especificación del Producto

Tabla 4.4. Característica de la bomba.

CARACTERISTICA	DESCRIPCION
Marca	Resun
Modelo	SP – 980
Voltaje	110 V
Altura mínima de elevación	1,50 M
Potencia	9 W
Volumen máximo de entrega	550 L/H

Fuente: Especificación del Producto



Imagen 4.13. Conexión de la bomba y válvula solenoide.

Los materiales utilizados para elaborar todo el sistema de tuberías se especifican en la tabla 4.5.

Tabla 4.5. Lista de materiales para las conexiones de tuberías.

MATERIAL	CANTIDAD
Tubería de cobre de 3/8 pulgadas	3 mts
Codo conexión flare	1 pieza
Copas de bronce de 3/8 pulgadas	6 unidades
Adaptador flare 3/8" a tubería de agua de 1/2"	2 unidades
Válvula solenoide 12V, conexión flare de 3/8"	1 pieza
Cortador de tubos	1 pieza
Rebordeador para tuberías de cobre	1 pieza



Bomba sumergible para pecera 12V	1 pieza
Niple PVC de 1/2" para tubería de agua, rosca continua	2 unidades
Empacadura de contratuerca para grifería de 1/2"	4 unidades
Teflón	1 rollo
Válvula manual	1 pieza
Adaptadores para tuberías de 3/8" flare	3 unidades
Niple de 24 cm para desagüe de 1/2"	1 pieza
Manguera flexible blanca	1/2 metro

La elaboración del sistema de tuberías se hizo de la siguiente manera:

- Se hicieron cortes de las piezas de tubería de acuerdo a las medidas necesarias.
- Se dobló en curva la pieza que va en la entrada del tanque B.
- Se rebordearon las tuberías de cobre que poseen conexiones tipo flare con la válvula solenoide.
- Se perforaron los orificios en la estructura de soporte para el paso de tuberías.
- A cada tanque se le colocó una salida hermética de desagüe, con niples PVC, Empacadura de goma, tuercas y teflón. Ver Imagen 4.14.a y 4.14.b
- Se atornillaron los tanques a la estructura se le perforaron los orificios de entrada de tuberías y salidas.
- Se armó el sistema de tuberías para cada tanque. Ver Imagen 4.14.c
- Se apretaron fuertemente las válvulas (Solenoide y manual) a las tuberías y se fijó este sistema en los tanques. El acoplamiento de la tubería con la bomba externa de gasolina se hizo a presión utilizando manguera flexible de goma blanca.
- Para el vaciado del tanque A se le colocó una válvula manual de paso que servirá de desagüe. El vaciado se realizó a un recipiente externo. Ver Imagen 4.14.d



(a)



(b)



(c)



(d)

Imagen 4.14. Imágenes de la instalación del sistema de tuberías del proceso.

4.2.3.4. Elaboración de sistema de sensores de nivel.

Se utilizaron 4 sensores de proximidad capacitivos ya que vamos a sensar líquido como el agua, de 18mm de diámetro ya que fueron los que mejor precisión nos daba al momento de detectar el nivel del líquido y con distancia de detección estándar de 8mm (Ver Imagen 4.15). Todos los sensores de tipo NPN y normalmente abiertos. Dos (2) de estos sensores son de una marca y los otros dos (2) de otra.



Imagen 4.15. Sensores capacitivos usados.

NOTA: La razón de esta diferencia entre la marca de los sensores se debió a que se dispuso de diferentes proveedores y únicamente por disponibilidad en el mercado. Es por esto que se procedió a buscar otra marca pero de funciones similares. Además de que el tanque b va a calentar líquido y el sensor debe ser más robusto como se muestra en la Imagen 4.15.

Las especificaciones técnicas de los sensores se muestran en la tabla 4.6.

Tabla 4.6. Características técnicas de los sensores capacitivos.

CARACTERISTICAS	DESCRIPCION
Marca N° 1	Autonics
N° de modelo	Sensor NPN: CR18 – 8DN
Distancia de detección	8mm \pm 10%
Alimentación	12 – 24V DC
Corriente de consumo	Max 15Ma
Carga máxima que soporta	200Ma
Temperatura de trabajo	-25 a 70 °C
Potencia interna	Contra polaridad inversa y picos de voltaje
Peso	72 g

Marca N° 2	Winston
N° de modelo	Sensor NPN: CM18 – 3005NA
Distancia de detección	0 – 5mm
Alimentación	6 – 36V DC
Corriente de consumo	Max 10mA
Carga máxima que soporta	200mA
Temperatura de trabajo	-25 a 80 °C
Potencia interna	Contra polaridad inversa y picos de voltaje
Peso	105 g

Fuente: Hoja de especificaciones técnicas del producto

Los sensores se colocan sobre un sistema de soporte vertical hecho con tubo de aluminio ranurado, los cuales se fijaron de la madera al lado de cada tanque, y por cuyas ranuras se realiza el desplazamiento vertical de dichos detectores. Cada sensor se sujeta mediante una pieza de acrílico transparente que posee un orificio, además un tornillo para el ajuste de la pieza a la altura deseada. (Ver Imagen 4.16).



Imagen 4.16. Imágenes para la elaboración del sistema de sensores.

4.2.3.5. Elaboración de interfaz del tablero frontal.

En la interfaz de tablero se colocaron 2 pulsadores. El pulsador ubicado a la izquierda de la DAQ significa “Start”, es decir, el botón de inicio del Banco de Proceso. El botón ubicado a la derecha denotado como “Stop” significa parada, es decir, a la hora de que el sistema presente alguna falla, tendrá su parada de emergencia para evitar accidente alguno. Luego se puede apreciar en la Imagen 17, tres (3) círculos de color rojo, que vienen a representar si algunos de los dispositivos como electroválvula, Bomba o Calentador estén ejecutando una acción. Estos estarán en modo “On” encendidos cuando se encienda algunos de los dispositivos. El recuadro blanco que se observa en la imagen es la Tarjeta de Adquisición de Datos.

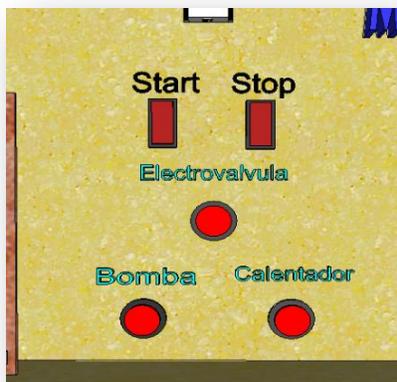


Imagen 4.17. Interfaz del tablero frontal

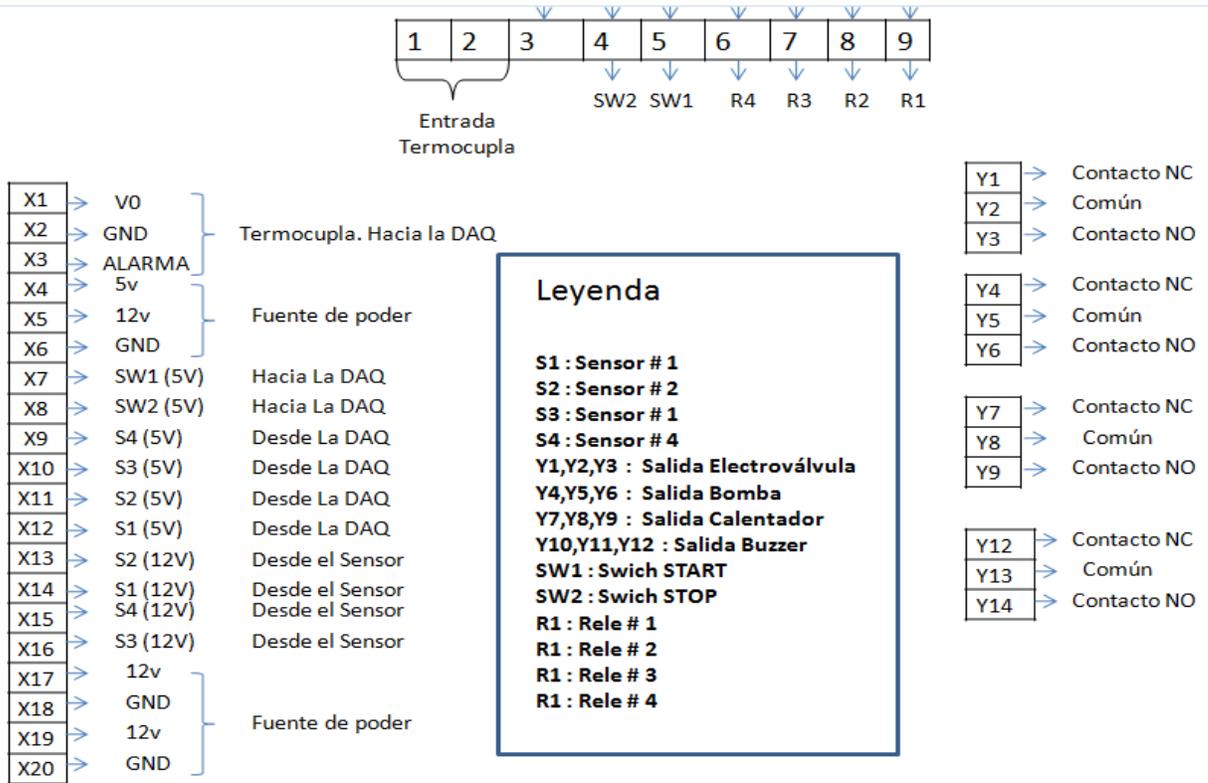
Tabla 4.7. Lista de materiales para la elaboración de interfaz del tablero.

MATERIAL	CANTIDAD
Pulsadores (12 VDC)	2 unidades
Luz Pilotos	3 unidades
Mecha para madera de 3/8"	1 pieza
Taladro	1
Mecha para puntear	1 pieza
Tornillo	1 pieza



4.2.3.6. Cableado del Banco de Control de Procesos.

Para el cableado del banco de procesos se uso cable número 18 AWG de color azul para los diferentes valores de tensiones ya que esto nos ayudó a obtener una mejor estética en el cableado y que este tipo de cable no se parte tan fácilmente al momento de ajustarlo a la tarjeta. Se uso también cable negro - rojo para las tierras (ver tabla 4.8). En la Imagen 4.18 se especifica cómo se distribuyeron las conexiones de todos los dispositivos hacia la tarjeta de acople al sistema. Esta se diseño con la finalidad de permitir la conexión entre los dispositivos del sistema y la DAQ, ya que los valores máximos que la misma puede tener en salidas es de 5V, al igual en las entradas. Se diseño con el fin de tomar esos valores de tensiones y hacerlos conectar con los sensores y otros dispositivos, ya que ellos se energizan con valores mayores al de la tarjeta. Esta tarjeta (Ver imagen 4.19 y 4.20) se diseñó con optoacopladores (ver imagen 4.21), para aislar eléctricamente como manera de protección los pines I/O (entradas y salidas) de la Tarjeta de Adquisición de Datos de los sensores que se conecten, el elemento empleado en este diseño es el 4N25, que es un optoacoplador estándar de un único canal, conformado por un diodo infrarrojo de arseniuro de galio y un fototransistor NPN, posee una velocidad de conmutación de 3 us y un aislamiento de 2500 V. Otros componentes de la tarjeta de acople es el relay con bobinas de 5VDC a 12 A – 120V (Modelos OE SIT-38-0512-L2) que consiste en un interruptor que opera magnéticamente al recibir una señal de voltaje por la bobina y permite el paso de señal a la salida, un integrado que tiene un arreglo de transistores Darlington (ver imagen 4.22), que se compone de siete transistores NPN con salida de cátodo común y diodo de fijación para la conmutación de cargas inductivas, proporcionando hasta 500 mA y 50 Voltios por salida, se seleccionó este dispositivo debido a que cumple con los requerimientos de corriente y tensión de los relés y adicionalmente requiere menor espacio que utilizar un circuito discreto para manejar cada salida, un amplificador operacional para la conversión de grados centígrados a voltios y leds como indicadores de los elementos activos en el proceso.



(a)



(b)



(c)

Imagen 4.18. Cableado del Banco de Control



Imagen 4.19. Tarjeta diseñada para conectar con la DAQ

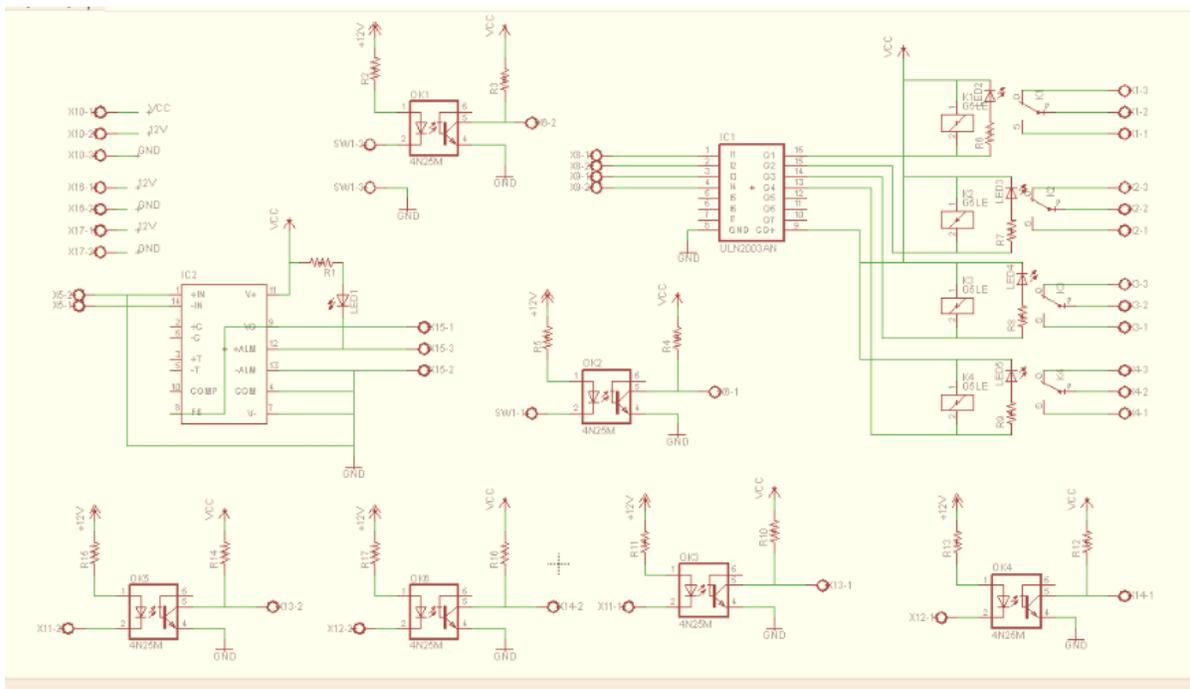


Imagen 4.20. Circuito de la Tarjeta diseñada

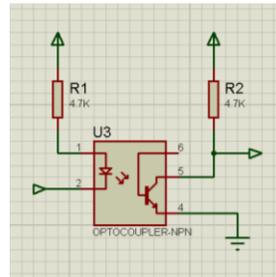


Imagen N° 4.21. Optoacoplador para la tarjeta diseñada

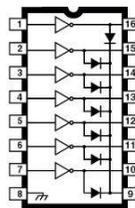


Imagen 4.22. Puenete Darlington

Tabla 4.8. Distribución de colores de cableado del tablero.

COLOR DEL CABLE	USO
Azul	Entrada digitales Salidas digitales Entrada analógica
Negro – Rojo	GND (tierra)
Rojo	Para conectores de la fuente a la tarjeta diseñada (5V) Para algunas regletas (5V)
Negro	Para las tierras de la fuente a la tarjeta diseñada. GND



	Para algunas regletas. GND
Amarillo	Para algunas regletas y conectores de la fuente. (12V)
Azul	Para conexiones con Leds y pulsadores (12V)
Negro – Rojo	Para las tierras de los Leds y Pulsadores

Tabla 4.9. Nomenclatura de identificación del cableado del tablero.

Función	Identificación	Significado
Entradas a la DAQ	TERM +	AI0
	TERM -	AI4
	S1	P0.0
	S2	P0.1
	S3	P0.2
	S4	P0.3
	SW1	P1.0
	SW2	P1.1
Salida a la DAQ	EV	P0.4
	B	P0.5
	CA	P0.6
	A	P0.7
Sensores de Entrada	S1	Sensor nivel bajo tanque inferior



	S2	Sensor nivel alto tanque inferior
	S3	Sensor nivel bajo tanque superior
	S4	Sensor nivel alto tanque superior
	SW1	START
	SW2	STOP
Dispositivos de Salida	EV	Electroválvula
	B	Bomba
	CA	Calentador
	A	Buzzer alarma

Tabla 4.10. Características del calentador.

CARACTERISTICA	DESCRIPCION
Alimentación	110V AC
Potencia	1200 W
Marca	Desconocida
Tipo de calentador	De bulbo
Termostato	0 – 40 – 60 – 80 °C de 110V AC
Corriente de consumo	15 A

Fuente: Hoja de especificaciones técnicas del producto



Imagen 4.23. Imágenes del calentador.

Tabla 4.11. Estudio de carga para la fuente de alimentación 12VDC.

Elemento de Carga	Corriente en estado ON	Cantidad de elementos	Corriente total en ON
Sensor Autonics	15Ma	2	30mA
Sensor Winston	10Ma	2	20mA
Bomba	3 A	1	3 A
Electroválvula	1,5 A	1	1,5 A
		TOTAL	5 A

El **Banco de Control de Procesos** ya finalizado, se puede apreciar en una galería de imágenes mostradas en el apéndice C del presente trabajo, especialmente una vista frontal y una vista posterior.

4.2.4. Fase N° 4: Programación en el software gráfico adaptado al banco de control.

Para la programación del Banco de Control de Procesos se hizo la implementación de un programa prueba, a fin de verificar el buen funcionamiento del sistema, conformado por la Tarjeta de Adquisición de Datos (DAQ), tarjeta de acople, los paneles de control y los elementos de Procesos de Temperatura. El programa de prueba consistió en lo siguiente:

“Haciendo uso del software LabVIEW® con su hardware que consiste en la Tarjeta de Adquisición de Datos y del sistema de Temperatura que se dispone en el banco, descrito previamente en el apartado 4.2.3 de este capítulo, implementamos el siguiente proceso de temperatura (el sistema se muestra nuevamente en la Imagen 4.24):

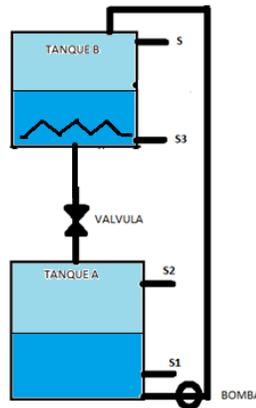


Imagen 4.24. Representación del sistema de Temperatura.

- ✓ Con el Tanque A suficientemente lleno y el tanque que calienta el líquido completamente vacío, al accionar un pulsador se enciende la bomba que comienza a llenar el Tanque B con el líquido del Tanque A. Cuando el Tanque A queda vacío su bomba debe apagarse.
- ✓ Cuando el contenido del tanque de temperatura llegue a su nivel alto, se debe encender el calentador.
- ✓ Una vez encendido el calentador se espera cierto tiempo hasta que el líquido llegue hasta su temperatura deseada a través de una recta simulada por motivos de problemas con la DAQ ya que esta tarjeta no tiene puerto para leer termocupla (ver conclusiones), para luego apagarse y abrir automáticamente la electroválvula para que el líquido calentado caiga por gravedad hasta llenar el Tanque A.
- ✓ Una vez lleno el Tanque A se cierra la electroválvula y en este instante concluye el proceso, la cual no debe repetirse cíclicamente”.

Para el desarrollo de la lógica de este programa se siguió la secuencia mostrada en diagrama de flujo de la Imagen 4.25.

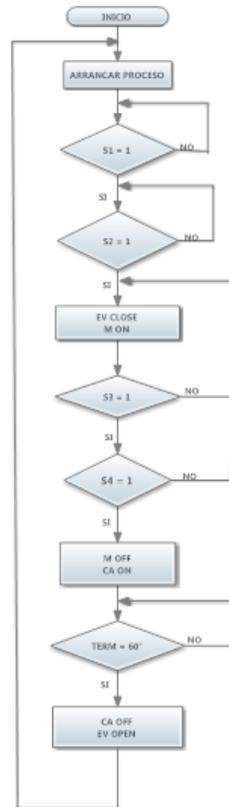


Imagen 4.25. Diagrama de flujo de proceso de temperatura.

Para esta práctica es necesaria la implementación de rutinas que permitan el buen funcionamiento de un sistema de control, tales arreglos son los siguientes:

Flip - Flop: Este arreglo permite emular una bobina Set y Reset que nos posibilita el control de una variable con respecto al estado de 2 o más entradas. Está compuesto por un “Loop While” y un “Case Structure”, el mismo tiene como entrada 3 casos las cuales son enviadas por una estructura llamada “Build Array Function” esta permite recibir las condiciones de las entradas involucradas, y actuar con respecto a la configuración de los casos antes mencionado.

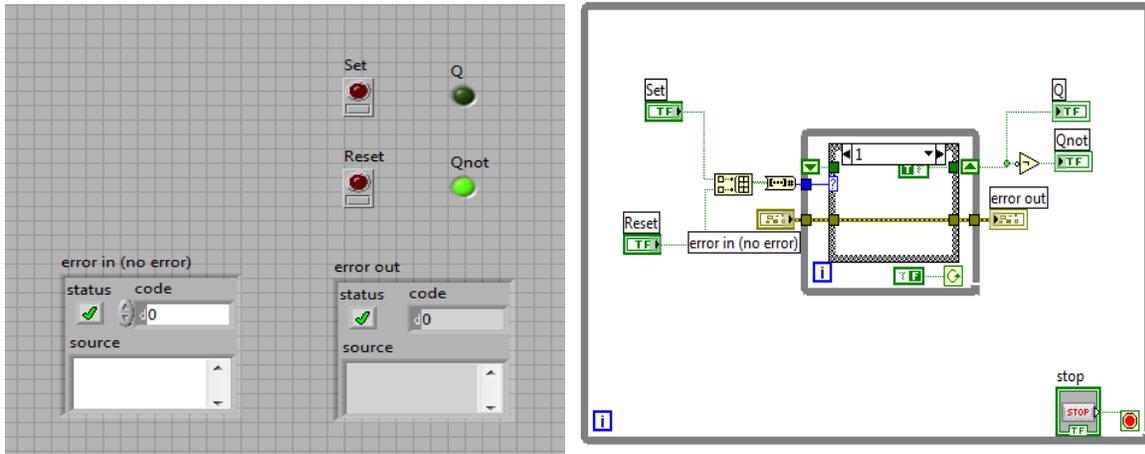


Imagen 4.26 Estructura FLIP-FLOP

Temporizadores: En este arreglo se implemento un “Loop While”, así como también un “Elapsed Time Express VI” este permite el ingreso del lapso del tiempo, también posee una salida booleana que indica el cumplimiento del lapso de tiempo, así como también el tiempo actual en que se encuentra el conteo, además este arreglo pose el valor agregado de que puede ser usado como un TON o un TOFF, con la simple modificación de colocar una compuerta negada a la salida del indicador booleano, con esto logramos una amplia mundo de aplicaciones en los cuales puede ser implementado sin problema.

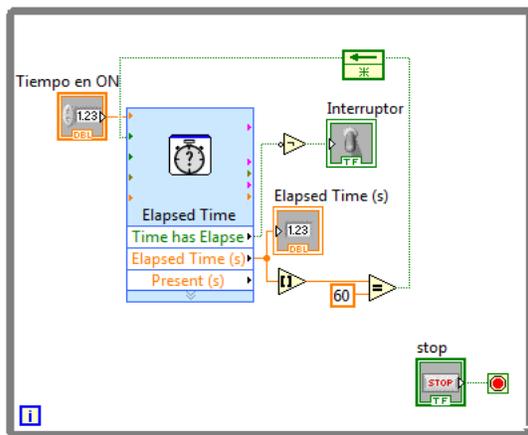


Imagen 4.27 Estructura Timer TON

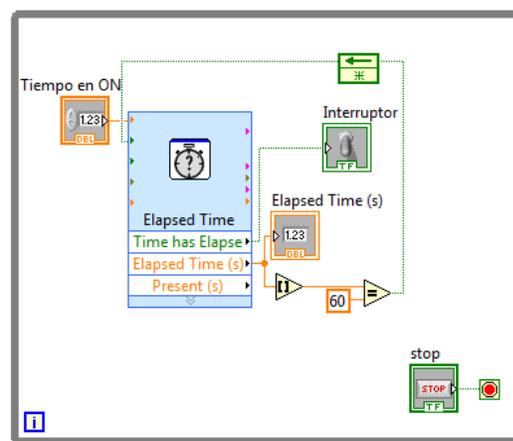


Imagen 4.28 Estructura Timer TOFF

A continuación se muestra el programa desarrollado mediante el programa LabVIEW® de la National Instruments donde se muestra cada paso gráficamente. (Ver Imagen 4.26).

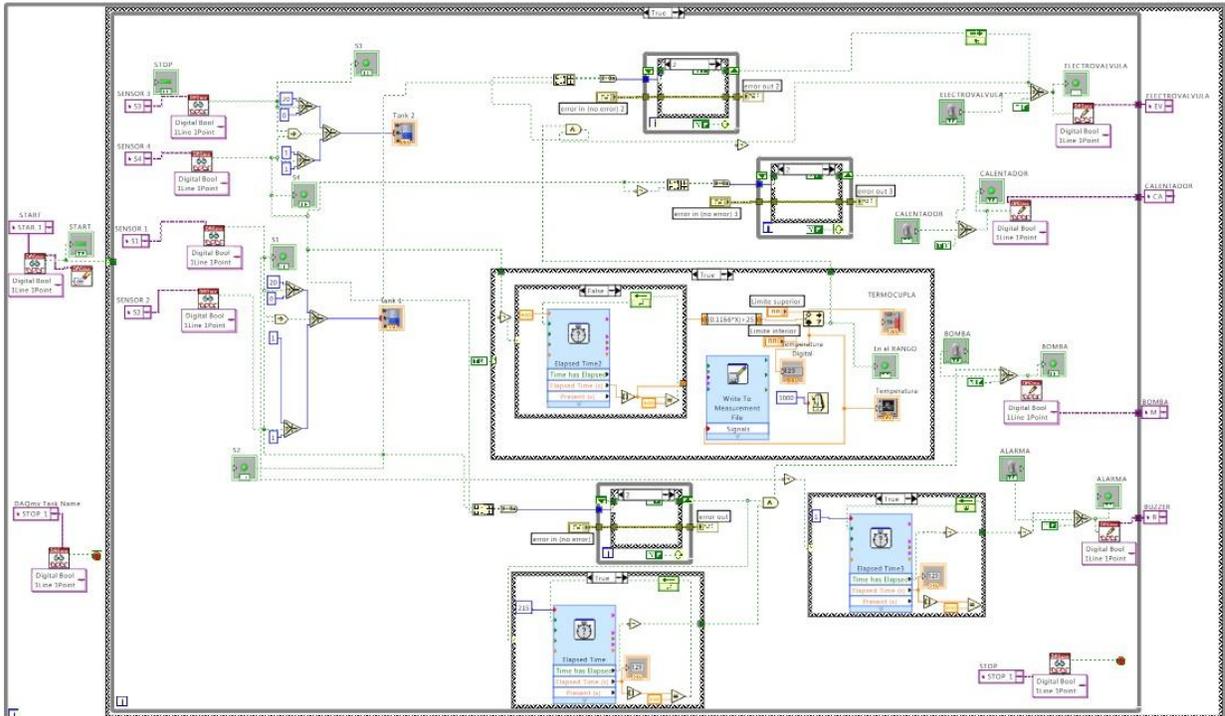


Imagen 4.29. Programación del VI en LabVIEW®

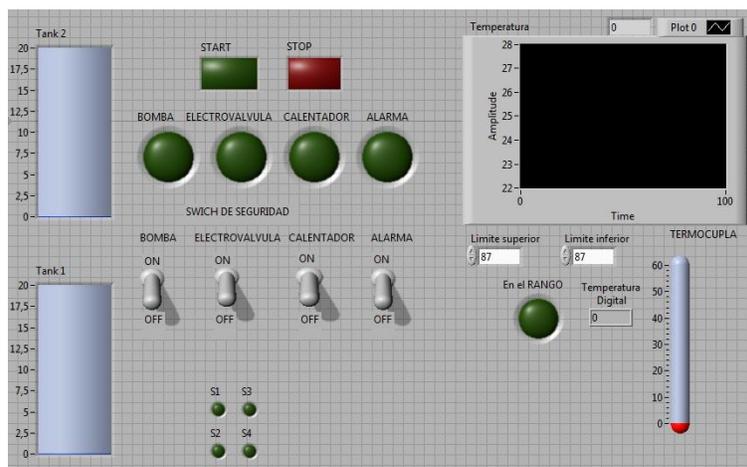


Imagen 4.30. Programación gráfica del proceso de temperatura. (Panel de Trabajo)



4.2.5. Fase N° 5: Verificación del sistema y software previamente programado.

Siguiendo la asignación de variables dadas en las tablas, se realizó el conexionado de los siguientes bornes de la Tarjeta de Adquisición de Datos con los del panel de control de la aplicación, se colocó alimentación de los sensores con una fuente de 12VDC externa, al igual que la electroválvula. Se llenó el Tanque A manualmente como lo exige las condiciones del proceso. Luego se hizo la comunicación de la Tarjeta de Adquisición con la computadora y al proceso para ser iniciado.

4.2.6. Fase N° 6: Puesta en marcha del banco de control.

Luego de ser verificado todo el Banco de Control de Procesos de Temperatura, se procedió a poner en marcha el mismo. Haciendo diferentes pruebas a diferentes temperaturas como set point. El banco de control en marcha respondió a todas las alarmas de emergencias programadas y logró mantener un control ON - OFF, otra vez hasta verificar todo.

4.2.7. Fase N° 7: Toma de Datos y Análisis.

Se procedió a tomar Datos de los diferentes tiempos durante el calentamiento del agua. Estos datos fueron enviados por el mismo programa a un archivo de formato Excel. Luego de tomar esos datos se procedió a hacer un análisis de los cambios que pueden ocurrir en caso de que exista una perturbación al proceso y si se mantiene o no estable durante la descarga del tanque B.

4.2.8. Fase N° 8: Elaboración de compendio de prácticas.

Práctica de Laboratorio N° 1: Proceso de vaciado y llenado de tanque.

En esta práctica se le mostrará al estudiante el proceso de cómo hacer el control para trasladar un fluido, en este caso agua, de un lugar a otro. Como objetivos se especifican:



- Analizar las características de hardware NI USB – 6008 así como el software LABVIEW 9.0, con la finalidad de conocer su funcionamiento y aplicabilidad.
- Familiarizarse con los elementos del banco de control a través de la implementación de un proceso de vaciado y llenado de tanques.

Práctica de Laboratorio N° 2: Proceso de vaciado y llenado de tanques mediante variación de temperatura Manual.

En esta práctica se le mostrará al estudiante el proceso de cómo hacer el control para manipular temperatura manualmente. Como objetivos se especifican:

- Analizar las propiedades de hardware NI USB – 6008 para adquirir señales analógicas.
- Uso de componentes en panel frontal como indicadores, perillas, termómetros y graficas en tiempo real.
- Uso de estructuras de registro de datos en el diagrama de bloques.

Práctica de Laboratorio N° 3: Proceso de vaciado y llenado de tanques mediante variación de temperatura por medio de ecuaciones.

En esta práctica se le mostrará al estudiante como hacer el control de temperatura del Banco de Procesos, a través de ecuaciones. Como objetivos se especifican:

- Analizar las propiedades de hardware NI USB – 6008 para adquirir señales analógicas.
- Uso de componentes en panel frontal como indicadores, termómetros y graficas en tiempo real.
- Uso de estructuras de registro de datos en el diagrama de bloques, así como también estructuras de expresiones matemáticas.



CAPÍTULO V

DESARROLLO DE PRÁCTICAS DE LABORATORIO

5.1. Introducción

El presente capítulo presenta un grupo de prácticas de laboratorio para el banco de control de procesos, las cuales podrán ser desarrolladas dentro del laboratorio de Control

Al inicio de cada practica, se señala los objetivos de la misma, y se propone el cumplimiento de un **Pre-laboratorio** con la finalidad de manejar conceptos teóricos necesarios para el buen desarrollo de la mima.

El Desarrollo de la práctica va acompañado de un dibujo donde se presenta las condiciones necesarias, dejando al alumno la tarea de programación en LABVIEW, ejecución del proceso, y verificación del cumplimiento correcto de la práctica.

Tomar en cuenta siempre:

- Es importante señalar que un tanque se encuentra vacío cuando su respectivo sensor de nivel bajo deja de detectar líquido independientemente de la altura donde esté ubicado dicho sensor. Igualmente, un tanque está lleno cuando su respectivo sensor de nivel alto detecta presencia de líquido.
- El llenado inicial de los tanques se debe realizar de forma manual utilizando los recipientes que se facilitan con el banco, y disponiendo de suficiente agua.
- Para el vaciado manual del tanque Inferior se debe colocar un recipiente bajo la tubería de vaciado manual que se encuentra en la parte inferior del banco.
- Se puede comprobar previamente el funcionamiento de cada “Dispositivo de Salida” conectando la alimentación del banco, y alimentando cada dispositivo directamente el borne de la fuente de 12 y 110V en la tarjeta encargada de la adaptación de voltajes.

5.2. Práctica de Laboratorio N° 1: Proceso de vaciado y llenado de tanque



Objetivos

- Analizar las características de hardware NI USB – 6008 así como el software LABVIEW 9.0, con la finalidad de conocer su funcionamiento y aplicabilidad.
- Familiarizarse con los elementos del banco de control a través de la incorporación de un proceso de vaciado y llenado de tanques.

Pre-Laboratorio

Investigar lo siguiente

- a) Investigar características eléctricas de NI USB – 6008
- b) Características del Software LABVIEW
- c) Funcionamiento de las siguientes operaciones dentro del software LABVIEW
 - i. Estructuras y ciclos (While , For , casos, eventos)
 - ii. Creación de tareas en NI-DAQmx por medio del software Measurement & Automation (salidas y entradas Digitales)
 - iii. Tipos de sincronización y temporizadores
 - iv. Estructuras Booleanas y de comparación.
- d) Funcionamiento del sensor de proximidad capacitivo y electroválvulas.

Desarrollo

Haciendo uso de hardware NI USB – 6008, del sistema que se dispone en el banco (descrito en el apartado 4.2.3 del capítulo IV), y de los interruptores que sean necesarios, implemente el siguiente proceso (ver Imagen 5.1)

- Con el tanque “A” lleno hasta el sensor S2 y el tanque “B” vacío, al accionar el botón de START arrancar la bomba.
Nota: si el tanque “A” está vacío, la bomba debe apagarse y sonará un buzzer durante 1 seg.
- Cuando el contenido del tanque superior alcanza el nivel máximo apagar la bomba.
- Luego, usar un proceso de temporizado de manera que el contenido permanezca en el tanque superior un tiempo de 10 seg.
- A continuación se dará apertura a la electroválvula.
- Se debe vaciar el tanque superior hasta llenar el tanque inferior a nivel máximo, con lo que concluirá el proceso.
Nota: en todo el proceso pulsando el botón de STOP se deberá parar el proceso.



Imagen 5.1. Banco de Control de Procesos (Nivel)



Imagen 5.2. Representación del sistema de vaciado y llenado de tanques

5.3. Práctica de Laboratorio N° 2: Proceso de vaciado y llenado de tanques mediante variación de temperatura Manual.

Objetivos

- Analizar las propiedades de hardware NI USB – 6008 para adquirir señales analógicas.
- Uso de componentes en panel frontal como indicadores, perillas, termómetros y graficas en tiempo real.
- Uso de estructuras de registro de datos en el diagrama de bloques.

Pre-Laboratorio

Investigar lo siguiente

- a) Características eléctricas de NI USB – 6008
- b) Funcionamiento de las siguientes operaciones dentro del software LABVIEW
 - i. Aplicación de graficas en tiempo real



- ii. Creación de tareas en NI-DAQmx por medio del software Measurement & Automation (salidas y entradas Analógicas)
- iii. Uso de Time delay
- c) Componentes del panel frontal como Indicadores, perillas y termómetros.
- d) Uso de “Write To Measurement File Express VI”, dentro de la ventana de diagramas de bloques

Desarrollo

Haciendo uso de hardware NI USB – 6008, del sistema de mezclado que se dispone en el banco (descrito en el apartado 4.2.3 del capítulo IV), y de los interruptores que sean necesarios, implemente el siguiente proceso (ver Imagen 5.3)

- Con el tanque “A” lleno hasta el sensor S1 y el tanque “B” vacío, al accionar el botón de START arrancar la bomba
Nota si el tanque “A” está vacío, la bomba debe apagarse y sonara un buzzer durante 1 seg.
Cuando el contenido del tanque “B” alcanza el Sensor S4 encender el calentador.
- Cuando el contenido del tanque “B” alcanza el Sensor S4 apagar la bomba
- Se debe mostrar en pantalla la temperatura actual del fluido en el tanque
- Mientras se calienta el agua subir la temperatura manualmente con una perilla en el panel frontal.
- Luego, al alcanzar una temperatura en un rango comprendido entre 55 °C y 60 °C se dará apertura a la electroválvula, y se apagará el calentador.
- Se debe vaciar el tanque “B” hasta llenar el tanque “A” a nivel máximo, con lo que concluirá el proceso.

Nota: en todo el proceso pulsando el botón de STOP se deberá parar el proceso.



Imagen 5.3. Banco de Control de Procesos (Temperatura)



Imagen 5.4. Representación del sistema de vaciado y llenado de tanques mediante variación de temperatura Manual



Post-Laboratorio

- a) Con los datos obtenidos en el registro de temperatura, y el archivo .lvm generado, realice lo siguiente:
 - i. Cambiar la extensión del archivo a .txt y crear un archivo en Excel con los datos obtenidos.
 - ii. Hacer uso del software MATLAB específicamente del comando “SISOTOOL” para el análisis e identificación del sistema estudiado.

- b) Implemente el uso de SubVI en “donde sea posible”, dentro de la programación planteada para la práctica.

5.4. Práctica de Laboratorio N° 3: Proceso de vaciado y llenado de tanques mediante variación de temperatura por medio de ecuaciones.

Objetivos

- Analizar las propiedades de hardware NI USB – 6008 para adquirir señales analógicas.
- Uso de componentes en panel frontal como indicadores, termómetros y graficas en tiempo real.
- Uso de estructuras de registro de datos en el diagrama de bloques, así como también estructuras de expresiones matemáticas.



Pre-Laboratorio

Investigar lo siguiente

- a) Características eléctricas de NI USB – 6008
- b) Funcionamiento de las siguientes operaciones dentro del software LABVIEW
 - i. Aplicación de graficas en tiempo real
 - ii. Creación de tareas en NI-DAQmx por medio del software Measurement & Automation (salidas y entradas Analógicas)
 - iii. Uso de Time delay
- c) Componentes del panel frontal como Indicadores, y termómetros.
- d) Uso de “Write To Measurement File Express VI”, dentro de la ventana de diagramas de bloques
- e) Uso de “Expression Node Function”, dentro de la ventana de diagramas de bloques

Desarrollo

Haciendo uso de hardware NI USB – 6008, del sistema de mezclado que se dispone en el banco (descrito en el apartado 4.2.3 del capítulo IV), y de los interruptores que sean necesarios, implemente el siguiente proceso (ver Imagen 5.5)

- Con el tanque “A” lleno hasta el sensor S1 y el tanque “B” vacío, al accionar el botón de START arrancar la bomba
- Nota si el tanque “A” está vacío, la bomba debe apagarse y sonara un buzzer durante 1 seg.
- Cuando el contenido del tanque “B” alcanza el Sensor S4 encender el calentador.

- Cuando el contenido del tanque “B” alcanza el Sensor S4 apagar la bomba
- Se debe mostrar en pantalla la temperatura actual del fluido en el tanque
- Mientras se calienta el agua la temperatura deberá aumentar con respecto a la siguiente ecuación $y = 0.1166 * x + 25$ (relación calculada con un tiempo de 5 min, que representa el calentamiento del agua hasta 60 grados con el calentador)
- Luego, al alcanzar una temperatura en un rango comprendido entre 55 y 60 °C se dará apertura a la electroválvula, y se apagará el calentador.
- Se debe vaciar el tanque “B” hasta llenar el tanque “A” a nivel máximo, con lo que concluirá el proceso.

Nota: en todo el proceso pulsando el botón de STOP se deberá parar el proceso.



Imagen 5.5. Banco de Control de Procesos (Temperatura)



Imagen 5.6. Representación del sistema de vaciado y llenado de tanques mediante variación de temperatura por medio de ecuaciones.

Post-Laboratorio

- a) Con los datos obtenidos en el registro de temperatura, y el archivo .lvm generado, realice lo siguiente:
 - iii. Cambiar la extensión del archivo a .txt y crear un archivo en Excel con los datos obtenidos.
 - iv. Hacer uso del software MATLAB específicamente del comando “SISOTOOL” para el análisis y la identificación del sistema estudiado.

- b) Implemente el uso de SubVI en donde sea posible, dentro de la programación planteada para la práctica.



Apéndice F

“Costo de Elaboración de Banco de Control de Procesos”



Material	Cantidad	Precio Unitario (Bs)	Total (Bs)
Tubo cuadrado de Hierro 1x1''	12	12,5	150
Lámina de madera	1	220	220
Tornillos pasantes de ½ x2''	30	2,5	75
Tuercas de ½''	30	1,5	45
Pies de amigo pequeños	2	7	14
Angulo de aluminio de ½ x ½''	6	35,71	214,26
Tornillos tira fondo de 1''	36	0,3	10,8
Tornillos tira fondo de ½''	36	0,25	9
Regletas 3A	8	5	40
Regleta de 15A	1	6	6
Remaches 5/32'' x 3/8''	36	0,22	7,92
Soporte de aluminio con ranura	4	2,68	10,72
Fabricación de tanque 20x20x25cm	1	440	440
Fabricación de tanque 20x20x30cm	1	470	470
Tubo de bronce 3/8''	3	42,41	127,23
Corta tubo B 1/8 x 1	1	41,8	41,8
Rebordador Tubo A	1	76,34	76,34
Abre madera 6pz espada	1	102,68	102,68
Teflón 1/2" profesional	1	1,92	1,92
Copa flare 3/8''	6	12,05	72,3
Codo flare 3/8'' x 90	1	26,74	26,74
Conector M 3/8x3/8''	2	15,13	30,26
Conector M 3/8x1/2''	1	17,81	17,81
Conector M 1/2x1/2''	2	17,81	35,62
Barniz Caoba Claro en Galón de 1/4	2	59	118
Mecha cilíndrica HSS 7/64	4	5,61	22,44
Teflón líquido	1	43,44	43,44
Hoja para segueta	2	8,55	17,1
Niple hexagonal	2	9,57	19,14
Conexiones de plástico	2	7,14	14,28
Marco para puerta de baño	1	70	70
Cable automotriz AWG 18 Azul	20	2,37	47,4
Pulsador redondo	2	6,61	13,22
Leds 12 VDC	3	16,54	49,62



Cable de audio	10	4	40
Válvula solenoide 3/8" NC 12 VDC	1	550	550
Sensor capacitivo CR18-8DN. Autonics	2	310	620
Bomba sumergible SP 980, 110V	1	214,29	214,29
Sensor capacitivo CM18-8DN. Winston	2	368	736
Belt Cre resistencia. Calentador	1	87,25	87,25
Buzzer 12 mm 6-15 V	1	17,86	17,86
Goma reductora	1	2,34	2,34
Pega tanque	1	22,77	22,77
Tubería de cobre 0,5	1	17,66	17,66
Controlador de temperatura. Termostato	1	99	99
Lamina de Acrílico 30x35 cm	1	80	80
Calcomanías (Varias)	1	80	80
Regleta 6 tomas	1	38,84	38,84
Pega multiuso	1	44,6	44,6
Breaker Hyundai C25	1	150	150
Marca cables	5	20	100
Cable N° 2x12 THW	5	9	45
Fabricación de tarjeta de acople	1	800	800
Relés de 12 VDC	2	40	80
Conector para relé	2	40	80
Canaleta de 3 mts	3	15	45
Fuente de poder de 550 W	1	210	210
		SUBTOTAL	6820,65
	IVA 12%	818,478	818,478
		TOTAL	7639,128



Apéndice D

“Especificaciones de la DAQ NI USB 9211”

Registrador de Temperatura de 4 Canales por USB

NI USB-9211A

- Dispositivo pequeño y portátil (12.1 x 8.6 x 2.5 cm)
- 4 canales de entradas de termopares
- Sensor interno para compensación de junta fría
- Soporte para termopares tipo J, K, R, S, T, N, E, y B
- Mediciones hasta de 1 muestra cada 166 ms
- Conectividad plug-and-play via USB
- Alimentación por bus



Sistemas Operativos

- Windows Vista/XP/2000

Software Incluido

- LabVIEW SignalExpress LE para adquisición, despliegue y registro de datos
- Software controlador NI-DAQmx

Introducción

El USB-9211A de National Instruments es un dispositivo registrador de temperatura con capacidad para conectar directamente hasta 4 termopares. Cuenta con conexión y alimentación de energía por USB, lo que lo hace ideal para tomar mediciones desde una PC en algún laboratorio o laptop en campo. Además, sus características como aislamiento de hasta 30 V canal-a-común, inmunidad al ruido y alto nivel de rechazo a voltaje de modo común permiten su uso en ambientes industriales.

El registrador de temperatura USB-9211A es ideal para diversas aplicaciones donde es importante tener un equipo portátil y pequeño como:

- Registro de temperatura portátil
- Medición de pocos canales
- Uso académico en laboratorio
- Monitoreo ambiental
- Aplicaciones embebidas para OEMs
- Adquisición de datos en vehículo



Software

El USB-9211A se incluye con el software configurable para realizar mediciones y guardar datos NI LabVIEW SignalExpress LE. Este software cuenta con características para configurar su aplicación de medición, graficar los valores de temperatura, almacenar los datos en archivo, o exportar los resultados a paquetes como Microsoft Excel. Si necesita capacidades adicionales de análisis o alarmas, puede adquirir la versión completa de LabVIEW SignalExpress. Además, el USB-9211A incluye el software controlador (*driver*) NI-DAQmx para su instalación, configuración y administración en Windows.

Cómo comprar

NI USB-9211A	Por correo: info.mexico@ni.com
Precio: \$580 USD*	Por teléfono: 01 800 010 0793
Número de parte: 779436-01	* Precio no incluye IVA, costos de importación y envío.

Tabla de Especificaciones

Número de Canales	4 canales de termopar, 1 canal interno de auto-cero, 1 canal interno de compensación de junta fría
Termopares soportados	J, K, R, S, T, N, E, B
Resolución	24 bits
Filtros	Paso bajo de 15 Hz
Protección de sobre voltaje	± 30 V entre cualquier entrada y común
Rango de voltaje de modo común	± 250 V entre común y tierra
Impedancia de entrada	20 M Ω
Corriente de entrada	50 nA
Consumo de corriente por USB	500 mA máx (2.5 mA en modo suspendido)
Dimensiones	12.1 x 8.6 x 2.5 cm
Conectividad a sensor	Terminal de tornillo
Interfaz de conexión a la PC	Hi-Speed USB 2.0

Número de mediciones por segundo

Número de canales medidos	Tipo de medición		
	Temperatura con CJC y auto-cero	Temperatura con auto-cero	Voltaje directo
1	4	6	12
2	3	4	6
3	2.4	3	4
4	2	2.4	3



Figura d.1. Modelo de DAQ USB 9211.



Apéndice B

Diagrama de las diferentes conexiones del Banco de Procesos

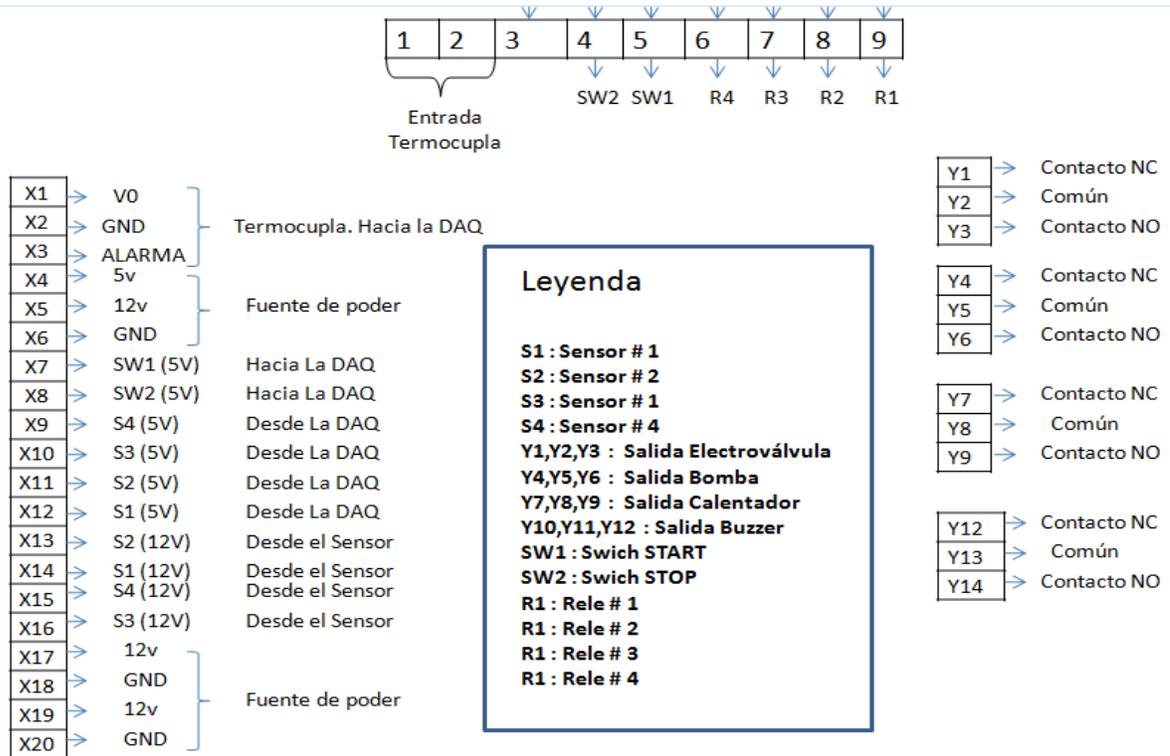


Figura b.1. Diagrama de conexión de Tarjeta de Acople

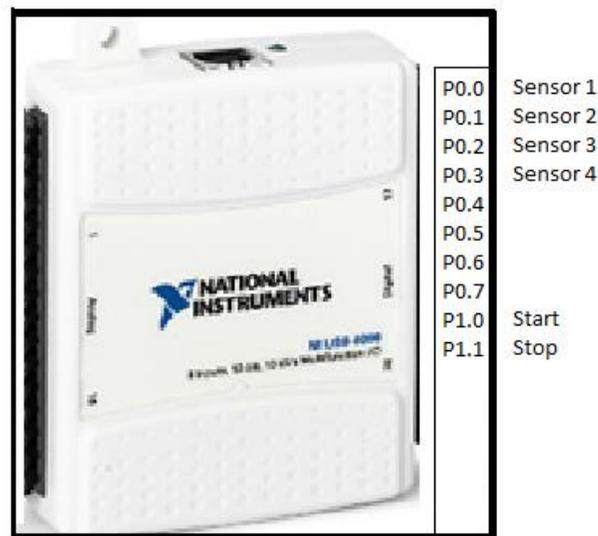


Figura b.2. Diagrama de Conexión de Entradas Digitales de DAQ

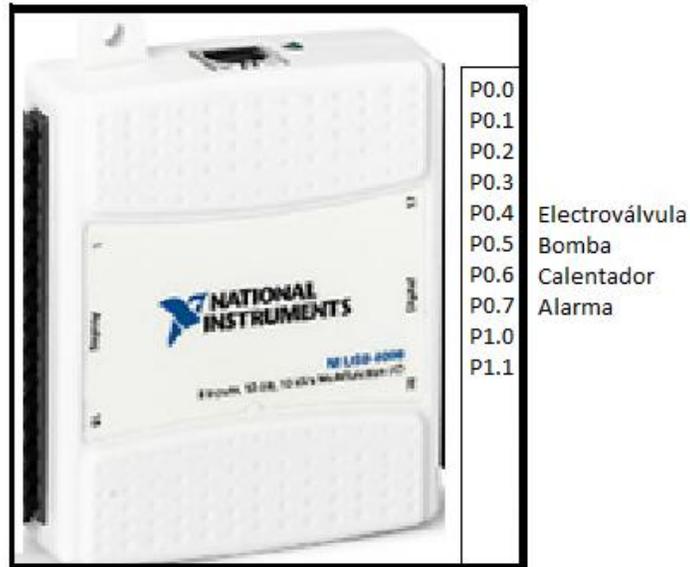


Figura b.3. Diagrama de Conexión de Salidas Digitales de DAQ

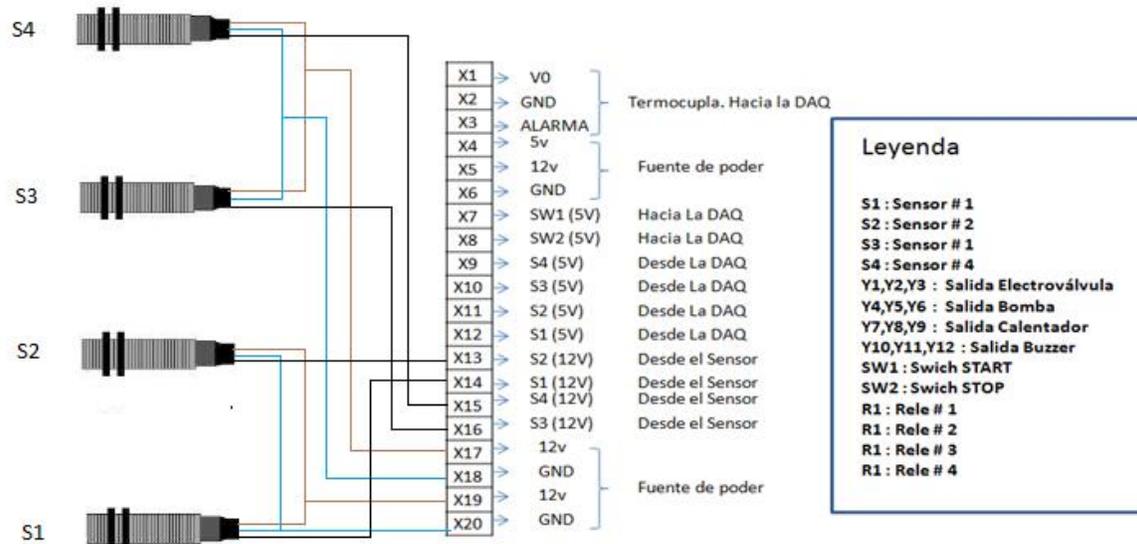


Figura b.4. Diagrama de Conexión de Sensores (Entradas)

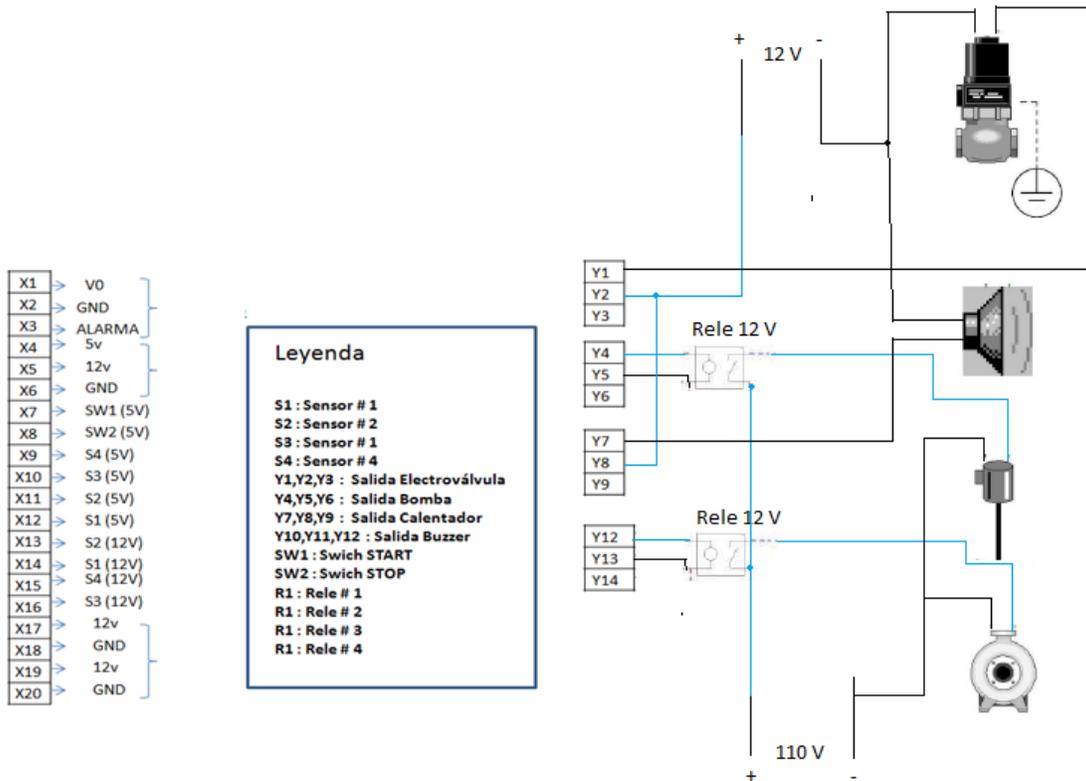


Figura b.5. Diagrama de Conexión de dispositivos de Salida.



Apéndice C

“Galería de Imágenes del Banco de Control de Procesos”

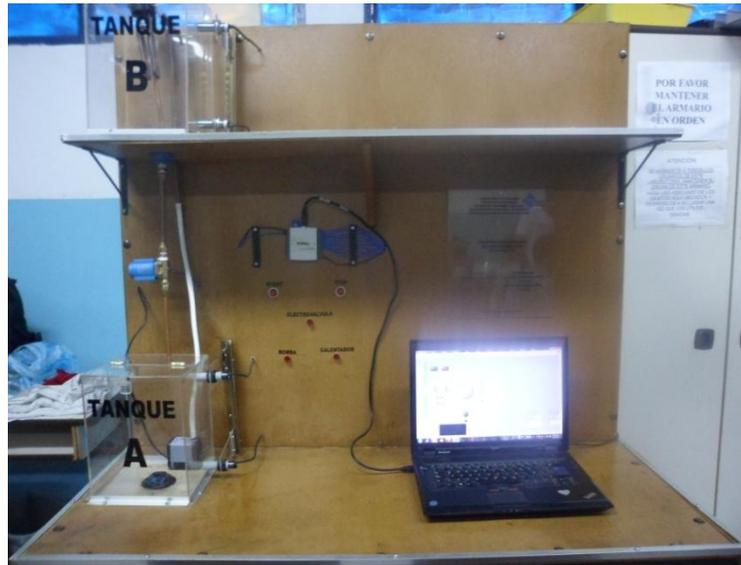


Figura C.1. Vista frontal a medio cuerpo del banco de proceso.



Figura C.2. Vista frontal a cuerpo completo del banco de procesos



Figura C.3. Vista del tanque A

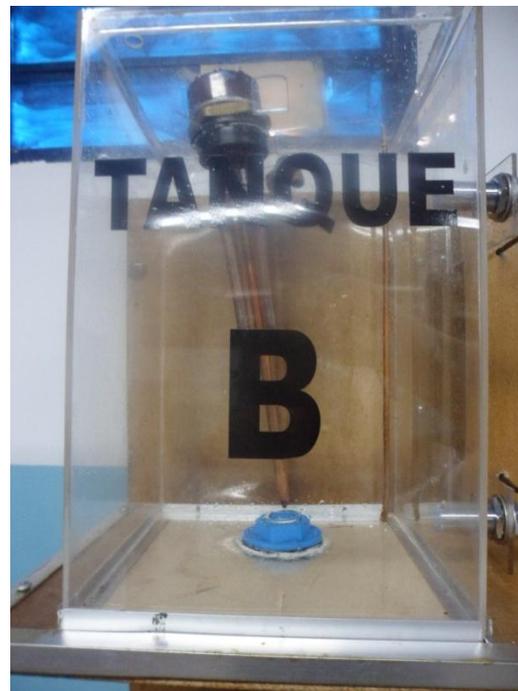


Figura C.4. Vista del tanque B



(A)



(B)

Figura C.5. Vista de los sensores.

A. (S1 debajo - S2 arriba) B. (S3 debajo, S4 arriba)



Figura C.6. Vista de los de instalación del banco de control

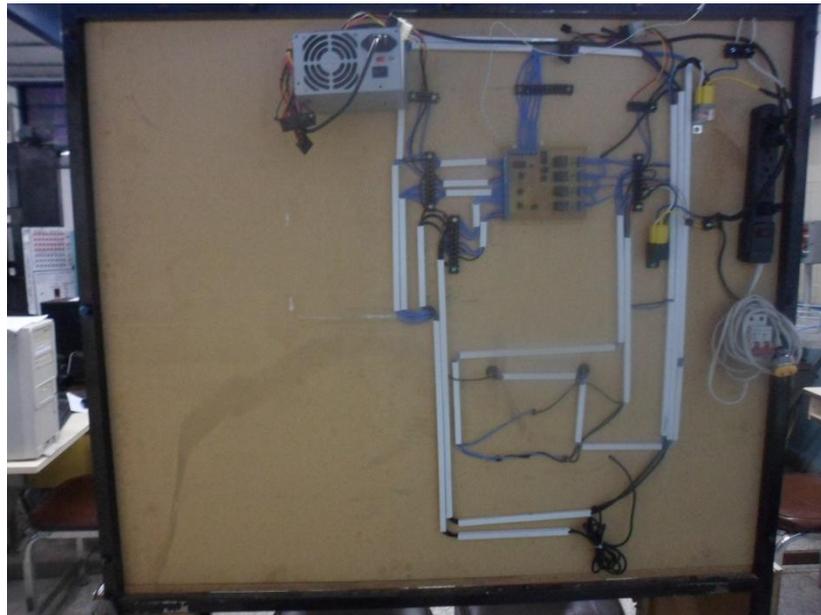


Figura C.7. Circuitería del banco de control

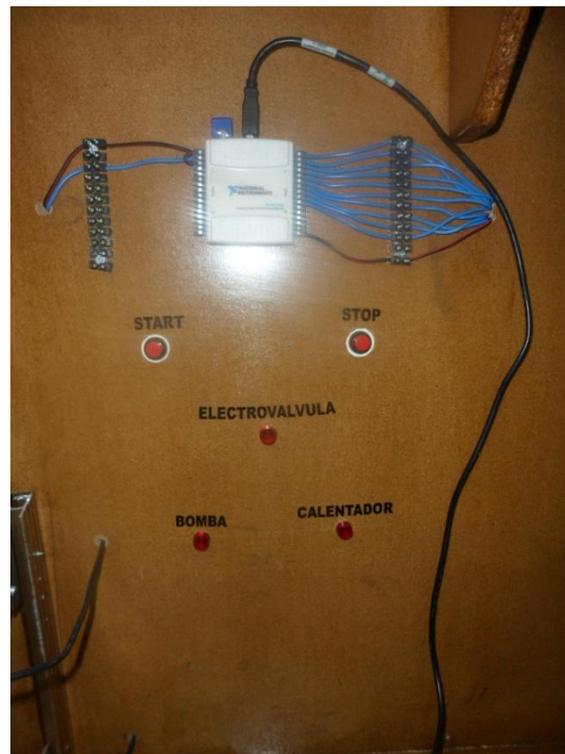


Figura C.8. Panel indicador - Inicio (Start) y Parada (Stop) del Proceso Tarjeta de Adquisición de Datos.



Apéndice A

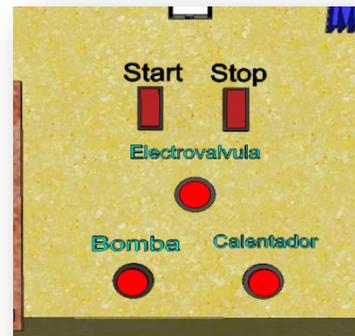
“Manual de usuario”

1. Descripción del Banco de Control

El banco de control es un modulo didáctico, que por medio de una Tarjeta de Adquisición de datos (DAQ), nos permite implementar conceptos de control industrial de manera sencilla e intuitiva.



Tarjeta de adquisición de datos (DAQ)

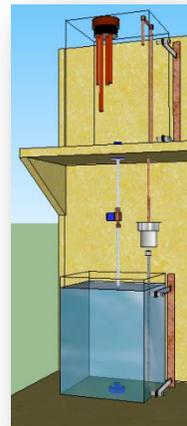


Panel de leds y pulsadores

y conexiones físicas.



Tarjeta de acople de Voltajes.



Simulador de nivel y temperatura



a) Tarjeta de adquisición de datos (DAQ) y conexiones

Esta permite la conexión entre las entradas y salidas de la DAQ (en este caso una NI USB-6008) con elementos como sensores, electroválvulas entre otros, a fin de ejecutar instrucciones que permiten darle un control a voluntad.

b) Panel de leds y pulsadores

Este nos permite interacción con el banco, con la posibilidad de arrancar o detener el proceso en un momento determinado, además de mostrarnos el estado de los elementos interactuantes durante el proceso.

c) Tarjeta de acople de Voltajes

Esta permite el debido acople entre la Tarjeta de adquisición de Datos (DAQ) y el sistema de control. En las especificaciones técnicas de la DAQ se muestra que su voltaje de trabajo nominal es de 5 VDC, sin embargo nuestro sistema de control como electroválvulas, sensores capacitivos, calentador y bomba trabajan en un voltaje de 12 a 110 V, haciéndonos así factible el control de procesos físicos desde el software LabVIEW®.

d) Simulador de nivel y temperatura

Esta instalación permite en menor escala simular un proceso vaciado y llenado de tanques, así como también variaciones de temperatura y registro de las mismas tal como se llevaría a cabo en una industria. La aplicación consta de un sistema de tanques y varios elementos sensores y actuadores (electroválvulas, sensores capacitivos, calentador y bomba, etc.) con los cuales se puede elaborar un proceso a través del software LabVIEW® y la DAQ.

2. Operaciones básicas

a) Encendido del banco.

Para el encendido del banco, conecte el cable de alimentación a la toma de 110 Va (preferiblemente con puesta a tierra), y coloque el breaker principal del banco (ubicado en la parte trasera del mismo), en posición de encendido (ON) con esto energizara la regleta donde está conectada la fuente de poder, el calentador y la bomba, también encender el computador ya sea laptop o desktop.



Cable y Breaker principal

b) Encendido de la tarjeta de adquisición de datos (DAQ)

Para encender la tarjeta de adquisición de datos es necesario solo conectar el cable USB (el cual está disponible en el laboratorio) al computador, esta encenderá un led verde en la parte superior indicando que esta energizado.



Encendido de la DAQ

c) Encendido de la tarjeta de acople

Para encender la tarjeta de acople solo es necesario encender la fuente de poder en el switch que se encuentra en la parte posterior de la misma, esta dará energía a la circuitería que se encuentra en ella, así como también a los relés y leds. Se encenderán los 4 leds que representan los relés de las salidas del sistema, indicándonos el estado de la tarjeta (On/Off).





Encendido tarjeta de Acople

NOTA:

- Siempre al usar el banco de control primero encender la tarjeta de adquisición conectándola al equipo y **luego** encender la fuente de poder que alimenta la tarjeta de acople, esto evita daños en la tarjeta de acople y en los relés que controlan las salidas del sistema.
- Al apagar el banco de control seguir el siguiente orden: apagar la fuente de poder, luego desconectar el cable de alimentación y por ultimo desconectar la tarjeta de adquisición (DAQ)

3. Puesta en marcha del banco

a) Proceso de vaciado y llenado de tanque

Para operar y poner en marcha el proceso de vaciado y llenado de tanques en el banco realice lo siguiente:

- I. Conecte la tarjeta de adquisición de datos al computador vía USB. Así como también el banco de procesos y la tarjeta de acople.
- II. Llenar de agua el tanque A.
- III. Verificar que al llenar el tanque A, los sensores capacitivos detectan el líquido, si al aproximarlos se enciende el led interno que poseen.
- IV. Ejecutar el proceso programado.
- V. Al finalizar apague la fuente de poder que alimenta la tarjeta de acople, abra las válvulas manuales para vaciar el agua del tanque A sobre un recipiente



- VI. Por último desconecte el cable de alimentación del banco desenergizando por completo el mismo.

NOTA:

- Cuando el banco de control este energizado **evite tocar los componentes de la parte inferior del banco**, así como también desenergizar la tarjeta de adquisición (DAQ).
- En el caso en que el líquido llegase a quedar en la parte superior, es factible hacer un “VI” sencillo conectado a DAQ que permita activar la salida correspondiente a la electroválvula.

b) Proceso de vaciado y llenado de tanque mediante variación de temperatura

Para operar y poner en marcha el proceso de vaciado y llenado de tanques en el banco realice lo siguiente:

- I. Conecte la tarjeta de adquisición de datos al computador vía USB. Así como también el banco de procesos y la tarjeta de acople.
- II. Llenar de agua el tanque A.
- III. Verificar que al llenar el tanque A, los sensores capacitivos detectan el líquido, si al aproximarlos se enciende el led interno que poseen.
- IV. Ejecutar el proceso programado.
- V. Al finalizar apague la fuente de poder que alimenta la tarjeta de acople, abra las válvulas manuales para vaciar el agua del tanque A sobre un recipiente
- VI. Por último desconecte el cable de alimentación del banco desenergizando por completo el mismo.

NOTA:

- Cuando el banco de control este energizado **evite tocar los componentes de la parte inferior del banco**, así como también desenergizar la tarjeta de adquisición (DAQ).
- En el caso en que el líquido llegase a quedar en la parte superior, es factible hacer un “VI” sencillo conectado a DAQ que permita activar la salida correspondiente a la electroválvula.
- Cuando se encienda el calentador, “**evitar a toda costa tocar el mismo**”, este podría generar quemaduras en su piel.

4. Guía de programación de NI USB-6008

La tarjeta de adquisición de datos es una NI USB-6008 de la Nacional Instrument, que dispone de 8 canales de entrada analógica de 12 bits, 12 líneas DIO, 2 salidas analógicas, 1 contador de 32 bits. La apariencia física de la DAQ es la siguiente:



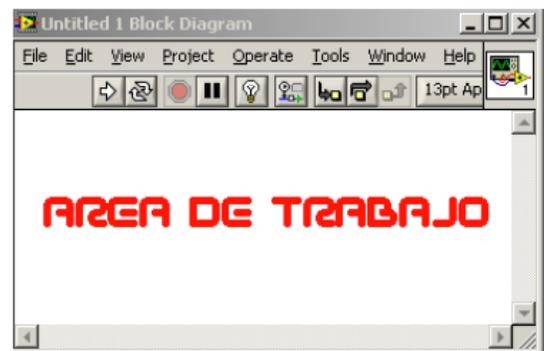
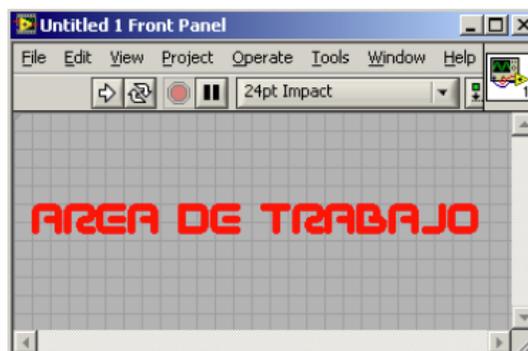
Para Programar la tarjeta de adquisición se utiliza el software LabVIEW®, la versión instala es la 9.0 de 32Bits (ver apartado 2.2.3.4 “¿Cómo trabaja LabVIEW®? del capítulo II”) a continuación se dará una breve introducción al software.

a) Edición de un programa

Al arrancar el software a través del acceso directo o del menú de Inicio» Programas» Nacional Instruments LabVIEW 9.0, se nos presenta la siguiente ventana en donde se da inicio a la construcción del VI.



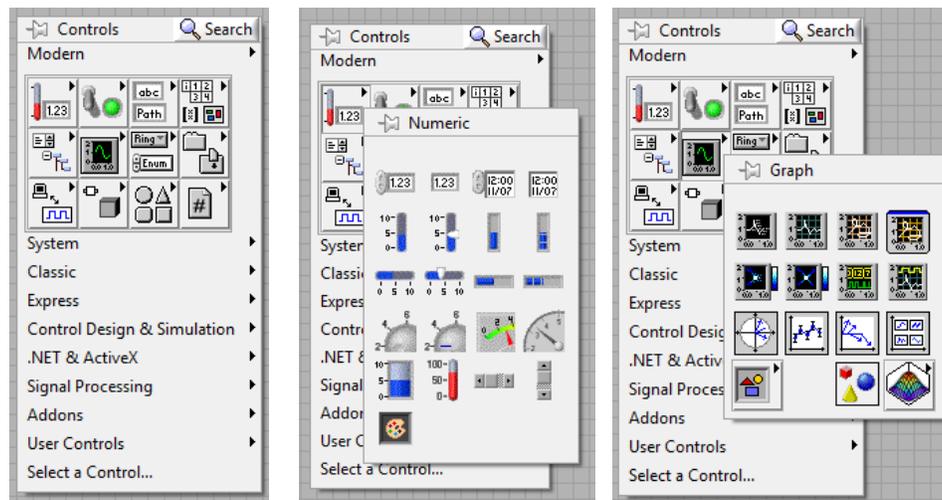
Al hacer clic en “Blank VI”, automáticamente se abren 2 ventanas. Una contiene al Panel Frontal (izquierda) y la otra al Diagrama de bloque (derecha).



controles e indicadores que se necesitan para crear el panel frontal. En pantalla aparecen

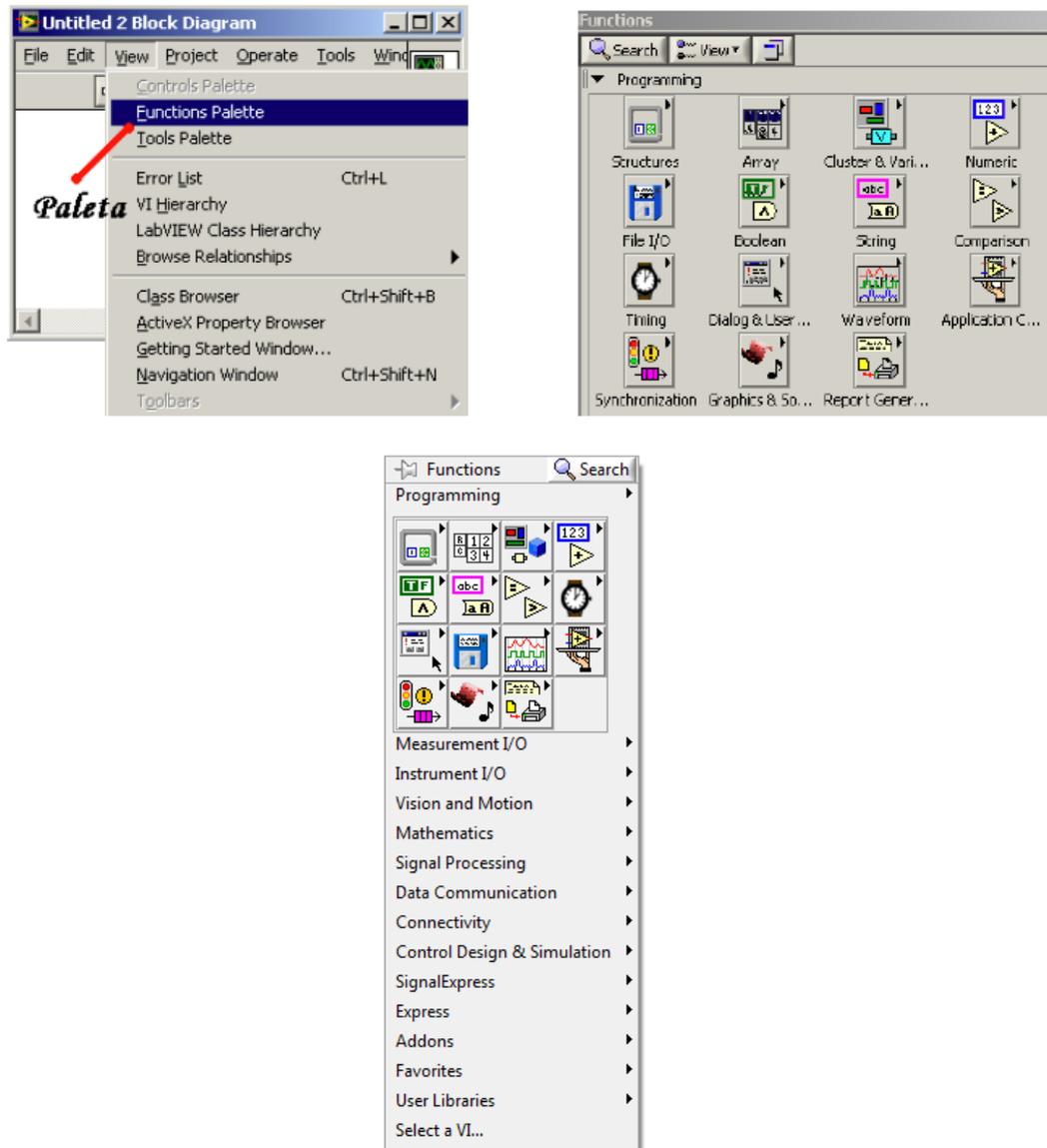
aparece haciendo clic con el botón derecho del mouse sobre el área de trabajo del panel frontal.

Se utiliza únicamente en el *panel frontal*. Contiene todos los *controles e indicadores* que se emplearán para crear la interfaz del VI con el usuario.



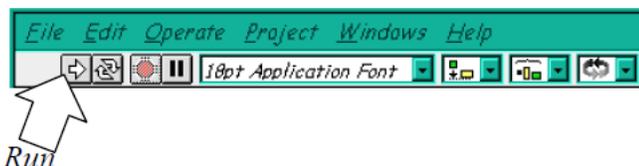
- **Paleta de funciones (Functions Palette)**

Se emplea en el diseño del *diagrama de bloques*. La *paleta de funciones* contiene todos los objetos que se emplean en la implementación del programa del VI, ya sean *funciones* aritméticas, de entrada/salida de señales, entrada/salida de datos a fichero, adquisición de señales, temporización de la ejecución del programa.



b) Ejecución del programa

Una vez se ha concluido la programación del VI sin errores se debe proceder a su ejecución. Para ello la ventana activa debe ser el *panel frontal* (si se está en la ventana del *diagrama de bloques*, se debe seleccionar la opción *Show Panel* del menú *Window*). Una vez situados en el *panel frontal*, se pulsará el botón de *Run*, situado en la barra de herramientas.



- | | |
|---|--|
| <i>Run</i> : ejecución | <i>Stop</i> : finaliza la ejecución. |
| <i>Continuamente Run</i> : ejecución continua | <i>Pausa</i> : detención del programa. |
| Error | Ejecutándose (ambos modos) |

El programa comenzará a ejecutarse. Mientras dura la ejecución del mismo, la apariencia del botón de *Run* es la que se muestra a continuación:

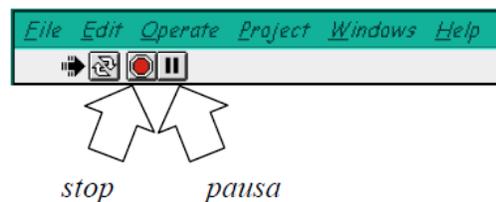


De este modo el programa se ejecutará una sola vez. Si se desea una ejecución continua, se pulsará el botón situado a la derecha del de *Run* (*Continuous Run*). Si durante el funcionamiento continuo del programa se vuelve a pulsar el citado botón, se finalizará la última ejecución del mismo, tras lo cual el programa se parará.



Continuous Run

Para finalizar la ejecución de un programa se puede operar de dos formas. La primera, y la más aconsejable, es emplear un botón en el *panel frontal* del VI, cuya pulsación produzca la interrupción del bucle de ejecución de la aplicación. La segunda forma de detener la ejecución del VI es pulsando el botón de *pausa* o el de *stop*. La diferencia entre ambos es que si se pulsa *stop*, la ejecución del programa finaliza inmediatamente, mientras que si se pulsa *pausa*, se produce una detención en el funcionamiento del programa, retomándose su ejecución una vez se vuelve a pulsar el mismo botón.



Si existe error, se puede pulsar el mismo botón de Run y nos muestra el tipo de error que se está cometiendo (líneas rotas, conexiones malas, etc.). Según la necesidad del programador, este puede colocar un botón de stop en el panel frontal de su VI, cuya pulsación provoque una interrupción en el bucle de ejecución de la aplicación.



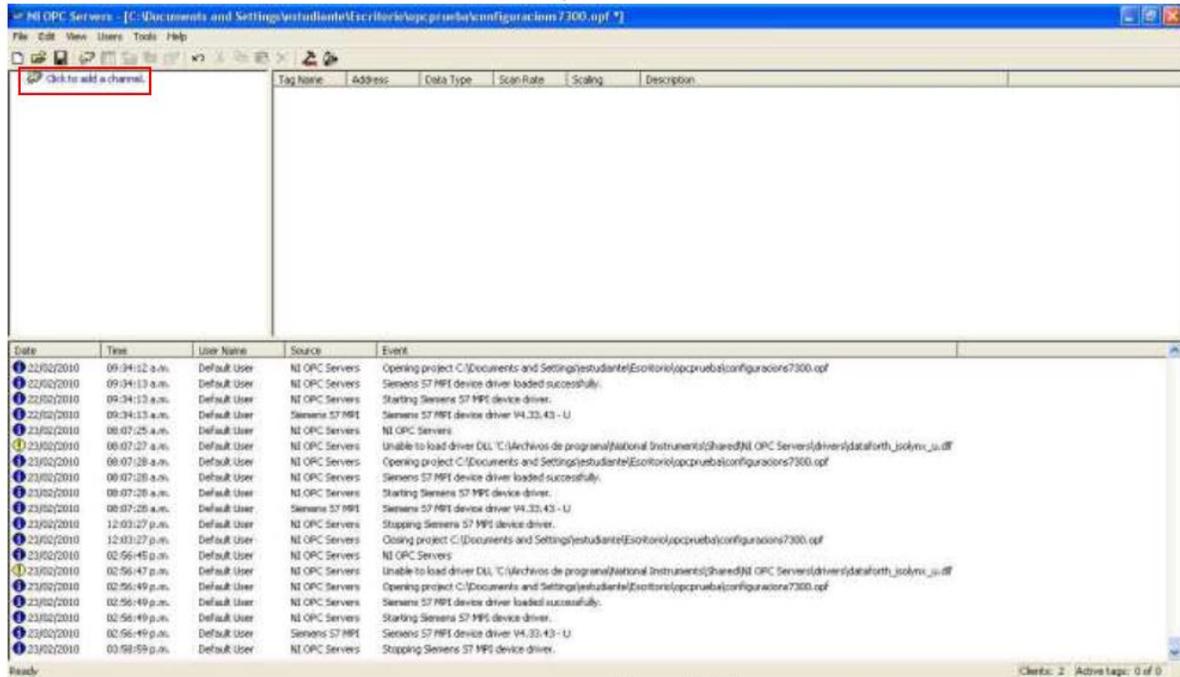
Apéndice E

“Manual para comunicación OPC SERVE con LabVIEW®”

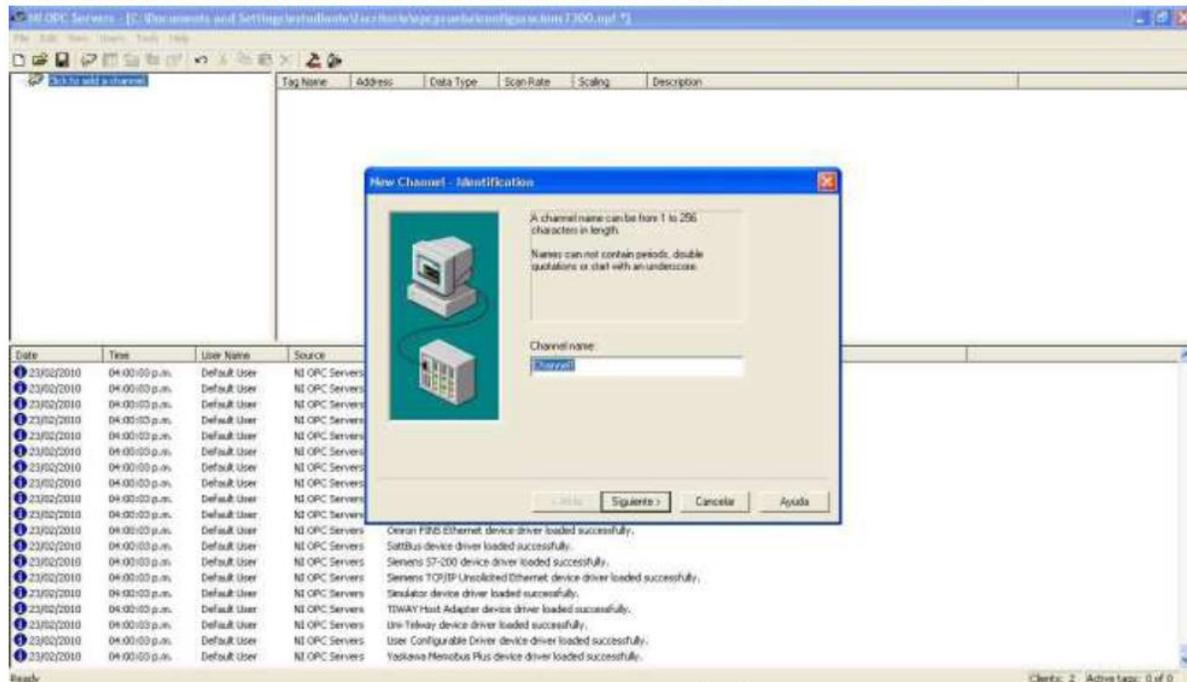
Tutorial

Configuración del Servidor NI OPC SERVER:

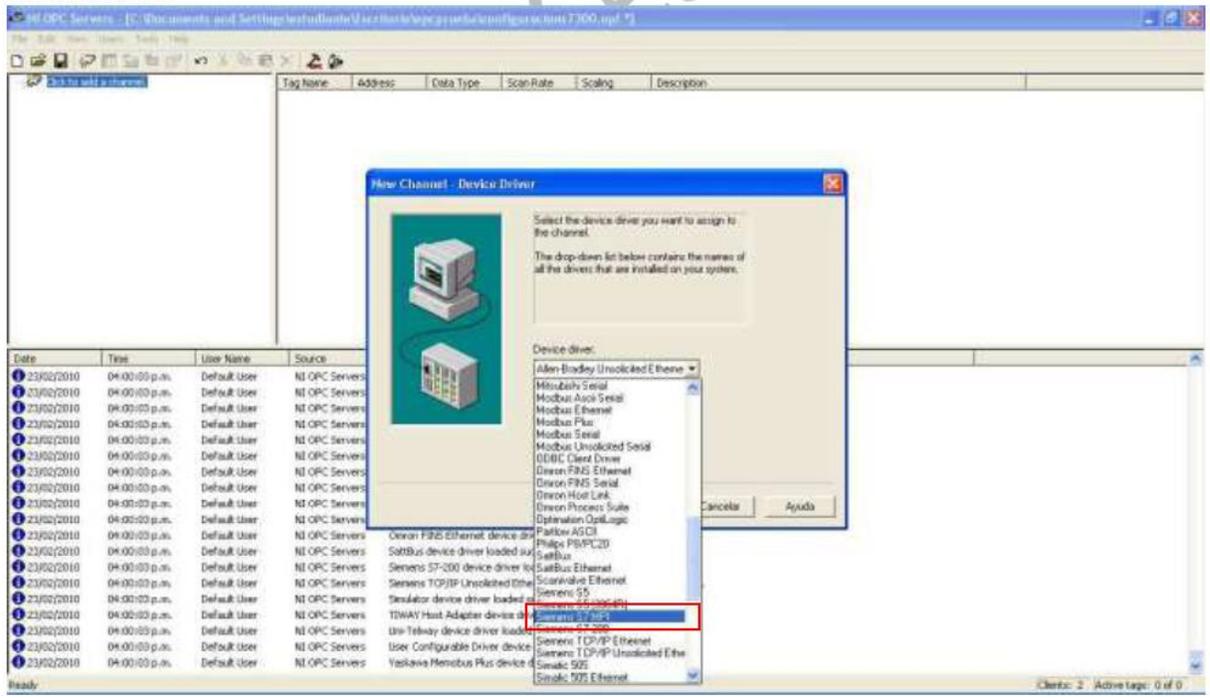
PASO 1: UNA VEZ INICIADO EL ADMINISTRADOR, DAR DOBLE CLIC EN ADICIONAR CANAL



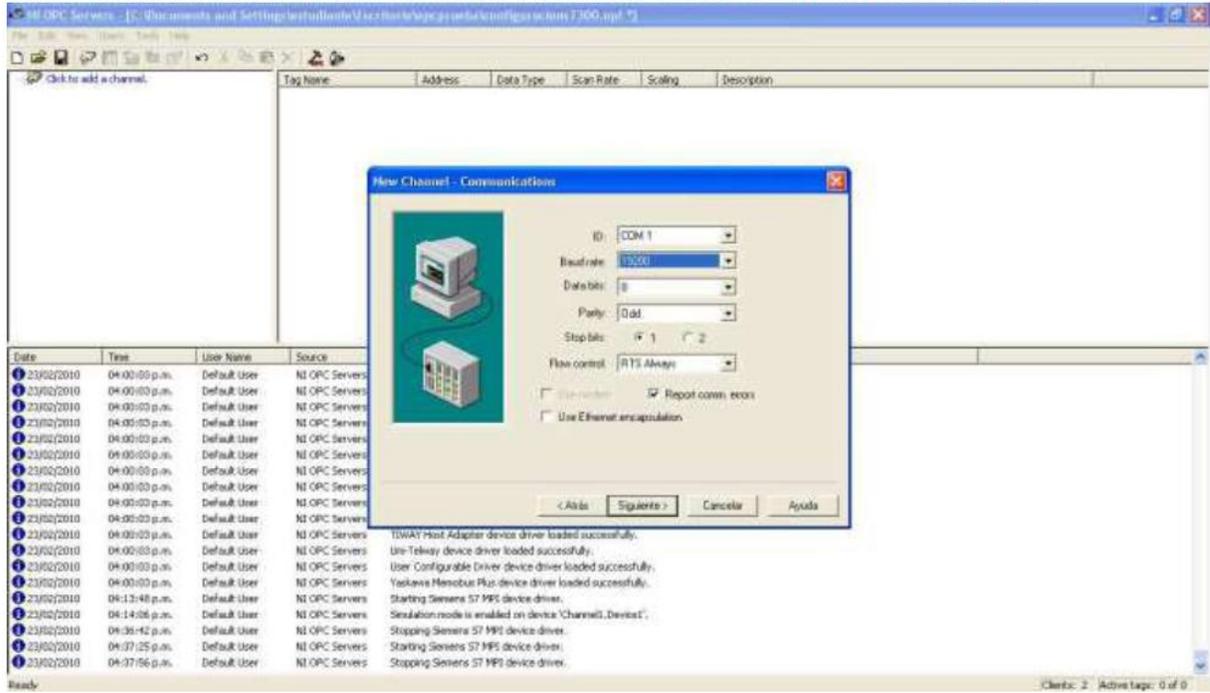
PASO 2: SE DESIGNA UN NOMBRE PARA EL CANAL DE ACUERDO A SU PREFERENCIA.



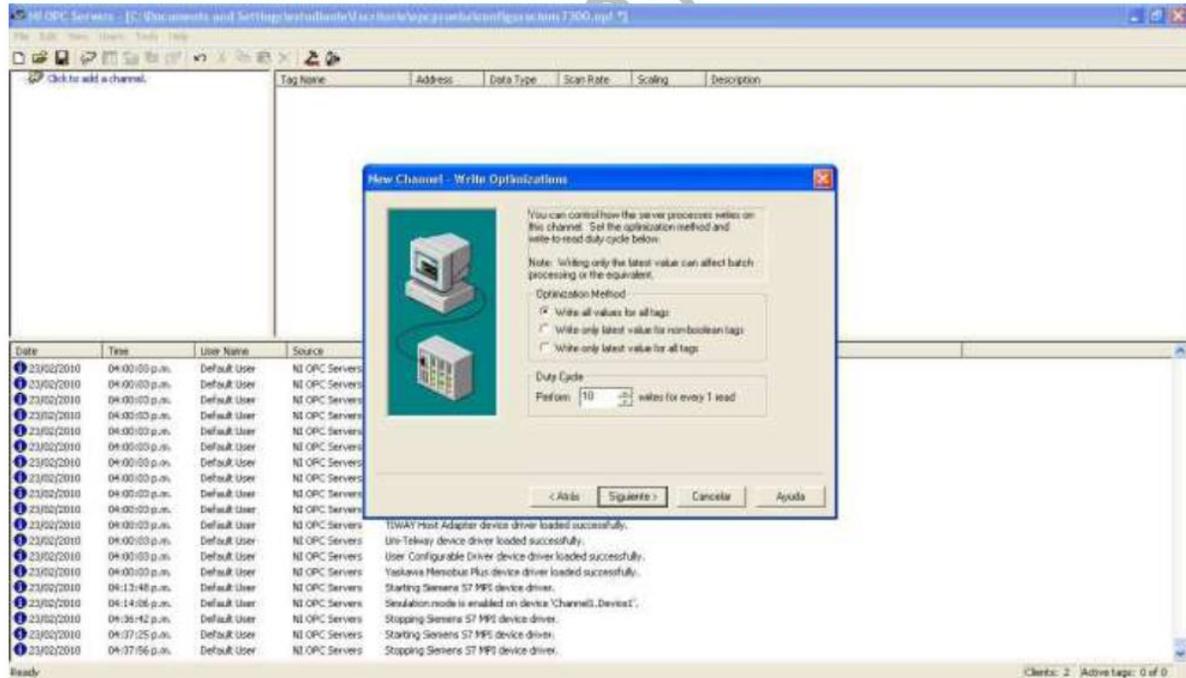
PASO 3: EN ESTE PASO SE DEBERA BUSCAR LA REFERENCIA DEL PLC O DISPOSITIVO QUE SE VA A ADMINISTRAR



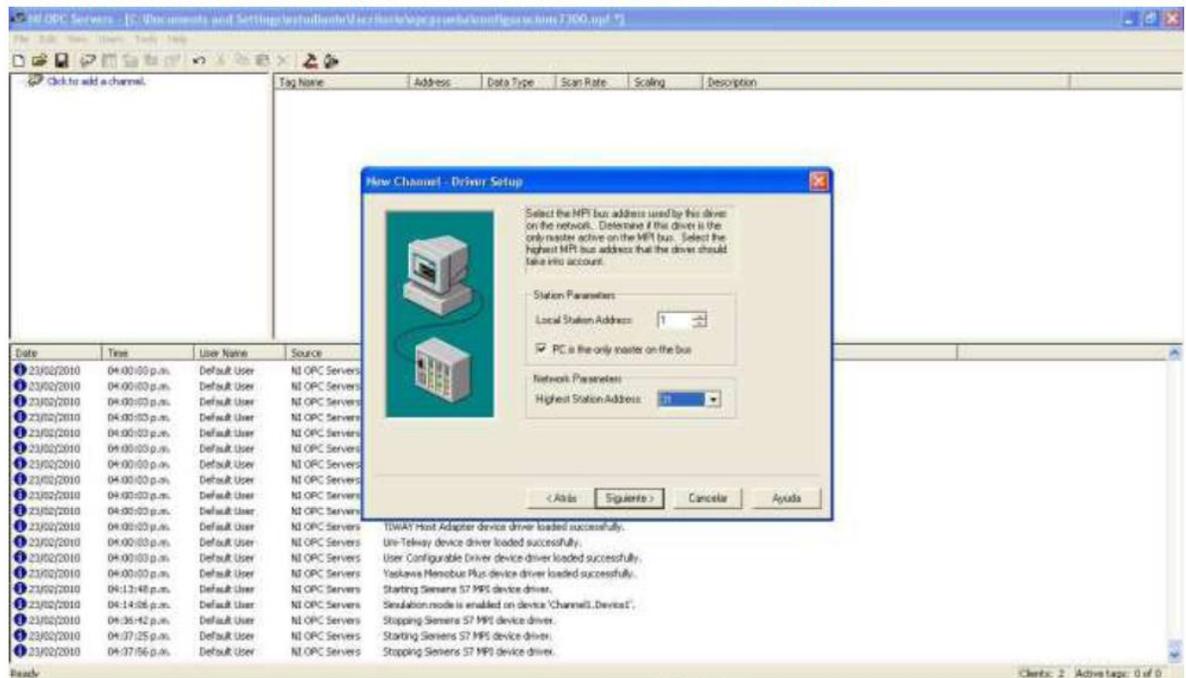
PASO 4: SE DEBERAN DEFINIR LOS PARAMETROS COMO SE MUESTRA A CONTINUACION.



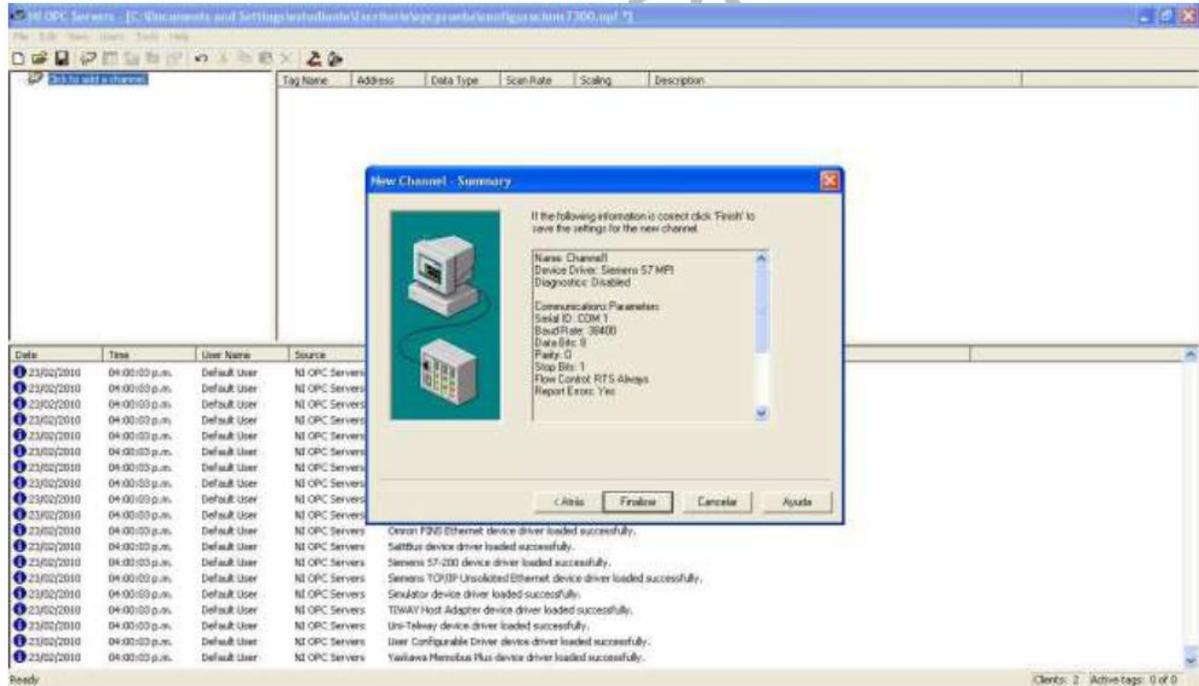
PASO 5: CONFIGURAR EL METODO DE OPTIMIZACION COMO SE MUESTRA.



PASO 6: CONFIGURAR LOS PARAMETROS DE LA ESTACION COMO SE MUESTRA.

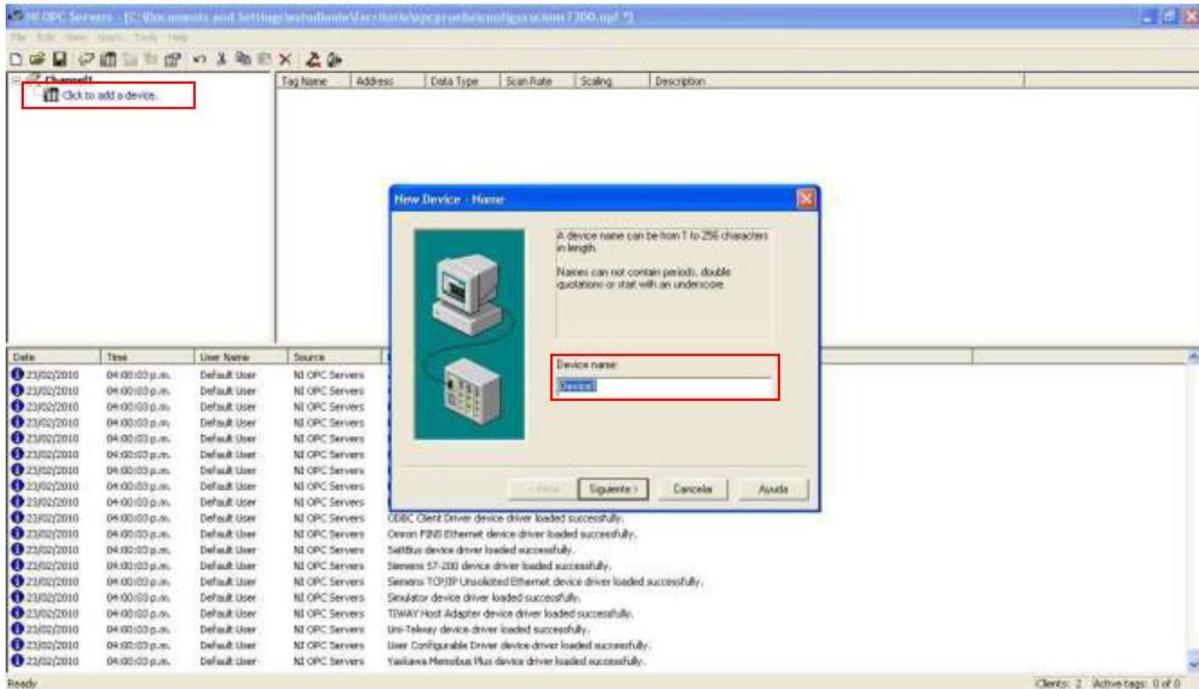


A CONTINUACION SE DA CLIC EN SIGUIENTE Y LUEGO EN FINALIZAR.

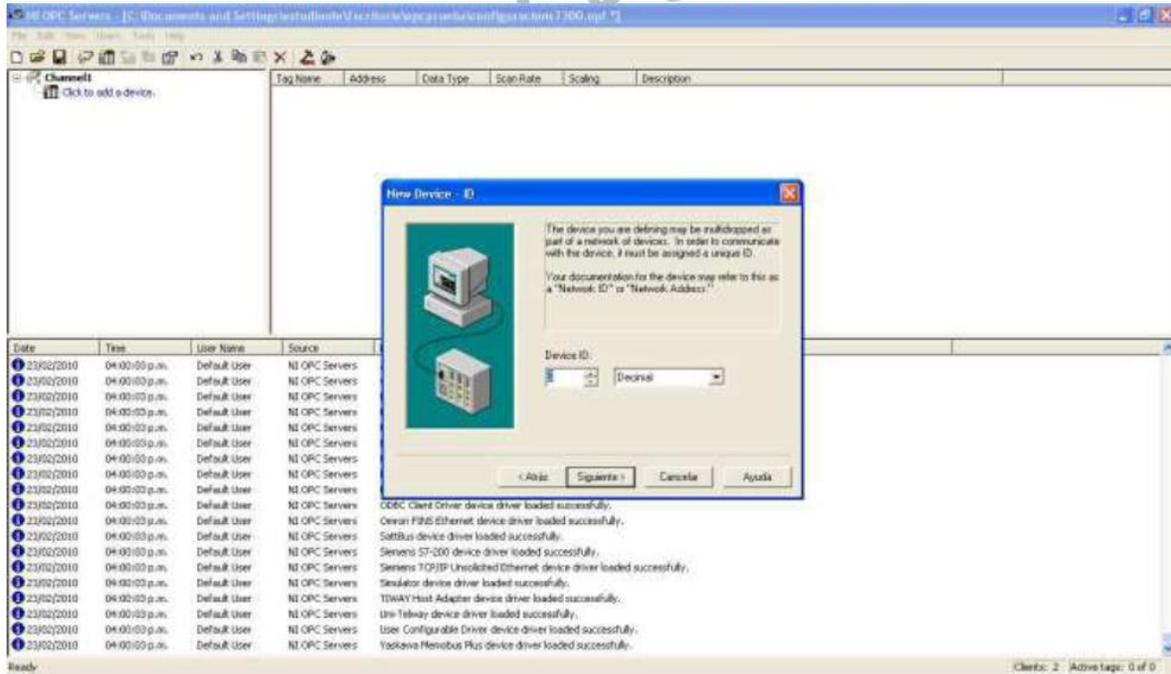


Agregando dispositivo al Servidor NI OPC SERVE:

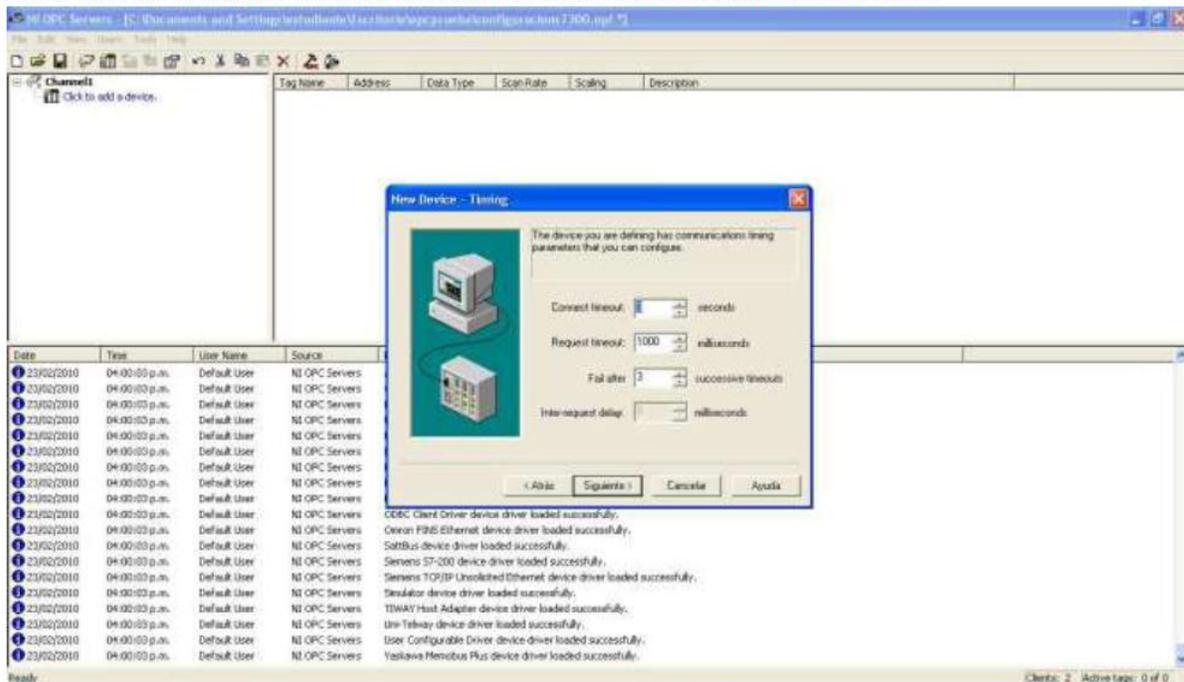
PASO 7: A CONTINUACION SE DEBE DAR CLIC EN ADICIONAR DISPOSITIVO Y LUEGO AGREGAR UN NOMBRE A ESTE ELEMENTO.



PASO 9: CONFIGURAR LA IDENTIFICACION DEL DISPOSITIVO COMO SE MUESTRA:

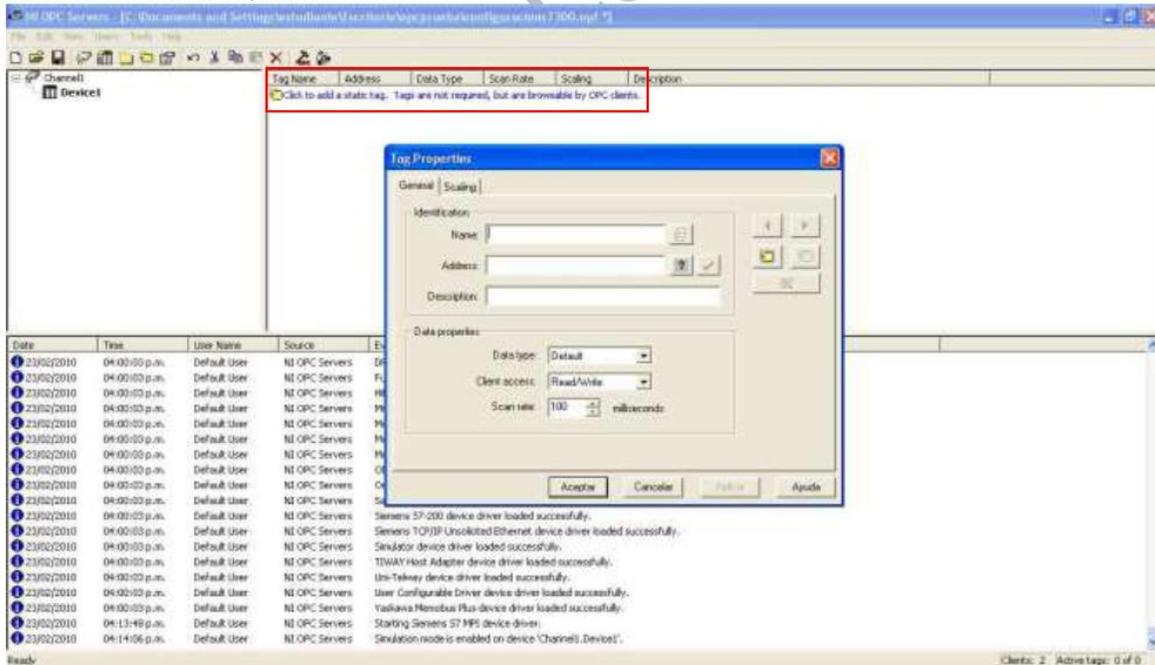


PASO 8: CONFIGURAR LA IDENTIFICACION DEL DISPOSITIVO COMO SE MUESTRA:

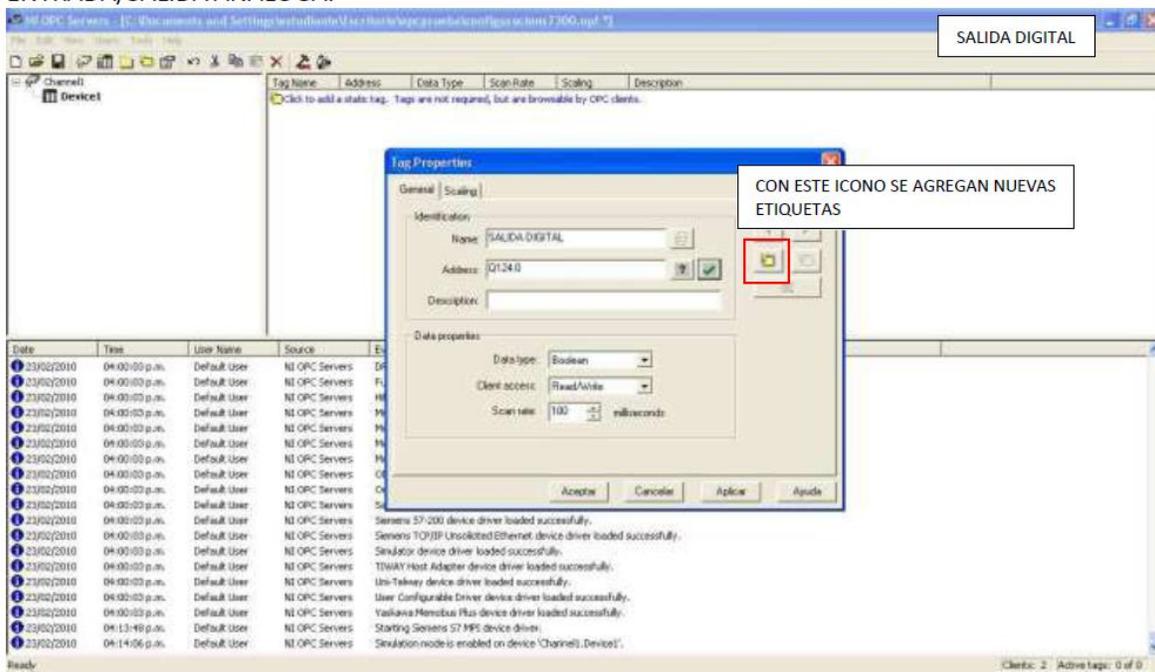


Agregando etiquetas para la identificación.

PASO 11: PARA AGREGAR ETIQUETAS QUE IMPORTEN LA INFORMACION, SE DEBE DAR CLIC EN ADICIONAR UNA ETIQUETA:



PASO 12: EN ESTE PASO SE CONFIGURARA CADA ETIQUETA PARA ENTRADA/SALIDA DIGITAL, ENTRADA/SALIDA ANALOGA.





ENTRADA DIGITAL

Tag Name	Address	Data Type	Scan Rate	Scaling	Description
ENTRADA DIGITAL	124.0	Boolean	100	None	
SALIDA DIGITAL	124.0	Boolean	100	None	

Date	Time	User Name	Source
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:13:49 p.m.	Default User	NI OPC Servers
23/02/2010	04:14:06 p.m.	Default User	NI OPC Servers

Ready

Client: 2 Active tags: 0 of 0

ENTRADA ANALOGA

Tag Name	Address	Data Type	Scan Rate	Scaling	Description
ENTRADA ANALOGA	124.0	Word	100	None	
ENTRADA DIGITAL	124.0	Boolean	100	None	
SALIDA DIGITAL	124.0	Boolean	100	None	

Date	Time	User Name	Source
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:00:00 p.m.	Default User	NI OPC Servers
23/02/2010	04:13:49 p.m.	Default User	NI OPC Servers
23/02/2010	04:14:06 p.m.	Default User	NI OPC Servers

Ready

Client: 2 Active tags: 0 of 0



The screenshot shows the NI OPC Servers interface. A 'Tag Properties' dialog box is open for the tag 'SALIDA ANALOGA'. The 'General' tab is selected, showing the following configuration:

- Name: SALIDA ANALOGA
- Address: M#2
- Description: (empty)
- Data type: Word
- Client access: Read/Write
- Scan rate: 100 milliseconds

The background shows a table of tags and a log of events.

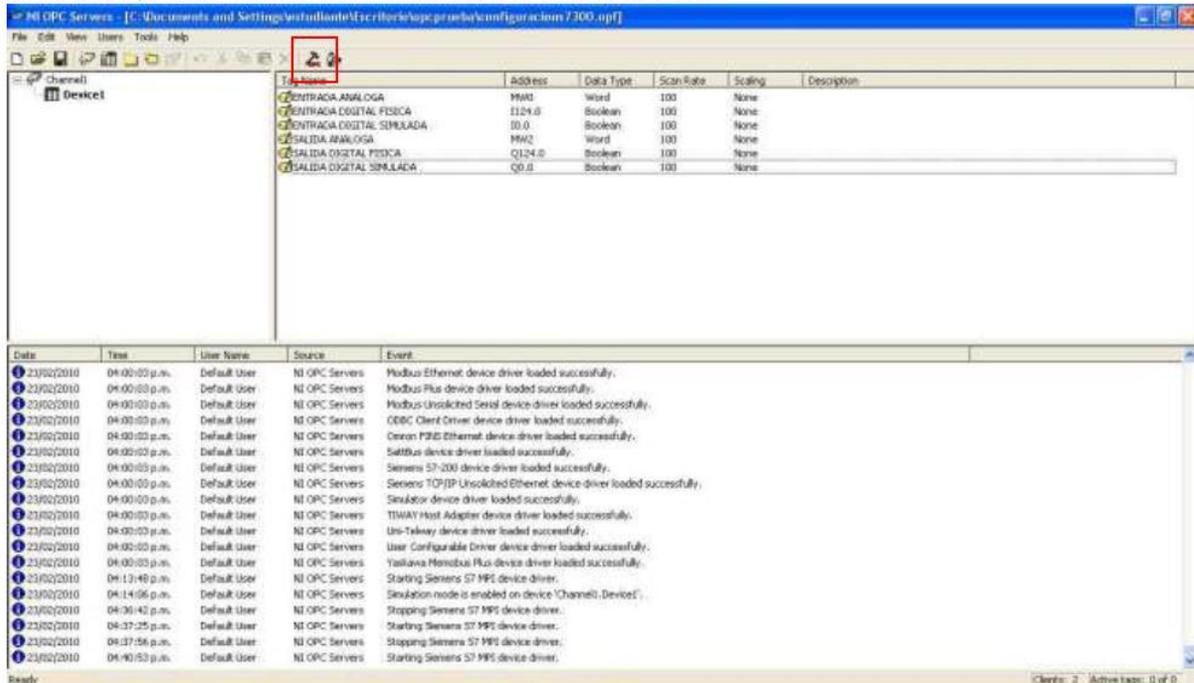
Tag Name	Address	Data Type	Scan Rate	Scaling	Description
ENTRADA ANALOGA	M#0	Word	100	None	
ENTRADA DIGITAL	I124.0	Boolean	100	None	
SALIDA ANALOGA	M#2	Word	100	None	
SALIDA DIGITAL	Q124.0	Boolean	100	None	

FINALMENTE SE DEBERA ESTABLECER LA CONFIGURACION COMO SE MUESTRA A CONTINUACION:

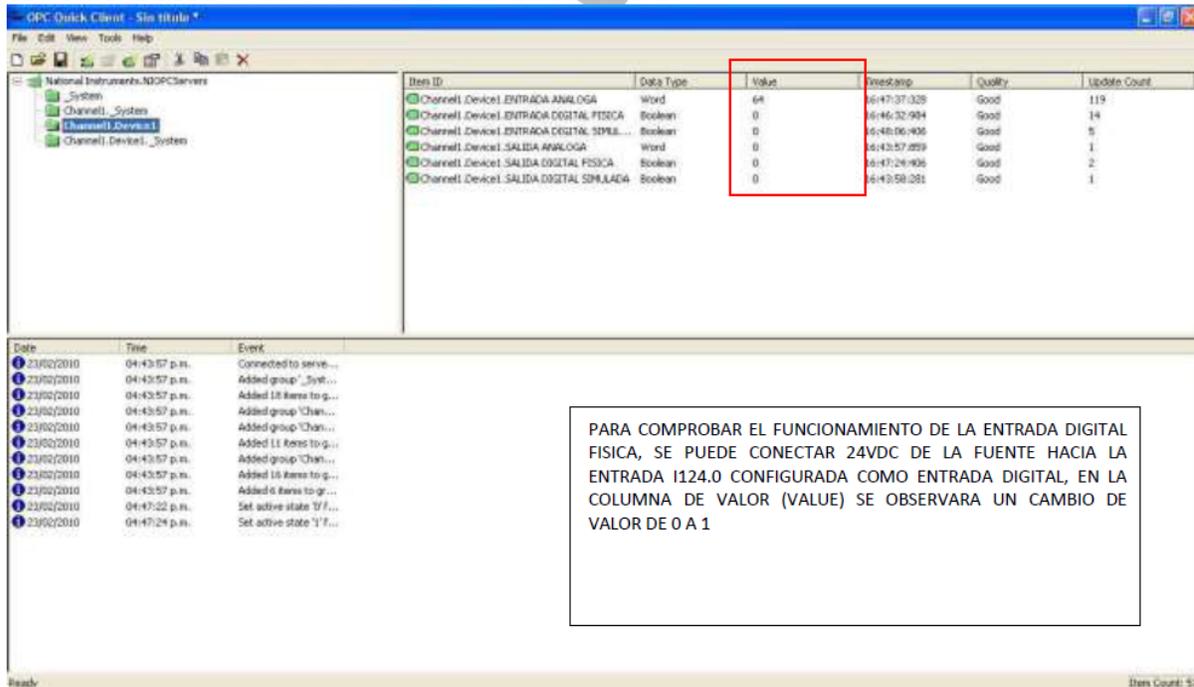
The screenshot shows the final configuration of the NI OPC Servers. The tag list is updated to include simulated digital tags. The 'Event' log shows the successful loading of various device drivers.

Tag Name	Address	Data Type	Scan Rate	Scaling	Description
ENTRADA ANALOGA	M#0	Word	100	None	
ENTRADA DIGITAL FISICA	I124.0	Boolean	100	None	
ENTRADA DIGITAL SIMULADA	I0.0	Boolean	100	None	
SALIDA ANALOGA	M#2	Word	100	None	
SALIDA DIGITAL FISICA	Q124.0	Boolean	100	None	
SALIDA DIGITAL SIMULADA	Q0.0	Boolean	100	None	

PASO 13: A CONTINUACION SE DEBE COMPROBAR LA CONFIGURACION CORRECTA DE LAS ETIQUETAS, PARA ESTO SE DEBE DAR CLIC EN QUICK CLIENT.



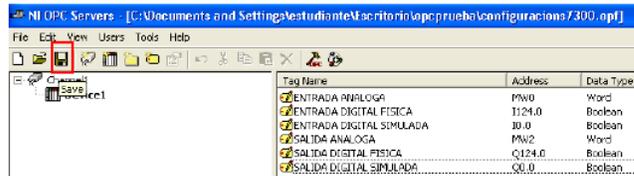
PASO 16: PARA COMPROBAR EL FUNCIONAMIENTO DE LAS ETIQUETAS CREADAS, SE DEBERA BUSCAR EL NOMBRE DEL DISPOSITIVO CREADO ANTERIORMENTE.



PARA COMPROBAR EL FUNCIONAMIENTO DE LA ENTRADA DIGITAL FISICA, SE PUEDE CONECTAR 24VDC DE LA FUENTE HACIA LA ENTRADA I124.0 CONFIGURADA COMO ENTRADA DIGITAL, EN LA COLUMNA DE VALOR (VALUE) SE OBSERVARA UN CAMBIO DE VALOR DE 0 A 1



PASO 14: FINALMENTE, ESTA VENTANA ANTERIOR SE PUEDE CERRAR SIN NECESIDAD DE GUARDAR LOS DATOS. SIN EMBARGO SI SE DEBE GUARDAR LA CONFIGURACION HECHA EN NI OPC SERVER, PARA ELLO SE DARA CLIC EN GUARDAR (SAVE). UNA VEZ GUARDADA LA INFORMACION SE PUEDE CERRAR EL NI OPC SERVER.



Configuración de LabVIEW para supervisar y programar el PLC.

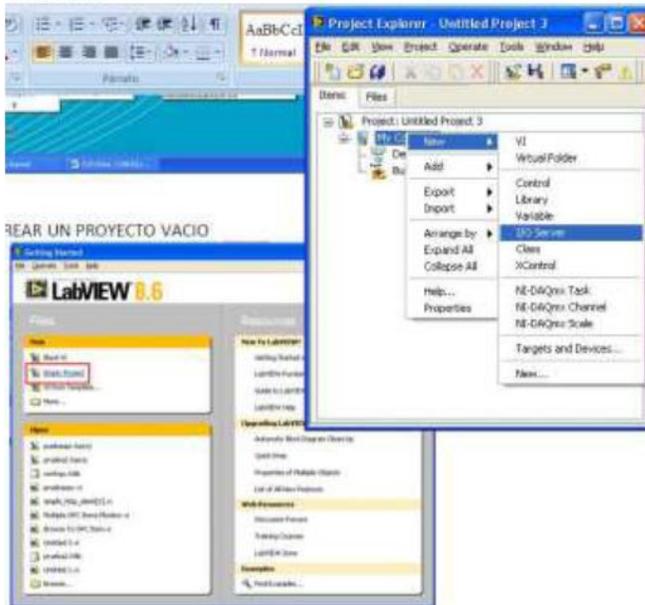
EN PRIMERA INSTANCIA, SE DEBERA ACCEDER AL PROGRAMA MEDIANTE EL ICONO UBICADO EN EL ESCRITORIO O POR LA BARRA DE INICIO

INICIO – PROGRAMAS – NATIONAL INSTRUMENTS – LabVIEW (según versión) – LabVIEW – ENTER

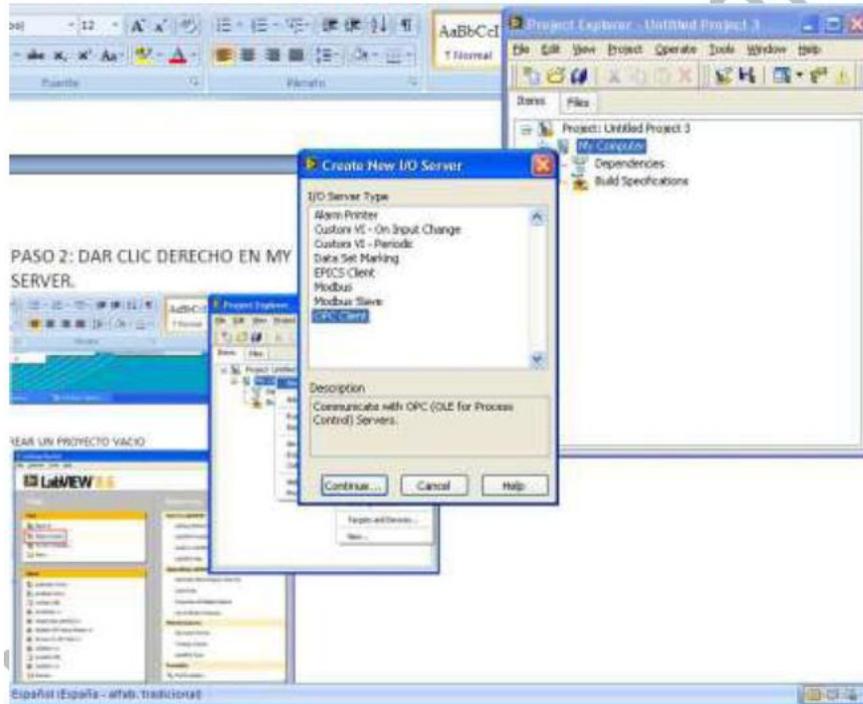
PASO 1: SE DEBERA CREAR UN PROYECTO VACIO



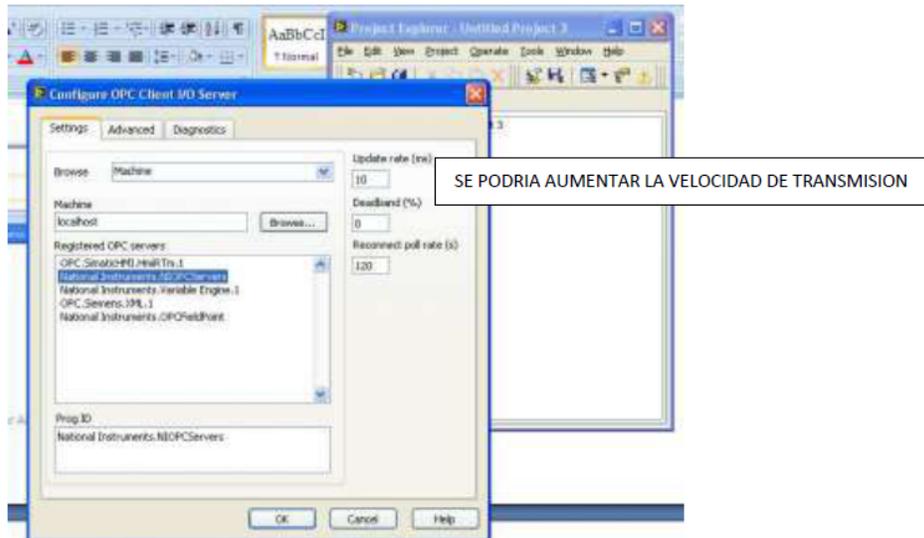
PASO 2: DAR CLIC DERECHO EN MY COMPUTER, LUEGO DIRIGIRSE A NEW, PARA SELECCIONAR I/O SERVER.



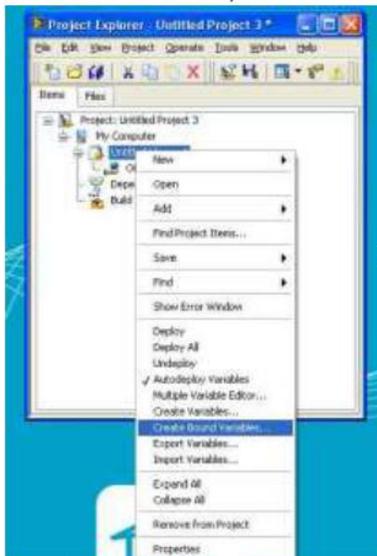
PASO 3: EN LA NUEVA VENTANA, SELECCIONAR OPC CLIENT, LUEGO PRESIONAR CONTINUE.



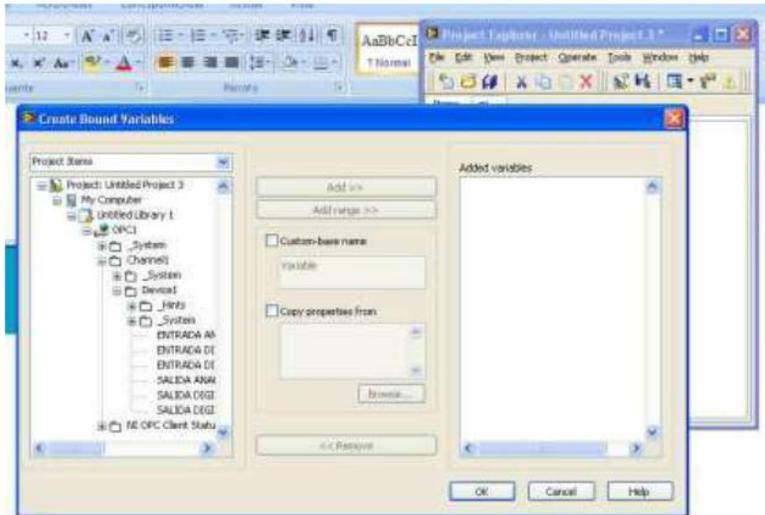
PASO 4: EN LA NUEVA VENTANA, SELECCIONAR NATIONAL INSTRUMENTS.NIOPCSERVERS, LUEGO DAR CLIC EN OK.



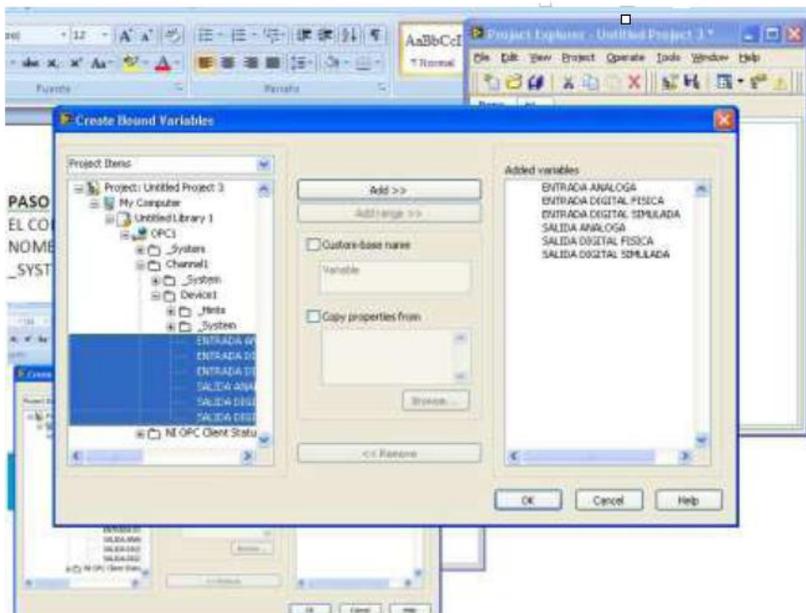
PASO 5: PARA VINCULAR LAS VARIABLES CREADAS ANTERIORMENTE, SE DEBE DAR CLIC DERECHO EN UNTITLED LIBRARY, LUEGO BUSCAR LA OPCION DE CREATE BOUND VARIABLE.



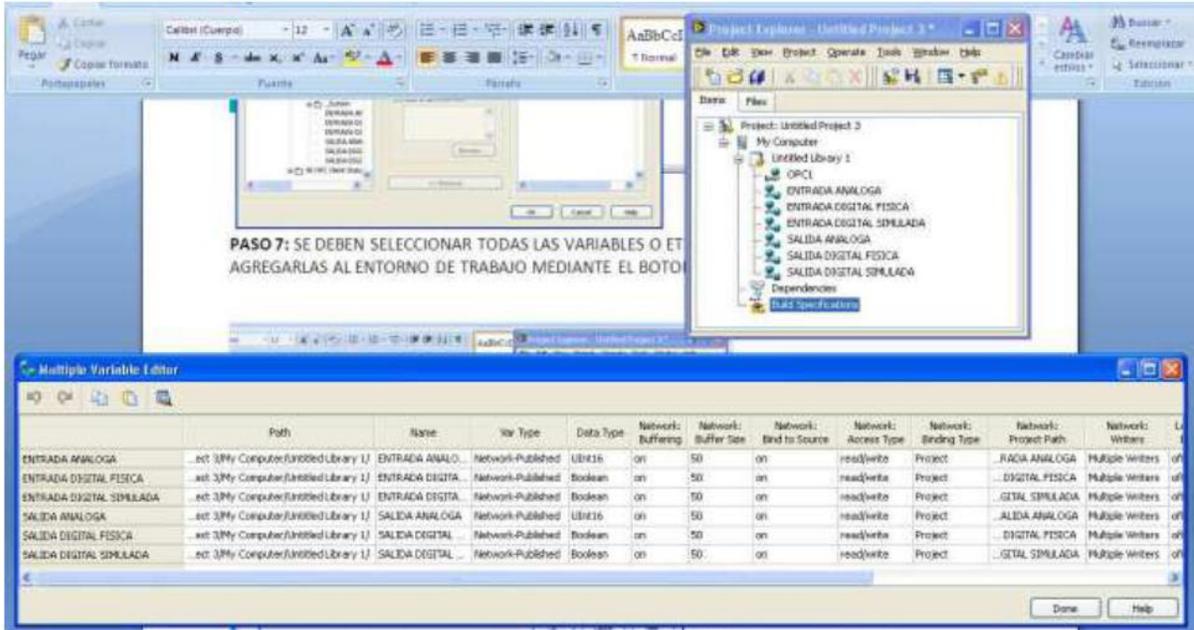
PASO 6: SE DEBE BUSCAR LA UBICACIÓN DE LAS VARIABLES CREADAS ANTERIORMENTE, SE DESPLAZA EL CONTENEDOR DE PROJECT, LUEGO MY COMPUTER, LUEGO UNTITLED LIBRARY 1, OPC1, EL NOMBRE DEL CANAL CREADO, EL NOMBRE DEL DISPOSITIVO CREADO, FINALMENTE DEBAJO DE _SYSTEM SE PODRAN APRECIAR LAS VARIABLES CREADAS EN EL NI OPC SERVER.



PASO 7: SE DEBEN SELECCIONAR TODAS LAS VARIABLES O ETIQUETAS CREADAS, PARA PODER AGREGARLAS AL ENTORNO DE TRABAJO MEDIANTE EL BOTON ADD>>. FINALMENTE DAR CLIC EN OK.



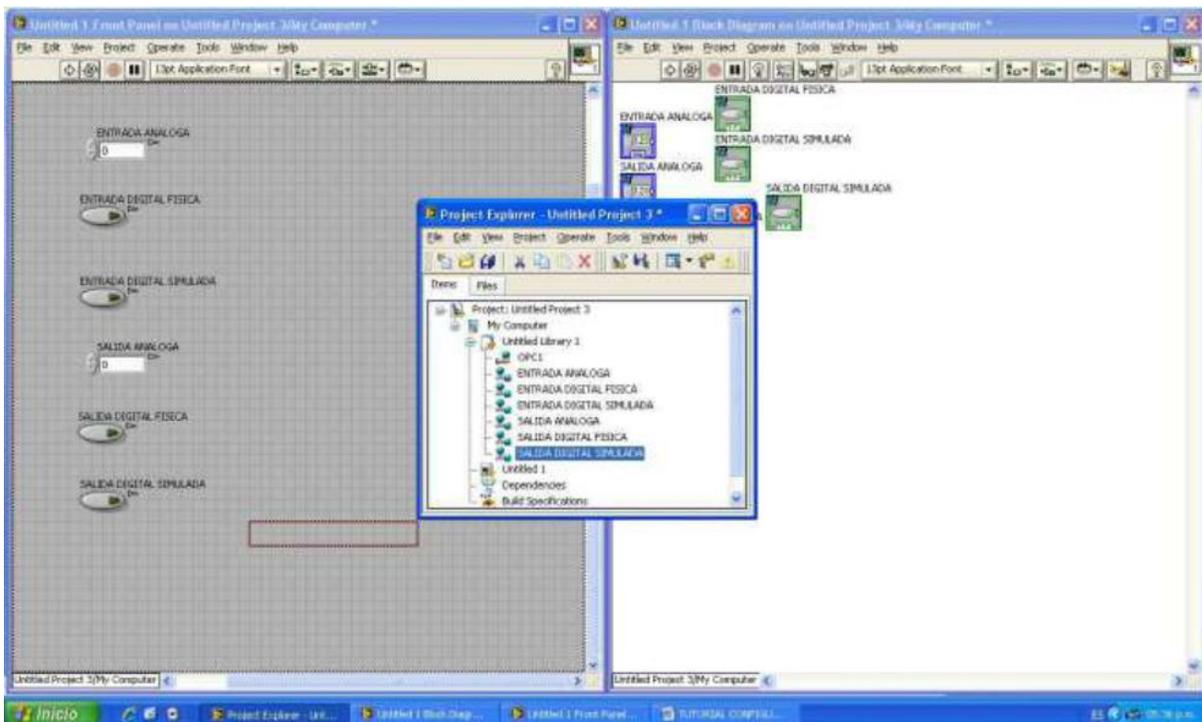
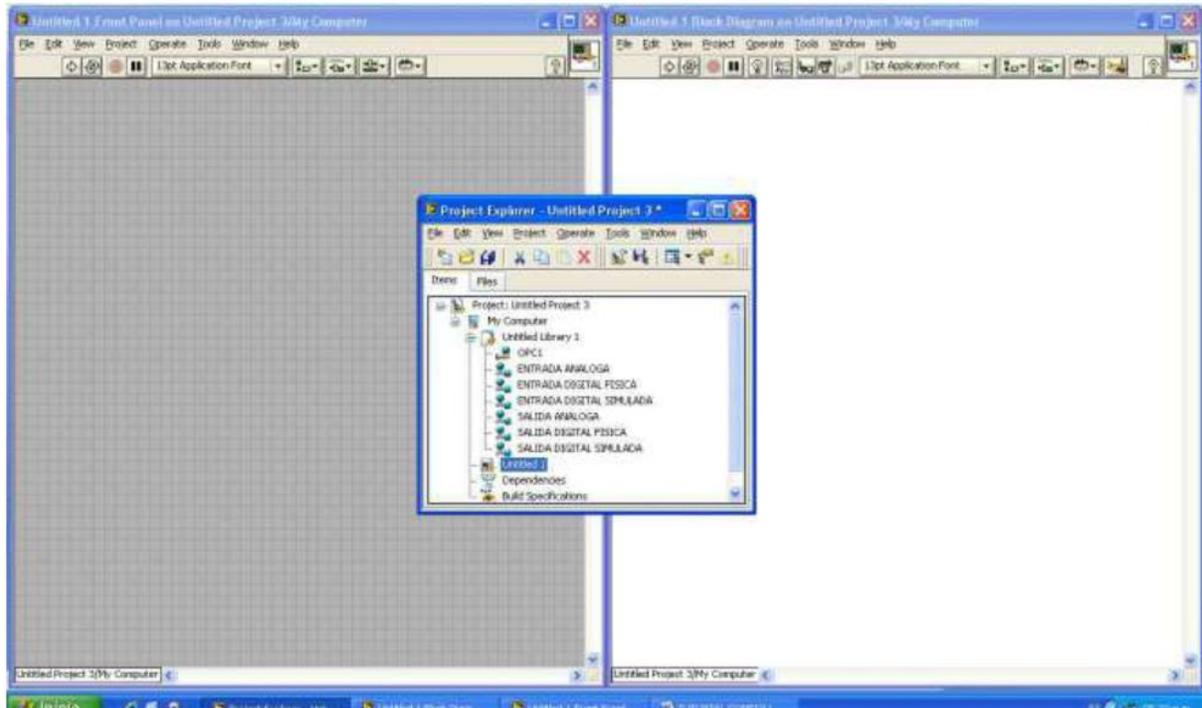
PASO 8: SE DEBERA VISUALIZAR UNA VENTANA QUE MUESTRA CADA VARIABLE Y SUS PROPIEDADES CONFIGURADAS PREVIAMENTE. DAR CLIC EN DONE.



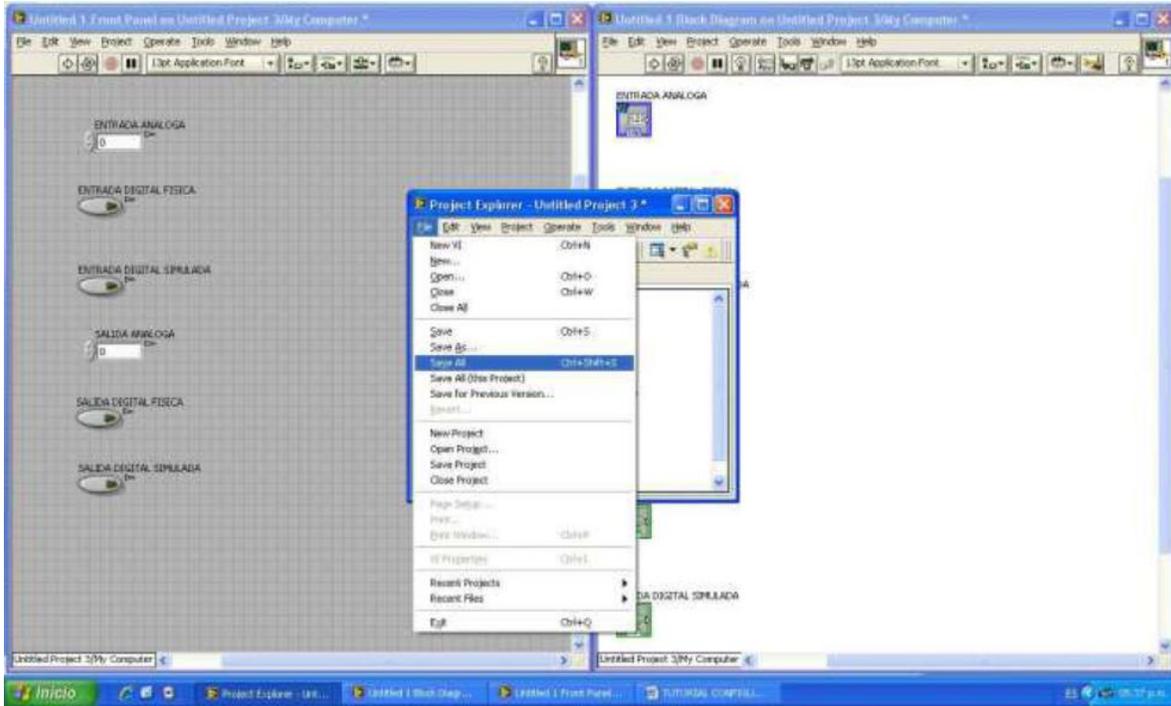
PASO 9: EN ESTE MOMENTO SE HAN AGREGADO LAS VARIABLES AL ENTORNO DE LABVIEW, SIN EMBARGO ES NECESARIO LLEVARLAS A UNA INTERFAZ DE PROGRAMACION/SUPERVISION. PARA ELLO, SE DARA CLIC DERECHO EN MY COMPUTER, LUEGO EN NEW, PARA TERMINAR CON VI.



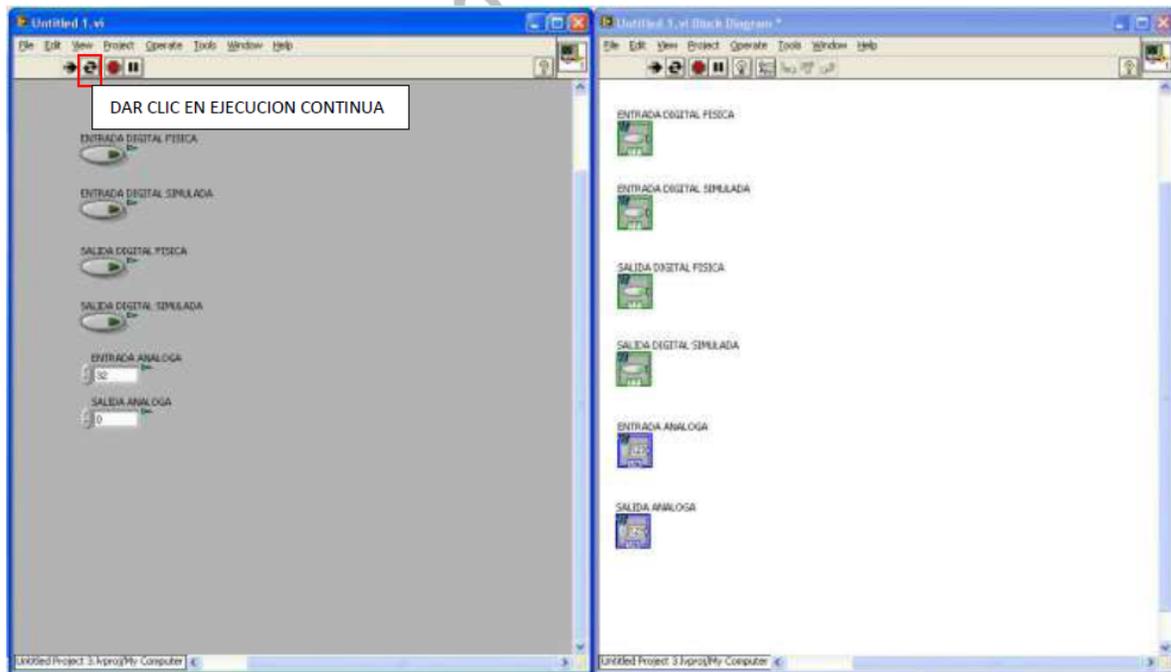
PASO 10: SE HA CREADO UN VI (CTRL + T PARA EXPANDIR VENTANAS). PARA USAR LAS VARIABLES CREADAS SE PUEDEN ELEGIR DEL PROJECT EXPLORER Y ARRASTRAR HACIA EL FRONT PANEL DEL VI.



PASO 11: ES IMPORTANTE GUARDAR TODO EL ENTORNO DE TRABAJO CREADO, PARA ESTO, SE DEBERA DAR CLIC EN FILE, LUEGO EN SAVE ALL Y ELEGIR UN NOMBRE Y DESTINO.



PASO 12: FINALMENTE SE PUEDEN ORDENAR LOS ELEMENTOS COMO SE MUESTRA A CONTINUACION PARA SU FUNCIONAMIENTO MEDIANTE LA IMPLEMENTACION NECESARIA PARA CADA ENTRADA O SALIDA DISPUESTA.





CONCLUSIONES

El estudio de los sistemas de control existentes es de alta importancia para los estudiantes de ingeniería, genera alto nivel de competitividad y respalda los buenos oficios de la docencia en nuestra casa de estudio.

- Se alcanzaron los objetivos planteados en el desarrollo del banco de control para el uso del software grafico aplicado, logrando de esta manera contar con una nueva herramienta para la diversificación de equipos didácticos del laboratorio de control de la escuela de Ingeniería Eléctrica de la Universidad de Carabobo.
- Se logro el poner en práctica, de manera interactiva, el concepto de control industrial al tener que implementar mediante el ordenador y la tarjeta de adquisición de datos (DAQ) el control de un proceso real, con la ventaja de tener todos estos implementos necesarios en un mismo lugar.
- Se desarrollo con éxito un modelo de aplicaciones sobre “tanques interactuantes”, donde se puede operar sobre un sistema real para simular en menor escala diferentes procesos de llenado, así como también cambios de temperatura a través de ecuaciones y ejemplo con potenciómetro que podrían ser realizados en la industria, por otro lado, la aplicación también se diseño como un sistema abierto para permitir simular procesos similares, dependiendo de la creatividad y necesidad del usuario.
- No se logro con éxito el registro de temperatura en el banco de control debido a las características de la tarjeta y su baja resolución, a pesar de haber amplificado la señal fue bastante inestable y errática.



- Se realizó con éxito la identificación de los componentes del banco para el fácil discernimiento de los elementos existentes así como también su estado actual, además de contar con etiquetas que reseñan los mismos, permitiendo familiarización tanto del estudiante como del docente para su uso efectivo.

- Se logró crear el tablero eléctrico el cual está debidamente probado y operativo, con sus conexiones y cableado debidamente organizado con el fin de hacer comprensible y sencillo al estudiante el uso del banco, además del enfoque práctico y de programación que este brinda.

- Se desarrolló un cronograma de actividades del laboratorio de Control, así como también prácticas de laboratorios que permiten a los estudiantes aplicar los conocimientos teóricos adquiridos en clase, haciendo uso del equipo, también se creó un diseño virtual del banco de proceso en 3D, con la finalidad de hacer más ilustrativas las prácticas y detallar lo que se quiere durante su ejecución.

- Se desarrolló diversas prácticas (ver capítulo V), en las cuales el resultado fue satisfactorio, en vista de que el sistema cumplió correctamente con todas las órdenes programadas, verificándose así una vez más, el buen funcionamiento del Banco de Control de Temperatura, sin embargo la creación de ecuaciones en la práctica “5.3. Práctica de Laboratorio N° 3: Proceso de vaciado y llenado de tanques mediante variación de temperatura por medio de ecuaciones”, permitió solventar la carencia de un dispositivo capaz de registrar temperatura en el proceso.



RECOMENDACIONES

En primer lugar, se recomienda ampliamente la utilización del Banco de Control de Procesos en el Laboratorio de Control, a través de las diferentes prácticas desarrolladas para el banco en el presente Trabajo Especial de Grado.

Antes de cada práctica se recomienda leer las instrucciones del banco, a fin de hacer un buen uso del equipo y evitar el deterioro de componentes como la Tarjeta de Adquisición de Datos, los sensores, la válvula solenoide y bomba, además de con ello prevenir accidentes lamentables.

Adicionalmente, se pueden diseñar diferentes prácticas para el banco, ya que la programación es abierta y permite hacer cambios a cada paso que el sistema valla a actuar como temporizar los llenados y otros. Especialmente se pueden desarrollar prácticas para complementar el Banco de Control de Procesos con otros bancos que existan en el laboratorio de automatización II.

También es posible la conectividad entre LabVIEW® y una gran gama de PLCs soportados por NI OPC Server. Entre los más usados tenemos los modelos S7-200, S7-300, S7-400 entre otros las cuales contamos con bancos de procesos que tienen integrados este tipo de autómatas programables en el laboratorio de automatización II. Para una mejor idea de cómo se programa esto se recomienda visitar el siguiente enlace web y revisar el tutorial que aparece en bibliografía [10], o apéndice e.

Como actividad complementaria se puede mostrar a los estudiantes el cableado del sistema que se encuentra en la parte trasera del banco, permitiendo que observen el tipo de conexiones e identifiquen conductores por medio de nomenclatura asignada, los diferentes valores de tensiones que existen, ya que con ellos se da a conocer ciertas normas y reglas



del cableado de control y además se incentiva la realización de futuros trabajos de este tipo como la creación de la tarjeta de conexiones.

Se puede probar implementar la instalación de otras Tarjetas de Adquisición de datos en el banco de control, lo cual es posible siempre que estos posean algunas características similares como la cantidad de entradas y salidas digitales, analógicas, temporizadores y otros, siempre y cuando su alimentación sea vía USB. En el caso de tarjetas de Adquisición de datos que se implementen posean con otros niveles de tensión se recomienda usar fuentes de poder acordes a la misma y protecciones a fin de evitar daños en algún dispositivo o la tarjeta de adquisición, se recomienda al igual que en este trabajo el uso de relés a cada salida para proteger los componentes.

Debido a las restricciones y carencias de la tarjeta usada para este trabajo, se recomienda también hacer el uso de la Tarjeta de Adquisición de Datos NI USB 9211A que permite leer señales de termocupla gracias a su resolución y que está entre sus características la adquisición de este tipo de señales, con esta tarjeta podrá tomar las señales de temperatura en tiempo real.

Debido a la utilidad que posee este tipo de herramientas interactivas para el aprendizaje en el área de Control, se deja como propuesta final desarrollar nuevos “bancos” donde se implementen diferentes aplicaciones de la industria en menor escala, así como también el uso de dispositivos de medición analógicos, lograron que el banco de control sea versátil, conservando las características de sencillez, eficiencia y portabilidad que posee el Banco de Control de Procesos desarrollado en el presente Trabajo Especial de Grado.



BIBLIOGRAFÍA

[1] Hidalgo, Katherine (2008).

“Desarrollo de un banco de procesos didáctico para el Laboratorio de Automatización Industrial I de la Escuela de Ingeniería Eléctrica de la Universidad de Carabobo”

[2] Johan W. Janampa P. (2008)

“Desarrollo de un Banco de Control de Procesos Portátil para el Laboratorio de Automatización Industrial I mediante un Micro PLC Simatic S7-200”

[3] Rodrigo .González C, Rodrigo P. Pino

<http://cybertesis.uach.cl/tesis/uach/2007/bmfci643a/doc/bmfci643a.pdf>

[4] Dra. Valentina, Ulver de Beluatti, Ing. Germán Pineda, Dr. Rodolfo Ávila, Dr. Hugo Juri “Aplicación de LabVIEW® como instrumento virtual en medición de la movilidad dental”

http://www.informedicajournal.org/a2n3/tc_2/beluatti.pdf

[5] Ing. Maria Paula Gómez Quintero, Ing. Gustavo Andrés Zabala Dominguez, Ing. Javier Andrés Dávila Rincón

<http://www.dyna.unalmed.edu.co/ediciones/169/articulos/a17v78n169/a17v78n169.pdf>

[6] Manual de prácticas de control avanzado utilizando la tarjeta de adquisición de datos NI USB 6008.

<http://es.scribd.com/doc/35553976/Manual-de-Practicas-de-Control-Avanzado-VERSION-Est>

[7] Software grafico. “LabVIEW^R”

<http://www.ni.com/labview/esa/>



[8] **Adquisición de Datos.**

<http://www.ni.com/dataacquisition/esa/>

[9] Seminarios Web y Videos sobre software grafico y tarjeta de adquisición de datos.

<http://www.ni.com/webcasts/esa/>

[10] Tutorial sobre configuración de OPC Server para comunicar LabVIEW con PLCs.

(<http://forums.ni.com/t5/Discusiones-sobre-Productos-NI/Conectar-LabView-con-PLC-mediante-NI-OPC-Server/m-p/2007119>)