



UNIVERSIDAD DE QUINTANA ROO
DIVISIÓN DE CIENCIAS E INGENIERÍA

INSTRUMENTACIÓN VIRTUAL DE UN ELECTROLIZADOR DE MEMBRANA DE INTERCAMBIO PROTÓNICO

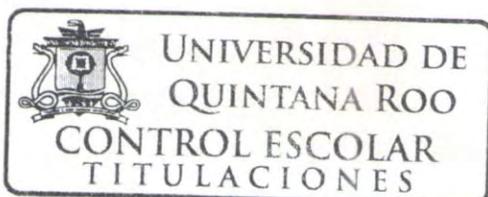
TESIS
PARA OBTENER EL GRADO DE

INGENIERO EN REDES

PRESENTA
ANDRÉS OMAR KIM BATÚN

DIRECTOR DE TESIS
DR. VÍCTOR MANUEL SÁNCHEZ HUERTA

ASESORES
DR. GLISERIO ROMELI BARBOSA POOL
DR. FREDDY IGNACIO CHAN PUC
DR. JAVIER VÁZQUEZ CASTILLO
DR. HOMERO TORAL CRUZ



CHETUMAL QUINTANA ROO, MÉXICO, AGOSTO DE 2018



UNIVERSIDAD DE QUINTANA ROO
DIVISIÓN DE CIENCIAS E INGENIERÍA

TRABAJO DE TESIS TITULADO
“INSTRUMENTACIÓN VIRTUAL DE UN ELECTROLIZADOR DE MEMBRANA DE INTERCAMBIO
PROTÓNICO”

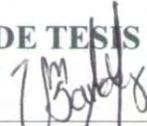
ELABORADO POR
ANDRÉS OMAR KIM BATÚN

BAJO SUPERVISIÓN DEL COMITÉ DE ASESORÍA Y APROBADO COMO REQUISITO PARCIAL
PARA OBTENER EL GRADO DE:

INGENIERO EN REDES

COMITÉ DE TESIS

DIRECTOR:



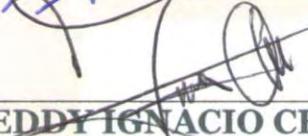
DR. VÍCTOR MANUEL SÁNCHEZ HUERTA

ASESOR:



DR. GLISERIO ROMELI BARBOSA POOL

ASESOR:



DR. FREDDY IGNACIO CHAN PUC

ASESOR:

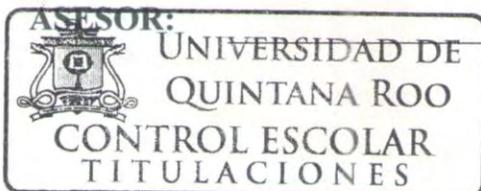


DR. JAVIER VAZQUEZ CASTILLO

ASESOR:



DR. HOMERO TORAL CRUZ



Agradecimientos

Agradezco a mi familia, maestros y amigos por el apoyo recibido para poder concluir con mis estudios.

Resumen

En este trabajo de tesis se desarrolló un sistema de adquisición de datos remoto mediante un sistema embebido basado en un controlador en tiempo real y una FPGA (Field Gate Programmable Array por sus siglas en inglés). El sistema de adquisición de datos está basado en la tecnología de la instrumentación virtual y el cual fue desarrollado mediante el sistema embebido NI-myRIO y programado mediante el software LabVIEW. El instrumento virtual desarrollado permite visualizar y almacenar los datos adquiridos en tiempo real. Esto significa que se puede adquirir un conjunto de n variables al mismo tiempo sin que haya un tiempo de retraso entre los mismos.

La técnica de la instrumentación virtual consiste en la programación de un instrumento virtual de medición en un sistema computacional. La ventaja que tiene el desarrollo de un instrumento virtual es que permite desarrollar instrumentos de medición acorde a las necesidades que requiera el usuario.

El instrumento virtual que se desarrolló en este trabajo mide el voltaje y corriente eléctrica de CD (Corriente Directa) aplicadas a la celda electrolizadora PEM y calcula la potencia instantánea suministrada al electrolizador. El instrumento virtual funciona mediante comunicación WiFi configurado en la tarjeta myRIO en el cual se pueden visualizar las gráficas y medidores, los cuales indican los datos que se están adquiriendo en tiempo real desde la tarjeta en una página web generada por el software LabVIEW. De igual forma, el instrumento virtual permite configurar algunos parámetros como el tiempo de muestreo de la medición de las variables, cambiar el nombre del archivo con el que se guardan los datos adquiridos desde la misma página web, al igual que indica el espacio de almacenamiento del dispositivo en el que se está guardando el archivo

Los datos registrados por el instrumento virtual se almacenan en una memoria USB. Los archivos creados con los datos registrados se pueden acceder vía remota al servidor instalado en la tarjeta mediante el protocolo FTP (File Transfer Protocolo por sus siglas en inglés). Mediante el servidor FTP se puede descargar el archivo

de texto con los datos de tiempo de muestreo, voltaje, corriente y potencia eléctrica que se generó en el instrumento virtual mediante el registro de los valores de estas variables mediante la tarjeta myRIO.

Tabla de contenido

Agradecimientos.....	3
Resumen	4
Tabla de Figuras.....	8
Capítulo 1 INTRODUCCION.....	10
Objetivo General	11
Objetivos Específicos	11
Problemática.....	12
Justificación	13
Capítulo 2 Electrolizador tipo PEM	14
Tipos de electrolizadores.....	14
2.1 Electrolizador de membrana de intercambio protónico (PEM).....	14
2.2. electrolizador alcalino	14
2.3 Electrolizador de óxido sólido.....	15
2.4 Partes de un electrolizador tipo PEM.	16
Electrodos.....	16
Electrolito	17
Membrana.....	17
Contenedor del electrolito.....	17
2.5 ¿Qué es un electrolizador tipo PEM?	17
2.6 La instrumentación Virtual.....	18
Sistema de adquisición de datos.....	19
Componentes para la instrumentación virtual	19
Función de la PC en un sistema DAQ	23
Software controlador	23
Software de aplicación	23
Capítulo 3 Sistema embebido NI myRIO-1900 y software LabVIEW	24
3.1 Sistema embebido NI myRIO	24
3.2 ¿Qué es un sistema embebido?	25
3.2.1 Arquitectura de un sistema embebido.	26
3.3 ¿Qué es un sistema operativo de tiempo real (RTOS)?.....	28
3.4 ¿Qué es un FPGA?.....	28
3.5 Arquitectura de una FPGA.....	29

3.6 Descripción general del hardware	31
3.6.1 Grupo de puertos A B y C de la tarjeta myRIO-1900.....	31
3.6.2 CANALES DE ENTRADA ANALOGICA.....	34
3.6.3 Canales de salida analógica.....	35
3.6.4 Conversión de valores de datos sin procesar a voltaje para los puertos MXP.	36
3.7 software LabVIEW	37
3.8 Instrumento Virtuales (VIs)	38
3.8.1 Panel frontal.....	39
3.8.2. Paleta de controles.....	39
3.8.3 Diagrama de Bloques	43
Capítulo 4 Diseño del instrumento virtual.	45
4.1 configuración de la tarjeta myRIO-1900 como AP WiFi	45
4.2 Instalación del servidor FPT en la tarjeta NI myRIO-1900	51
4.3 programación del instrumento virtual a nivel FPGA de la tarjeta myRIO-1900	55
4.3.1 Panel frontal del programa a nivel FPGA	59
4.4 Desarrollo del instrumento virtual a nivel procesador de la tarjeta myRIO-1900	60
4.4.1 Programación para visualización de datos.....	60
4.4.2 Programación para creación de archivos y almacenamiento de datos	63
4.4.3 panel frontal del programa a nivel procesador.....	66
4.5 Simulación de señales a medir.	67
Capítulo 5 Resultados experimentales.	68
5.1 Medición de voltaje	69
5.2 Medición de Corriente	71
5.3 Medición de potencia eléctrica	73
Conclusiones	75
Trabajos Futuros	76
Bibliografía	77
Anexos	80
creación de un proyecto nuevo en LabVIEW	80
Especificaciones de la tarjeta NI myRIO-1900.....	86

Tabla de Figuras.

Figura 2.1 sistema de adquisición de datos	19
Figura 2.2 Proceso de cuantificación	22
Figura 3.1 Sistema embebido NI myRIO-1900.....	24
Figura 3.2 Arquitectura de un sistema embebido	26
Figura 3.3 sistema de procesamiento de un FPGA	29
Figura 3.4 Arquitectura de un FPGA	29
Figura 3.5 Diagrama de bloques componentes de la tarjeta myRIO-1900.....	31
Figura 3.6 Representación de pines de los puertos A y B de la tarjeta myRIO1900.....	32
Figura 3.7 Representación de pines en el puerto C de la tarjeta myRIO-1900.....	33
Figura 3.8 Topología de las entradas analógicas de la myRIO-1900.....	34
Figura 3.9 Topología de las salidas a analógicas de la myRIO-1900	35
Figura 3.10 Pantalla de inicio de LabVIEW para dispositivos myRIO	37
Figura 3.11 proceso para el diseño de un instrumento Virtual	38
Figura 3.12 (1) Ventana del Panel Frontal, (2) Barra de Herramientas, (3) paleta de Controles.....	39
Figura 3.13 Acceso a la paleta de controles.....	40
Figura 3.14 Paleta de controles	40
Figura 3.15 Diseño de un VI que contiene indicadores y controles en su interfaz de usuario	41
Figura 3.16 (1) Incremento/Decremento del valor numérico, (2) Control, (3) Indicador.....	42
Figura 3.17 Indicador y control Booleano.....	42
Figura 3.18 indicadores y controles de cadena de caracteres.....	43
Figura 3.19 (1) Indicadores (2) Cables (3) Nodos (4) Controles.	43
Figura 3.20 Código fuente gráfico.....	44
Figura 4.1 ventana de conexión de la tarjeta myRIO-1900 a la PC.....	46
Figura 4.2 Vinculación de la tarjeta myRIO a NI MAX.....	46
Figura 4.3 selección del tipo de dispositivo para agregar a NI MAX.....	47
Figura 4.4 Vinculación exitosa de la tarjeta myRIO a NI MAX	48
Figura 4.5 Tarjeta agregada a la aplicación NI MAX	48
Figura 4.6 Configuración y recursos de la tarjeta NI myRIO-1900.....	49
Figura 4.7 Configuración predeterminada de los adaptadores de red de la tarjeta NI myRIO-1900.....	49
Figura 4.8 configuración para utilizar la tarjeta NI myRIO como punto de acceso	50
Figura 4.9 conexión WiFi establecida entre la tarjeta NI myRIO-1900 y la PC	50
Figura 4.10 Primer paso para instalar el servidor FTP	51
Figura 4.11 Ventana de inicio de sesión como permiso para instalar el servidor FTP.	51
Figura 4.12 segundo paso para instalar el servidor FTP en la tarjeta.....	52
Figura 4.13 opciones que se muestran al desplegar el apartado Software Add-ons	52
Figura 4.14 tercer paso para instalar el servidor FTP en la tarjeta	53
Figura 4.15 aplicaciones seleccionadas para instalar en la tarjeta myRIO	54
Figura 4.16 conexión establecida al servidor FTP instalado en la tarjeta myRIO-1900.....	54
Figura 4.17 Ciclo While Loop y entradas analógicas en el diagrama de bloques	55
Figura 4.18 Indicadores numéricos tipo Entero y FXP	56
Figura 4.19 conexión del nodo conversor FXP e indicadores numéricos enteros de 16bits	56
Figura 4.20 configuración del nodo conversor y el indicador FXP numérico para representación de datos de 12 bits.....	56

Figura 4.21 Ubicación y configuración del Express VI (Mean, Variance, and Standard Deviation)..	57
Figura 4.22 conexión del Express VI Express VI (Mean, Variance, and Standard Deviation) al nodo convertidor FXP	57
Figura 4.23 multiplicaciones para obtener el valor de corriente y voltaje en números decimales..	58
Figura 4.24 Diseño final del diagrama de bloques.....	59
Figura 4.25 Diseño final del panel Frontal	59
Figura 4.26 ciclo Time Loop y Nodos para abrir, leer y cerrar un VI referenciado de la FPGA.....	60
Figura 4.27 configuración del nodo Open FPGA VI Reference.	61
Figura 4.28 conexión de los nodos para abrir, leer y cerrar el VI referenciado.....	61
Figura 4.29 conexión de los nodos conversores a las salidas del bloque Read/Write Control	62
Figura 4.30 Indicadores numéricos y graficas agregadas desde el panel frontal	62
Figura 4.31 Algoritmo para la Visualización de datos enviados por la FPGA.....	63
Figura 4.32 primera parte de componentes usados para el algoritmo de creación de archivos.	64
Figura 4.33 Primera parte del algoritmo para la creación de archivos en el Diagrama de Bloques.	64
Figura 4.34 Segunda parte del algoritmo para la creación de archivos.....	65
Figura 4.35 Diseño final del diagrama de bloques.....	66
Figura 4.36 Diseño final del panel frontal.....	66
Figura 4.37 simulación para la adquisición de datos de las señales de corriente y voltaje, y resultados adquiridos.....	67
Figura 5.1 gráfica de voltaje para el punto de control de 2 bares asignado al electrolizador.....	69
Figura 5.2 gráfica de voltaje para el punto de control de 4 bares asignado al electrolizador.....	70
Figura 5.3 gráfica de voltaje para el punto de control de 2 bares asignado al electrolizador.....	70
Figura 5.4 gráfica de corriente para el punto de control de 2 bares asignado al electrolizador.....	71
Figura 5.5 gráfica de corriente para el punto de control de 4 bares asignado al electrolizador.....	72
Figura 5.6 gráfica de corriente para el punto de control de 6 bares asignado al electrolizador.....	72
Figura 5.7 gráfica de potencia para el punto de control de 2 bares asignado al electrolizador.	73
Figura 5.8 gráfica de potencia para el punto de control de 4 bares asignado al electrolizador.	74
Figura 5.9 gráfica de potencia para el punto de control de 6 bares asignado al electrolizador.	74
Figura de anexo 1 Ventana de inicio de LabVIEW	80
Figura de anexo 2 Configuración y opciones requeridas para la creación de un proyecto en LabVIEW	81
Figura de anexo 3 Componentes del proyecto creado.....	82
Figura de anexo 4 Creación de un VI en la FPGA de la tarjeta myRIO-1900	82
Figura de anexo 5 cambio de nombre a las entradas analógicas AIO y AI1 de la tarjeta NI myRIO-1900.....	83
Figura de anexo 6 Entradas analógicas AIO y AI1 renombradas en el proyecto.....	83
Figura de anexo 7 conexión entre la tarjeta myRIO y el Software LabVIEW	84

Capítulo 1 INTRODUCCION

En la División de Ciencias e Ingeniería (DCI) de la Universidad de Quintana Roo (UQROO), una de sus principales áreas de investigación son las energías renovables. Para esta institución, las energías renovables son consideradas una opción viable para el estudio e investigación. En este sentido, el grupo de tecnologías del hidrógeno de la universidad cuenta con un electrolizador tipo PEM de 100 W para efectos de investigación. Una de las necesidades técnicas requeridas en este grupo de trabajo es la caracterización de la celda PEM del electrolizador para obtener su respuesta dinámica de voltaje, corriente y potencia eléctrica aplicada.

Este trabajo está centrado en el desarrollo en la instrumentación virtual de un electrolizador tipo PEM (Proton Exchange Membrane, por sus siglas en inglés). El sistema de instrumentación virtual consiste en el monitoreo en tiempo real de las variables eléctricas de corriente y voltaje consumido por la celda del electrolizador y el cual permite la visualización y el almacenamiento de datos en una memoria USB. Los datos almacenados se podrán descargar en archivos con formato CSV vía WEB desde un servidor FTP instalado en el sistema operativo de la tarjeta myRIO-1900.

El sistema está diseñado para la adquisición de señales eléctricas a nivel laboratorio para prototipos de tecnologías de hidrógeno como son los electrolizadores y celdas de combustible. El propósito de este proyecto es el desarrollo de una aplicación de bajo costo y alta flexibilidad en cuanto a programación para la adquisición de señales eléctricas.

Este proyecto se concreta en la programación de un sistema embebido NI myRIO-1900 con una FPGA integrada, en donde la FPGA es la encargada de procesar y acondicionar las señales adquiridas para poder tener una visualización de los datos adquiridos mediante gráficas, al igual se podrá hacer el almacenamiento en de los datos capturados en tiempo real en una memoria USB y poderlos descargar desde un servidor FTP. El instrumento virtual desarrollado se aplicará a un

electrolizador tipo PEM. Para la adquisición de los valores de corriente eléctrica a medir se instalará un sensor de corriente dentro del electrolizador con características específicas para mediciones con la tarjeta NI myRIO-1900, esto con el objetivo de caracterizar la celda tipo PEM para obtener el consumo de corriente eléctrica. Por otra parte, para la medición del voltaje aplicado al electrolizador no se necesita un circuito acondicionador de señal para su medición ya que los valores de voltaje aplicados sobre la celda electrolizadora se encuentran en el rango de medición del sistema embebido de la tarjeta myRIO.

Este trabajo de tesis se encuentra dividido en cinco capítulos. En el primer capítulo se presenta la introducción, donde se aborda la problemática y se describe la importancia del estudio y la investigación de las energías renovables en la Universidad de Quintana Roo. De igual forma se describe el diseño del sistema de adquisición de datos para la caracterización de celda electrolizadora tipo PEM. El segundo capítulo está enfocado al funcionamiento del electrolizador tipo PEM y la descripción de los componentes de un sistema de adquisición de datos. En el tercer capítulo se presenta la descripción del hardware NI myRIO-1900 y el lenguaje de programación usado por el software LabVIEW. En el capítulo cuatro se muestra el desarrollo del instrumento virtual, el cual consiste en la programación de sistema embebido. En el quinto capítulo se muestran los resultados experimentales obtenidos y finalmente se presentan las conclusiones obtenidas con el desarrollo de este trabajo de tesis.

Objetivo General

Desarrollar un instrumento virtual para un electrolizador tipo PEM de 100 W.

Objetivos Específicos

- Diseñar y construir circuitos de medición de tensión y corriente eléctrica con aislamiento eléctrico.
- Diseñar y programar un instrumento virtual en un sistema computacional reducido (sistema embebido) basado en un microcontrolador ARM y una FPGA.

- Generar conocimiento y experiencias técnicas en el desarrollo de instrumentos virtuales sobre sistemas embebidos.
- Obtener una base de datos para la caracterización del electrolizador tipo PEM de 100 W

Problemática

El hidrógeno es una opción viable como un medio para el almacenamiento de energía en sistemas de generación eléctrica con energías renovables. La generación de hidrógeno se realiza mediante un electrolizador. Existen diferentes tipos de electrolizadores para la generación de hidrógeno. Uno de estos tipos de electrolizador es el de membrana de intercambio protónico (PEM, por sus siglas en inglés Proton Exchange Membrane). Las principales ventajas de un electrolizador tipo PEM consisten en que genera hidrógeno con alto grado de pureza ($\approx 99.9\%$) y tienen una temperatura de operación baja ($\approx 55\text{ }^{\circ}\text{C}$).

El grupo de tecnologías del hidrógeno de la Universidad de Quintana Roo cuenta con un electrolizador tipo PEM de 100 W. Para efectos de investigación se requiere de la caracterización de la celda PEM del electrolizador para obtener su respuesta dinámica. Para ello se requiere medir el voltaje, corriente y potencia eléctrica aplicada a una de las celdas tipo PEM que integran el electrolizador.

En este trabajo de tesis se propone el desarrollo de un sistema de adquisición de datos remoto basado en la técnica de la instrumentación virtual mediante el software LabVIEW y un sistema embebido basado en un controlador en tiempo real y de una FPGA (Field Gate Programmable Array por sus siglas en inglés). Este sistema permitirá poder visualizar y almacenar los datos adquiridos en tiempo real. Esto quiere decir que podrá adquirir cualquier conjunto de n variables al mismo tiempo sin que haya un tiempo de retraso en cada variable.

La técnica de la instrumentación virtual consiste en la programación de las funciones de un instrumento de medición en un sistema computacional. La ventaja de la

instrumentación virtual es que permite desarrollar instrumentos de medición acorde a las necesidades del usuario.

El instrumento virtual por desarrollar en esta tesis medirá el voltaje y corriente eléctrica de CD aplicadas a la celda electrolizadora PEM y calculará la potencia instantánea suministrada a una de las celdas tipo PEM. El instrumento virtual se podrá operar vía remota mediante WiFi y se podrá configurar el tiempo de muestreo de la medición de las variables, generará el archivo de texto con los datos medidos y mediante un servidor FTP (File Transfer Protocol por sus siglas en inglés) acceder a estos archivos vía WEB.

Justificación

En este trabajo de tesis se propone desarrollar una aplicación de bajo costo y alta flexibilidad de programación que permita la adquisición de variables eléctricas de voltaje, corriente y potencia. El usuario podrá controlar el rango de muestreo al que requiera los datos. De esta forma se podrá disponer de una mayor versatilidad a los datos almacenados y permitir al usuario obtener una base de datos con archivos de los datos de las señales capturadas, para la manipulación de los datos y poder realizar trabajos futuros.

Capítulo 2 Electrolizador tipo PEM

Tipos de electrolizadores

Existen diferentes tipos de electrolizadores para la generación de hidrogeno. De esta forma, en las siguientes subsecciones se presentan las características de las principales tecnologías usadas en electrolizadores.

2.1 Electrolizador de membrana de intercambio protónico (PEM).

Este tipo de electrolizador realiza la separación de la molécula de agua en sus dos elementos principales, hidrógeno y oxígeno al aplicársele voltaje y corriente de CD en sus electrodos. Los electrolizadores tipo PEM (Proton Exchange Membrane) utilizan una membrana polimérica, de plástico Nafion®, universalmente usado como electrolito base. La temperatura de operación que alcanza el proceso de generación hidrógeno es de entre 80 y 100 °C con una pureza del gas generado de un 99.9999%. Los electrolizadores tipo PEM se consideran una tecnología prometedora en la producción de hidrógeno a largo plazo, para aplicaciones a pequeña o media escala, tales como alimentación de coches o aplicaciones más pequeñas. Una ventaja adicional de este tipo de electrolizadores es que pueden ser aprovechado para producir Hidrógeno utilizando como fuente de energía, las energías renovables como la solar, eólica. Etc. Los electrolizadores de Membrana intercambio Protónico se han consolidado como una tecnología aceptable industrialmente y se distinguen por ser compactos y limpios, en comparación con otros tipos de electrolizadores, como los alcalinos estos muestran operaciones de funcionamiento a voltajes bajos, pero por el contrario un alto consumo de corriente; así como los parámetros de temperatura y presión.

2.2. electrolizador alcalino

Los electrolizadores alcalinos de agua son similares a los PEM, pero como electrolito usan una solución alcalina, sea hidróxido de sodio o hidróxido de potasio. El proceso de generación de hidrógeno es el mismo que en un electrolizador tipo PEM, pero a una temperatura de operación más alta (entre 100 y 150 °C), y las reacciones en los electrodos son distintas. Actualmente, esta tecnología de

electrolizadores es el estándar actual para la electrólisis a gran escala de hidrógeno, ya que es la más robusta en comparación a los otros tipos de producción de hidrógeno mediante electrólisis. Los electrolizadores del tipo alcalino emplean un electrolito líquido principalmente de sodio (NaOH) o hidróxido de potasio (KOH). Este tipo de electrolito es apropiado porque se caracteriza su alta conductividad iónica.

2.3 Electrolizador de óxido sólido

Los electrolizadores de óxido sólido utilizan un material cerámico sólido como electrolito, el cual transmite los iones de oxígeno cargados negativamente a elevadas temperaturas (entre 500 y 800 °C) para disminuir la cantidad de energía eléctrica requerida. El proceso de generación de hidrógeno es diferente a los electrolizadores tipo PEM y alcalinos. En los electrolizadores de óxido sólido el agua reacciona liberando hidrógeno, así como iones de oxígeno cargados negativamente en el cátodo, y en el ánodo es donde se forma el oxígeno. La reacción de descomposición del agua para producir hidrógeno y oxígeno en forma de gas se produce de manera no espontánea similar a los electrolizadores PEM y alcalinos. Los electrolitos cerámicos utilizados en estos electrolizadores generan oxígeno a temperaturas de 750-1000 °C y el hidrógeno generado se produce con una temperatura de 450-750 °C. Con esta tecnología se puede reducir el gasto en el consumo de energía eléctrica, además de contar con los beneficios de un electrolito sólido; sin embargo, es necesario una fuente de calor y materiales capaces de trabajar altas temperaturas.

Tabla 1 Ventajas y desventajas de los electrolizadores alcalinos y tipo PEM

Electrolizador tipo PEM	
Ventajas	Desventajas
peso y tamaño son compacto	Tecnología a escalas (pruebas)
Alta pureza del gas	Tiempo de vida corto

Operación dinámica	Precios elevados
Altas eficiencias	Consumo de corriente elevado
Rápida respuesta del sistema	Coste elevado
Buen rendimiento a carga parcial	Producción de hidrógeno a pequeña escala
El mantenimiento fácil y rápido	
Baja temperatura	
Electrolizador Alcalino	
Baja consumo de corriente	Mantenimiento por personal capitado
Mayor tiempo de vida	Alta temperatura
Económico	Fuga de gases (grado de pureza)
Reducción de gastos	Electrolito corrosivo
Producción de hidrógeno a gran escala	Poca capacidad en dinámica
Bajo consumo de corriente	Presión de operación baja
Tecnología bien consolidada	

2.4 Partes de un electrolizador tipo PEM.

Un electrolizador está construido a partir de los siguientes elementos.

Electrodos

Los electrodos son placas elaboradas de un material resistente a la corrosión, que realiza el fenómeno de electrolisis, al entrar en contacto con una corriente eléctrica. Para llevar a cabo la electrólisis se debe considerar que existen dos electrodos: uno con carga negativa denominado cátodo y otro con carga positiva denominado ánodo y separados por un electrolito. En la parte separada entre el electrolito y el ánodo se produce una reacción que forma oxígeno gaseoso y en la separación del cátodo

con el electrolito se produce una segunda reacción generando hidrógeno gaseoso. Hay algunos requisitos básicos que los electrodos deben cumplir para efectuar la electrólisis correctamente son:

- Resistentes a la corrosión.
- Buena conductividad.
- Buenas propiedades catalíticas.
- Integridad estructural.

Electrolito

Los electrolitos consisten en una sustancia que contienen iones libres, los cuales se comportan como un conductor eléctrico. Los electrólitos también son conocidos como soluciones iónicas, pueden ser electrólitos en solución acida o electrólitos sólidos.

Membrana

En los electrolizadores de tipo PEM, podemos encontrar una membrana que hace la función de un electrolito en dos cavidades, una en contacto con el ánodo y otra en contacto con el cátodo, para separar los gases producidos en la electrólisis.

Contenedor del electrolito.

El contenedor del electrolito, como lo indica su nombre, es el elemento del electrolizador que contiene al electrolito. Las características necesarias que debe cumplir un contenedor para el correcto funcionamiento dentro del electrolizador y así poder efectuar correctamente el fenómeno de la electrolisis son la resistencia a altas temperaturas y tener presión determinada para evitar fugas de gas.

2.5 ¿Qué es un electrolizador tipo PEM?

Un electrolizado tipo PEM, es un dispositivo electroquímico encargado de la producción de hidrogeno mediante la descomposición del agua en el proceso de electrólisis. Este proceso se hace de manera no espontanea, esto quiere decir, que requiere suministro de energía externa para la descomposición de la molécula del

agua en moléculas de hidrógeno y oxígeno. La energía suministrada es obtenida por el paso de corriente eléctrica a través de dos electrodos (ánodo y cátodo). El ánodo y cátodo están divididos por un electrolito sólido capaz de conducir iones a través de él, en este caso, una membrana de intercambio protónico (PEM, por sus siglas en inglés, Proton Exchange Membrane). En el electrodo negativo o cátodo se produce el hidrógeno y en el electrodo positivo o ánodo se libera oxígeno. Gracias al uso de membranas de intercambio protónico, los diseños de electrolizadores tipo PEM son de tamaño más compacto y su respuesta de operación es más rápida en comparación a los electrolizadores alcalinos. Para la producción de gases de alta pureza en estos tipos de electrolizadores el reactivo que se debe usar es agua destilada.

2.6 La instrumentación Virtual.

La instrumentación virtual es un concepto introducido por la compañía National Instrument. En el año 1983, Truchard y Kodosky [1] , de National Instrument, decidieron enfrentar el problema de crea un software que permitiera utilizar la computadora PC como un instrumento para realizar mediciones. El concepto de instrumentación virtual nace a partir de uso de una computadora personal (PC) como un “instrumento” de medición de señales eléctricas proporcionadas por un sensor, como puede ser la temperatura, la presión, etc. Esto quiere decir que la PC puede ser usada como herramienta de medición de fenómenos físicos representados mediante señales eléctricas, por ejemplo: corriente (4-20mA), voltaje (0-5V).

La instrumentación virtual ha surgido como una herramienta importante para la adquisición de datos y el monitoreo remoto proporcionando grandes ventajas sobre la instrumentación tradicional. La instrumentación virtual no solo son simples mediciones de variables eléctricas, sino que vas más haya de ello por lo que también está involucrado el procesamiento de señales, análisis, almacenamiento de datos y la información de medición de las señales eléctricas. Esto significa que la instrumentación virtual no solo está conformada por la adquisición de señales, sino que esta involucra la interacción entre hombre-máquina, tratamiento y

procesamiento de señales, almacenamiento de datos y comunicación entre dos o más equipos. En el siguiente capítulo se hablará sobre el concepto de la instrumentación virtual a fondo.

Sistema de adquisición de datos

Por su parte la adquisición de datos (DAQ, Data Acquisition, por sus siglas en inglés) es el proceso de medir con una computadora una señal eléctrica física como voltaje, corriente, temperatura, presión, sonido etc. Un sistema DAQ está formado por sensores, circuitos electrónicos acondicionadores de señal y software para programar el instrumento virtual. En la Figura 2.1 se muestra el esquema de un sistema de adquisición de datos basado en PC.



Figura 2.1 sistema de adquisición de datos

Componentes para la instrumentación virtual

Para la implementación de la instrumentación virtual se requiere de una tarjeta de adquisición de datos que contienen circuitos electrónicos acondicionadores de señales, software y una PC con características suficientes para poder ejecutar el software de la tarjeta de adquisición de datos. Para poder realizar la adquisición de datos y obtener buenos resultados en la medición se pueden considerar los siguientes elementos los cuales son esenciales para poder implementar la instrumentación virtual.

- Sensor o Transductor
- Hardware de acondicionamiento de señales
- PC y Software

Sensores (Transductores)

Los sensores, también llamado transductores, convierten una señal física en una señal eléctrica que se puede medir. Dependiendo del tipo de sensor, la señal eléctrica adquirida por el sensor puede ser una de voltaje, corriente, resistencia u otro variable eléctrica que cambia con el tiempo. Algunos sensores pueden requerir circuitos acondicionadores para producir correctamente una señal que puede ser leída con precisión y con toda seguridad por el hardware de adquisición de datos. En la **Tabla 2** se muestran algunos tipos de sensores y la señal de medición que producen.

Tabla 2 sensores y su señal de medición

Sensor	Señal de Medición
Termopar, RTD, Termistor	Temperatura
Fotosensor	Luz
Micrófono	Sonido
Galga Extensiométrica, Transductor Piezoeléctrico	Fuerza y Presión
Potenciómetro, LVDT, Codificador Óptico	Posición y Desplazamiento
Acelerómetro	Aceleración
Electrodo pH	pH

Hardware de adquisición de datos

El hardware de adquisición de datos actúa como la interfaz entre una PC y las señales analógicas adquiridas por los sensores. Funciona principalmente como un dispositivo que digitaliza señales analógicas entrantes para que una PC pueda interpretarlas. Los tres componentes clave de un dispositivo DAQ usado para medir una señal son: el circuito de acondicionamiento de señales, convertidor analógico-digital (ADC) y un bus de comunicación en la computadora.

Circuitos Acondicionadores de señales.

La señal de salida del sensor de un sistema de medición en general se debe procesar de una forma adecuada para la siguiente etapa de la operación. La señal puede ser, por ejemplo, demasiado pequeña, y sería necesario amplificarla; podría contener interferencias que eliminar; ser no lineal y requerir su linealización; ser analógica y requerir su digitalización; ser digital y convertirla en analógica, etc. a todos estos procesos que se llevan a cabo se le conoce como acondicionamiento de señales. Por ejemplo, las señales de un termopar proporcionan unos niveles muy pequeños de tensión que deben ser amplificados antes de la digitalización. Otros sensores, tales como detectores de temperatura mediante resistencias (por sus siglas en inglés RTD), termistores, y acelerómetros, requieren que se les aplique una fuente de voltaje o corriente para poder funcionar.

convertidor analógico-digital

Las señales analógicas de los sensores deben ser convertidas en digitales antes de ser manipuladas por el equipo digital como una PC. Un convertidor analógico-digital (ADC, por sus siglas en inglés) es un chip que proporciona una representación digital de una señal analógica en un instante de tiempo. Las señales analógicas varían continuamente con el tiempo y un ADC realiza "muestras" periódicas de la señal a una razón predefinida. Estas muestras son transferidas a una PC a través de un bus, donde la señal original es reconstruida desde las muestras en software

Para lograr la conversión de una señal analógica a una señal digital se tiene que hacer una serie de pasos para poder llevarse a cabo dicha conversión los cuales son:

Muestreo.

El muestreo (en inglés, sampling) consiste en tomar muestras periódicas de la amplitud de onda. La velocidad con que se toman esta muestra, es decir, el número de muestras por segundo es lo que se conoce como frecuencia de muestreo y está en función del teorema de Nyquist, que indica que la frecuencia de muestreo f_s será el doble de la frecuencia de la señal a muestrear.

Cuantificación

La cuantificación lo que hace es convertir una sucesión de muestras de amplitud continua en una sucesión de valores discretos preestablecidos según el código utilizado. Durante el proceso de cuantificación se mide el nivel de tensión de cada una de las muestras, obtenidas en el proceso de muestreo, y se le atribuye a un valor finito (discreto) de amplitud, seleccionado por aproximación dentro de un margen de niveles previamente fijado. En la **Figura 2.2** Proceso de cuantificación se muestra el proceso de cuantificación de una señal analógica.

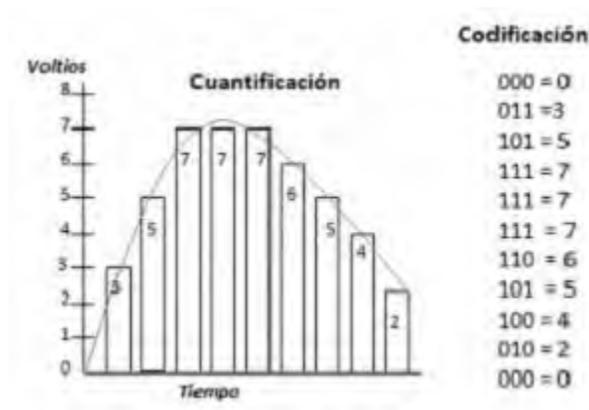


Figura 2.2 Proceso de cuantificación

Codificación.

La codificación consiste en la traducción de los valores de tensión eléctrica analógicos que ya han sido cuantificados (ponderados) al sistema binario, mediante códigos preestablecidos. La señal analógica va a quedar transformada en un tren de impulsos digital.

Bus de PC.

El hardware de adquisición de datos se conecta a una PC a través de una ranura o puerto. El bus de la PC sirve como la interfaz de comunicación entre el hardware de adquisición de datos y la PC para pasar instrucciones y datos medidos. Los buses de PC más comunes son USB, PCI, PCI Express y Ethernet. Los dispositivos DAQ igual están disponibles para 802.11 Wi-Fi para comunicación inalámbrica.

Función de la PC en un sistema DAQ

Una PC se usa como dispositivo intermedio entre un software programable y un dispositivo DAQ ya que el software programable es el encargado de controlar las operaciones del dispositivo DAQ y es usado para procesar, visualizar y almacenar datos de medida.

Software controlador

El software controlador ofrece al software de aplicación la habilidad de interactuar con un dispositivo DAQ. El software simplifica la comunicación con el dispositivo DAQ al abstraer comandos de hardware de bajo nivel y programación a nivel de registro. Generalmente, el software controlador DAQ expone una interfaz de programación de aplicaciones (API) que es usada en un entorno de programación para construir software de aplicación.

Software de aplicación

El software de aplicación facilita la interacción entre la PC y el usuario para adquirir, analizar y presentar datos de medidas. Las aplicaciones creadas en el software pueden ser diseñadas con funcionalidades predefinidas o diseñadas con funcionalidades personalizadas. Las aplicaciones personalizadas generalmente son usadas para automatizar múltiples funciones de un dispositivo DAQ, realizar algoritmos de procesamiento de señales y mostrar interfaces de usuario personalizadas.

Capítulo 3 Sistema embebido NI myRIO-1900 y software LabVIEW

3.1 Sistema embebido NI myRIO

El Dispositivos myRIO-1900 es un sistema embebido que cuenta con diversas características para el diseño de sistemas electrónicos. Algunas de sus principales características son: conectores de E/S, tecnología WIFI, un procesador ARM en tiempo real dual-core y una FPGA Xilinx, acelerómetro integrado, al igual cuenta con su propio software para el diseño de aplicaciones.

La tarjeta de National Instruments myRIO-1900 es un dispositivo portable de E/S reconfigurable (RIO, portable reconfigurable I/O, por sus siglas en ingles), la cual es una herramienta para hacer diseños de control, robótica y sistemas mecatrónicos.

Esta tarjeta tiene implementado un sistema operativo basado en LINUX que interpreta los comando compilados desde el software LabVIEW de una PC.

La tarjeta cuenta con un módulo WIFI incorporado lo cual nos permite establecer una conexión remota entre dicho dispositivo y una PC, esto se debe a que la tarjeta se puede configurarse como un AP (Access Point, por sus siglas en inglés) para que un PC puede conectarse vía WIFI y por medio del protocolo TCP/IP enviar datos entre la conexión establecida de la tarjeta y a la PC por medio de la red creada en la NI myRIO. En la Figura 3.1 se muestra la imagen de la tarjeta NI-myRIO 1900.



Figura 3.1 Sistema embebido NI myRIO-1900.

3.2 ¿Qué es un sistema embebido?

Es un dispositivo computacional el cual está diseñado para realizar una o varias tareas en específico, comúnmente se encuentran en dispositivos computacionales en tiempo real. Estos dispositivos, aunque se asemejan a los sistemas computacionales de propósito general (PC), que, si bien también están formadas por una combinación de hardware y software, las computadoras no son diseñadas para un uso específico como son los sistemas embebidos. Una de las principales diferencias que hay entre los sistemas computacionales convencionales y los sistemas embebidos, es que éstos últimos están conectados frecuentemente a ambientes físicos a través de sensores y actuadores.

Los sistemas embebidos se construyen generalmente con las siguientes características:

- Eficiencia en cuanto al consumo de energía, peso y costo.
- Deben reaccionar a cambios en el ambiente.
- Trabajar en tiempo real, realizar cálculos en tiempo real sin ningún retraso, es decir, en tiempos fijos ante cualquier eventualidad.
- Desarrollan un funcionamiento específico.
- respuestas a una solución óptima de tareas a resolver.
- Son sistemas híbridos, poseen partes analógicas y digitales.
- Capaz de Realizar tareas relativamente complejas.
- Estar constituido por módulos estrictamente indispensables para desarrollar la función encomendada.
- dedicados para ciertas aplicaciones
- Interfaces de usuario dedicadas (sin ratón, teclado y pantalla).

3.2.1 Arquitectura de un sistema embebido.

Un sistema embebido es básicamente un dispositivo controlado por una PC el cual es diseñado para realizar algunas tareas específicas. En la mayoría de los casos estas tareas son para el control o monitoreo en tiempo real de máquinas o procesos. La arquitectura básica empleada en un sistema embebido posee una arquitectura similar a la de una PC. Los componentes que integran la arquitectura de un sistema embebido se muestran en la Figura 3.2

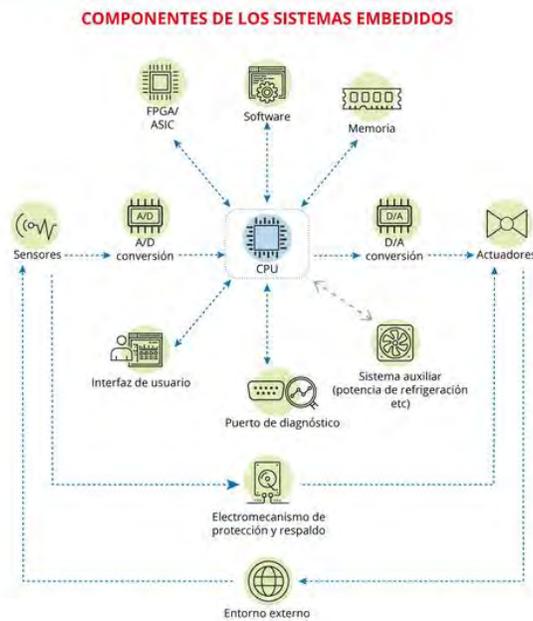


Figura 3.2 Arquitectura de un sistema embebido

Microprocesador.

Él cual es el encargado de realizar las operaciones de cálculo principales del sistema. Ejecuta código para realizar una determinada tarea y dirige el funcionamiento de los demás componentes por los que está compuesto dicho sistema.

Memoria.

Memoria de Acceso Aleatorio (RAM, por sus siglas en inglés): Es un chip de memoria de lectura y escritura alimentado con una pila donde se almacena el tipo y

ubicación de los dispositivos conectados a la placa madre (dispositivo de almacenamiento, puertos de entrada y salida, etc.).

Memoria de solo lectura (ROM, por sus siglas en inglés): es la memoria donde se encuentra el sistema básico de entrada y salida (BIOS, por sus siglas en inglés), el cual es el encargado de alojar el código para inicializar el sistema y para poner en comunicación los distintos elementos de la tarjeta madre.

Memoria Caché: es una memoria más rápida que la principal y tiene la función de almacenar los datos y el código compilados previamente. Dado que el sistema realiza microtarefas, muchas veces repetitivas, la memoria caché permite ahorrar tiempo ya que no hará falta ir a memoria principal si el dato o la instrucción ya se encuentra en la memoria caché.

Chip Set

Es el encargado de controlar las interrupciones dirigidas al microprocesador, tiene acceso directo a memoria (DMA), además de ofrecer temporizadores. La memoria RAM y el reloj de tiempo real están integrados en el Chip Set.

Reloj del sistema

El reloj del sistema es un componente muy importante de un sistema integrado ya que todos los procesos en un sistema integrado de funcionan en ciclos de reloj y requieren información de tiempo precisa. Este reloj consiste en generalmente un oscilador y algunos circuitos asociados. El reloj de un sistema embebido está integrado junto con la memoria RAM.

Puertos de Entrada / Salida (I/O)

Son pines (nodos) en los que los dispositivos de entrada y salida se pueden conectar y pueden intercambiar información con la memoria y el procesador central. Los puertos contienen en sí un número definido de registros, los cuales se utilizan para el almacenamiento temporal de varios tipos de datos.

3.3 ¿Qué es un sistema operativo de tiempo real (RTOS)?

Un sistema operativo de tiempo real es una clase de sistemas operativos que cumple unos determinados requisitos para dar soporte a aplicaciones de tiempo real. Un sistema operativo de tiempo real (RTOS, por sus siglas en inglés) o también llamado Núcleo de tiempo real (RTK, por sus siglas en inglés) tiene que cumplir las siguientes características:

Multiprogramación: Esta característica se refiere a que un RTOS tiene que dar soporte a varias tareas de una aplicación. Las aplicaciones de tiempo real son multitarea y requieren un núcleo que permita crear tareas, gestionarlas atendiendo a los plazos de éstas y tener la capacidad para no atender a una tarea de menor prioridad.

Un RTOS debe ofrecer comportamiento determinista en todos los servicios del sistema. Son especialmente significativas las medidas de ciertas operaciones de bajo nivel que realiza muy frecuentemente el sistema y que requieren, además de estar acotadas, un coste temporal bajo. Algunas de estas medidas críticas son:

- Latencia de la gestión de interrupciones garantizada
- Cambio de contexto acotado
- Baja sobrecarga introducida por el planificador
- Gestión eficiente del reloj y los temporizadores
- Tiempo de respuesta del hardware

Un RTOS facilita la ejecución de un sistema de tiempo real pero no garantiza que el producto final cumpla las restricciones temporales. Esto requiere un desarrollo del software de aplicación correcto y analizable.

3.4 ¿Qué es un FPGA?

Los arreglos de compuertas programables en campo (FPGAs, por sus siglas en inglés) son chips de silicio reprogramables, ofrecen una alta velocidad de procesamiento por hardware y fiabilidad, sin requerir altos volúmenes de recursos. Una FPGA tiene la misma capacidad que un software de ejecutar instrucciones y

operaciones similares a los sistemas basados en procesadores, pero no están limitados a la cantidad de núcleos disponibles.

A diferencia de los procesadores, su sistema de procesamiento de los FPGAs es verdaderamente paralelo por naturaleza, así las diferentes operaciones de procesamiento no tienen que competir por los mismos recursos de sistema. Cada tarea de procesamiento es independiente y es asignada a una sección del chip y puede ejecutarse de manera autónoma sin ser afectada por otros bloques de lógica. Como resultado, el rendimiento no se ve afectado cuando se agregan otros procesos. Uno de los beneficios de los FPGAs ante los sistemas basados en procesador es que la lógica de aplicación es implementada en circuitos de hardware en lugar de ejecutarse secuencialmente por software. La Figura 3.3 representa el sistema procesamiento que utiliza una FPGA.

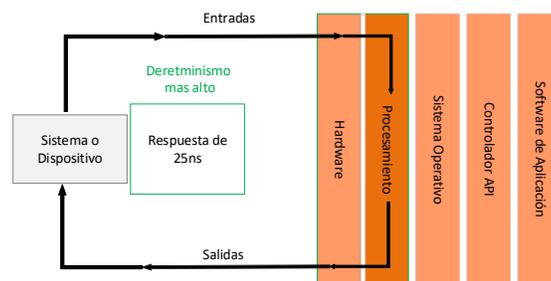


Figura 3.3 sistema de procesamiento de un FPGA

3.5 Arquitectura de una FPGA

Las FPGA están compuestas por tres componentes los cuales son: los bloques de lógica, las interconexiones programables y los bloques de E/S (entrada y salida). La Figura 3.4 muestra la arquitectura básica de una FPGA.

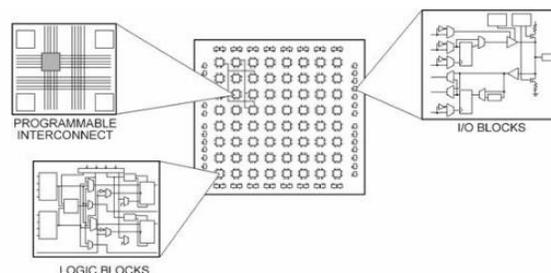


Figura 3.4 Arquitectura de un FPGA

Los bloques de lógica configurables

Los bloques de lógica configurables (CLBs, por sus siglas en inglés), son la unidad de lógica básica de un FPGA. Algunas veces referido como segmentos o células de lógica, los CLBs están hechos de dos componentes básicos: flip-flops y tablas de consulta (LUTs, por sus siglas en inglés).

Interconexiones programables

interconexiones programables permite a los bloques lógicos de un FPGA ser interconectados según la necesidad del diseñador del sistema. Estos bloques lógicos e interconexiones pueden ser programados después del proceso de manufactura por el usuario/diseñador, de esta forma una FPGA puede desempeñar cualquier función lógica necesaria.

Bloques de E/S

Los bloques de E/S Los bloques de E/S (IOBs por sus siglas en inglés) están agrupados en lo que se llaman bancos de bloques de E/S (grupos de IOBs). Dependiendo del FPGA, cada banco puede tener entre 20 y 40 IOBs. Cada banco de IOBs tiene su propia tensión de alimentación (VCCO) y su propia tensión de referencia VREF que son comunes a todos los IOBs. Es por esta disposición de tensiones de VCCO y VREF por banco, que si se quieren usar diferentes estándares de E/S con diferente VCCO, se deberán usar diferentes bancos de E/S

3.6 Descripción general del hardware

El dispositivo NI myRIO-1900 cuenta con múltiples entradas y salidas analógicas y digitales, entrada de audio, alimentación de salida, el procesador en tiempo real y un FPGA integrados en el sistema embebido. En la Figura 3.5 se muestra un diagrama a bloques de los componentes que integran a la tarjeta.

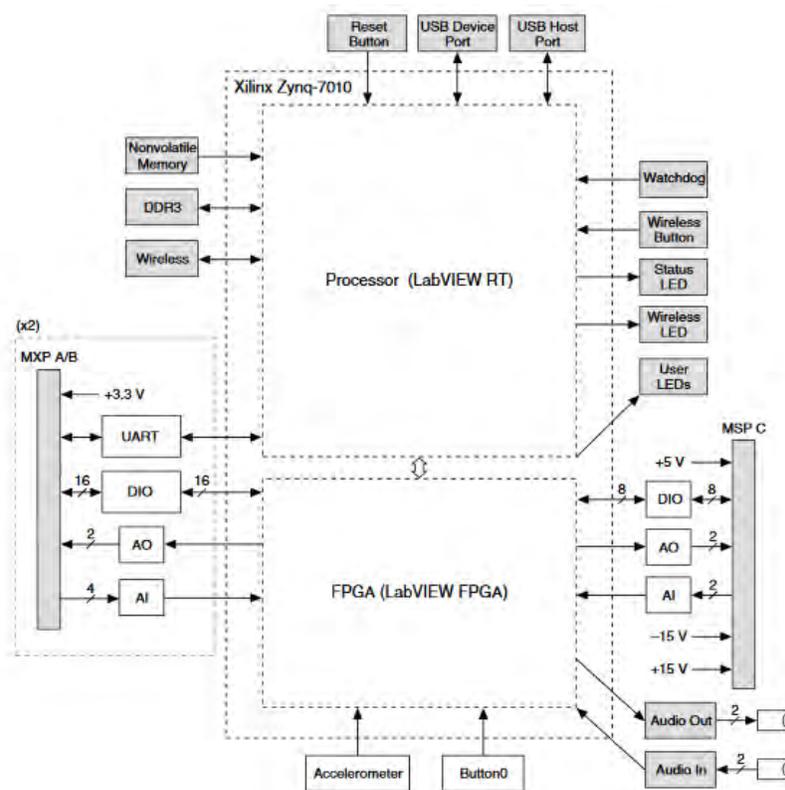


Figura 3.5 Diagrama de bloques componentes de la tarjeta myRIO-1900

3.6.1 Grupo de puertos A B y C de la tarjeta myRIO-1900

La tarjeta NI myRIO-1900 cuenta con dos conectores de expansión tipo MXP. En los puertos A y B de NI myRIO-1900 las señales de los conectores son idénticas, estas señales se pueden distinguir en el software por el nombre del conector, por ejemplo. Conector A/DIO0 y conector B/DIO0. La Figura 3.6, muestra las señales de los conectores A y B. Algunos pines llevan funciones secundarias, así como funciones primarias.

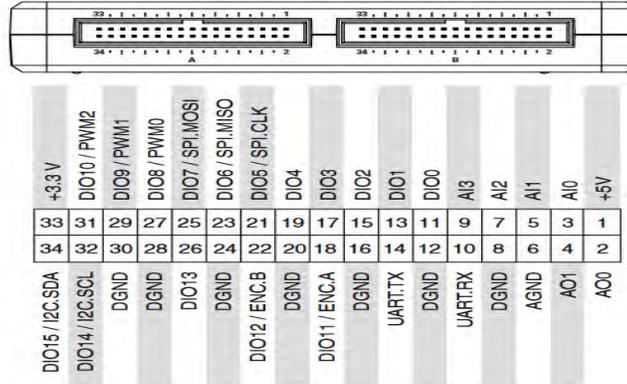


Figura 3.6 Representación de pines de los puertos A y B de la tarjeta myRIO1900

En la **Tabla 3** se muestra las características de los conectores MXP A y B

Nombre de la señal	Referencia	Tipo (Entrada o Salida)	Descripción
+5V	DGND	Salida	+5V de tensión de salida
AI <0...3>	AGND	Entrada	0 a 5 V, referenciado a canales de entrada analógica single-ended
AO <0...1>	AGND	Salida	0 a 5 V, referenciado a canales de salida analógica single-ended
AGND	N/A	N/A	Referenciado para entradas y salidas analógicas
+3.3V	DGND	Salida	+3.3v de tensión de salida
DIO <0...15>	DGND	Entrada o Salida	Líneas digitales de propósito general con 3,3 V de salida, 3,3 V / entrada compatible con 5V.
UART.RX	DGND	Entrada	UART recepción (entrada). Las líneas UART son eléctricamente idénticas a las de DIO
UART.TX	DGND	Salida	UART transmisión (salida) Las líneas UART son eléctricamente idénticas a las de DIO
DGND	N/A	N/A	Referenciado para señales de 5V y 3.3V

En la Figura 3.7 se muestra las señales de captura del conector C (MSP), algunos pines tienen funciones secundarias.

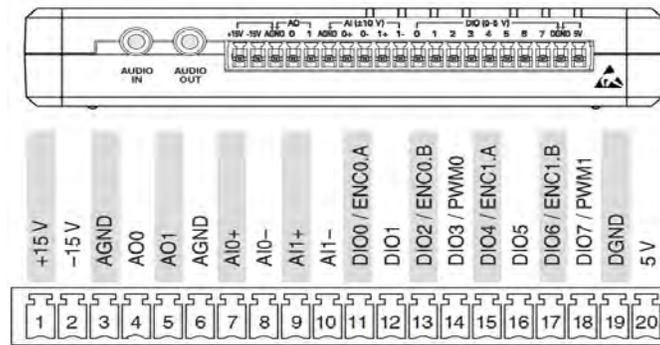


Figura 3.7 Representación de pines en el puerto C de la tarjeta myRIO-1900

En la Tabla 4 se muestra las características del conector C MSP.

Nombre de la señal	Referencia	Tipo (Entrada/Salida)	Descripción
+15V/-15V	AGND	Salida	+15V/-15V tensión de salida
AI0+/AI0-; AI1+/AI1-	AGND	Entrada	± 10V, en canales de entradas analógicas diferenciales.
AO <0...1>	AGND	Salida	± 10V, en canales de salidas analógicas diferenciales.
AGND	N/A	N/A	Referencia para entradas, salidas y +15V/-15V de tensión de salida
+5V	DGND	Salida	+5V tensión de salida
DIO <0...7>	DGND	Entrada o Salida	Líneas digitales de propósito general con 3,3 V de salida, entradas compatibles con, 3,3V/ 5V, referir a las líneas DIO
DGND	N/A	N/A	Referencia para líneas digitales y +5V de tensión de salida.

3.6.2 CANALES DE ENTRADA ANALOGICA

La NI myRIO-1900 tiene canales de entradas analógicas en los conectores A, B y en el conector C, también cuenta con un conector de entrada de audio estéreo. Las entradas analógicas son multiplexadas para un solo convertidor ADC para las muestras de todos los canales.

Los conectores A y B de los puertos de expansión (MXP, por sus siglas en inglés) tienen cuatro canales de entrada analógica referenciados a tierra por conector, AI0-AI3, que se pueden usar para medir señales de 0-5V, el conector C MSP tiene dos canales de entrada de alta impedancia analógica diferencia, AI0 y AI1, que se pueden usar para medir señales hasta ± 10 V. Los niveles de las entradas de audio tienen un rango de escala de ± 2.5 V.

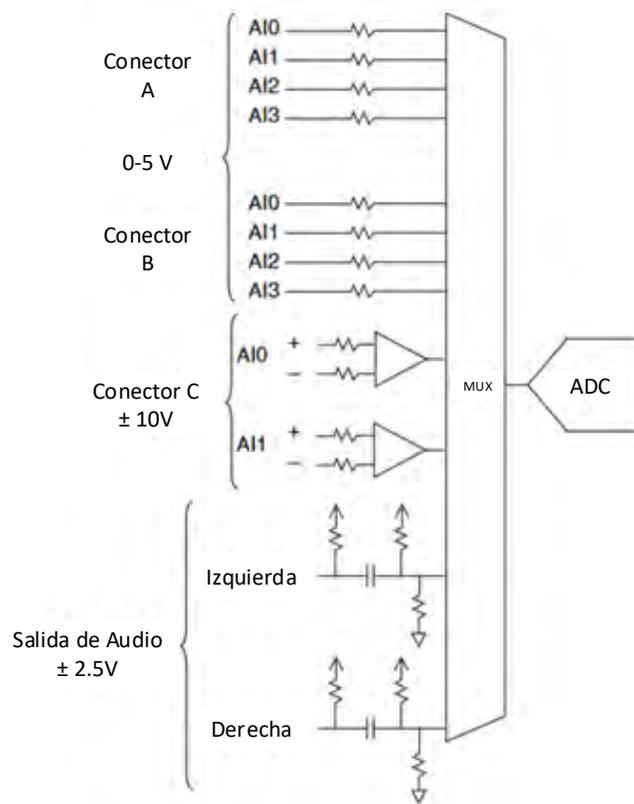


Figura 3.8 Topología de las entradas analógicas de la myRIO-1900

3.6.3 Canales de salida analógica

La NI myRIO-1900 tiene canales de salida analógica en los conectores A, B y conector C y un conector de salida de audio estéreo. Cada canal de salida analógica tiene dedicado un convertidor digital-analógico (DAC, por sus siglas en inglés)

Los conectores A y B MXP tienen dos canales de salida analógica por conector, AO0 y AO1, que pueden ser usados para generar señales de 0-5V. el conector C MSP tiene dos canales de salidas analógica AO0 y AO1 que pueden ser usados para generar señales de ± 10 V.

La Figura 3.9 muestra la topología de las salidas analógicas de la tarjeta myRIO-1900

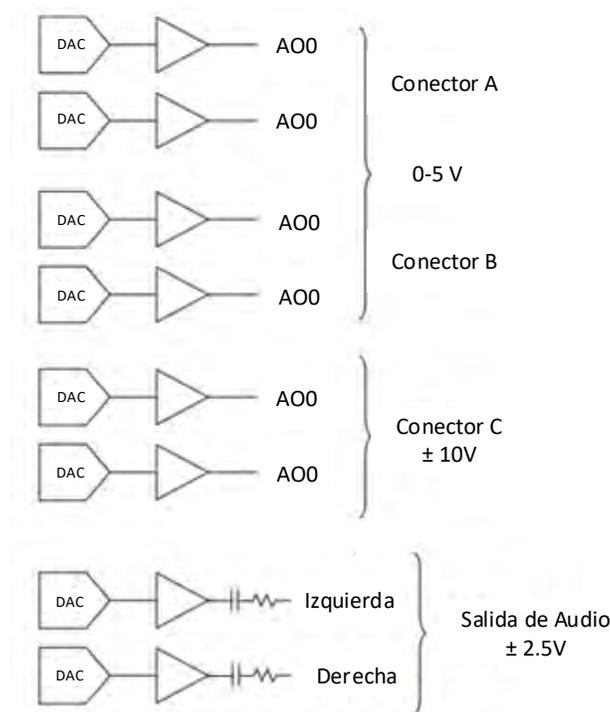


Figura 3.9 Topología de las salidas a analógicas de la myRIO-1900

3.6.4 Conversión de valores de datos sin procesar a voltaje para los puertos MXP.

La tarjeta Ni myRIO-1900 tiene un convertidor Analógico-Digital de 12 bits por lo que el número de combinaciones es de bits del ADC es de $2^{12} = 4096$, esto significa que, cada combinación de bits del ADC corresponde a un voltaje analógico. Con respecto a la ecuación (3.1), para calcular el valor de lectura máxima de los canales AI y AO de los canales MXP el valor del bit menos significativo (LSB Weight, por sus siglas en ingles).

LSB Weight:

$$LSB\ Weight = 5V \div 2^{12} = 1.221mV \quad (3.1)$$

Esto se consigue dividiendo el valor de datos sin procesar (5V) y el número máximo de combinaciones del ADC ($2^{12} = 4096$).

El valor resultante $1.221mV$ es el valor del bit menos significativo el cual servirá para calcular el valor de lectura máxima (Maximum Reading, del inglés) que nos permiten obtener los canales MXP, este valor se puede calcular con la ecuación (3.2).

$$Maximum\ Reading = 4095 * 1.221\ mV = 4.999\ V \quad (3.2)$$

Para calcular el valor de lectura máxima se multiplica el número máximo de combinaciones del ADC ($2^{12} = 4096$) por el valor del bit menos significativo el cual se obtuvo en la ecuación (3.1) cuyo resultado fue $1.221mV$, con el resultado de la multiplicación obtendremos que la lectura máxima de los canales MXP es de $4.999\ V$.

3.7 software LabVIEW

La programación de la tarjeta myRIO se lleva a cabo a través del software de programación grafica LabVIEW (Laboratorio Virtual Instrument Egeineering Workbench) 2017 myRIO. Está versión del software está dedicada para la programación de sistemas embebidos de la empresa National Instruments (NI) para los controladores myRIO. El entorno de programación que brinda LabVIEW es un entorno gráfico, el cual está basado en iconos y cables gráficos e intuitivos en lugar del lenguaje textual (programación escrita), Las aplicaciones creadas en este software son llamadas VIs o Instrumentos Virtuales (Virtual Instruments, por sus siglas en inglés). Los instrumentos virtuales pueden ser personalizados y enfocados a las necesidades técnicas que se requieran. Los instrumentos virtuales están divididos en dos partes principales, una llamada "Panel frontal", y otra "Diagrama de bloques". LabVIEW está principalmente enfocado a la instrumentación virtual, es decir, a la medición, análisis, monitoreo y visualización de datos medidos mediante un hardware especializado.

La **Figura 3.10** Pantalla de inicio de LabVIEW para dispositivos myRIO. **Figura 3.10** muestra la ventana de inicio del software LabVIEW para dispositivos myRIO



Figura 3.10 Pantalla de inicio de LabVIEW para dispositivos myRIO

3.8 Instrumento Virtuales (VIs)

Los programas que se desarrollan en LabVIEW son llamados instrumentos virtuales, o VI, debido a que la apariencia y las operaciones que ejecuta imita el comportamiento de un instrumento físico, como puede ser el caso de un multímetro o de un osciloscopio. Las funciones principales de un VI son adquirir, procesar, almacenar, la visualización y comunicación de datos. El termino virtual se refiere a que el instrumento de medición no existe, comparado con instrumento de medición físico. Un instrumento virtual no es una simulación como tal, sino que es un componente de un sistema basado en software el cual es el encargado de simular en la pantalla de un pc la apariencia de un instrumento real.

En LabVIEW se debe iniciar por la construcción de un panel frontal o interfaz de usuario principalmente con botones, controles e indicadores. Los controles están representados gráficamente por dispositivos de entrada como interruptores o botones, los indicadores están representados por dispositivos de salida como gráficas, medidores, termómetros, etc. Después de terminar con el diseño del panel frontal el siguiente paso es añadir el “código” utilizando instrumentos virtuales y estructuras para controlar los objetos añadidos en el panel frontal, esta parte tiene como nombre diagrama de bloques. La siguiente Figura 3.11 muestra el proceso para el diseño de un Vi7

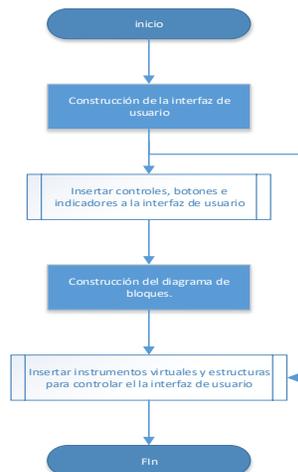


Figura 3.11 proceso para el diseño de un instrumento Virtual

3.8.1 Panel frontal

Cuando se empieza a trabajar en un nuevo VI, se extiende en la pantalla de la PC dos ventanas del programa LabVIEW la primera tiene como nombre Panel Frontal y la segunda Diagrama de bloques. La ventana del Panel Frontal es la interfaz donde el usuario puede interactuar con el programada que se está ejecutando y la ventana de Diagrama de Bloques es donde se desarrollan las funciones que se van a llevar a cabo en el Panel Frontal. En la Figura 3.12 se muestra el entorno y los componentes del panel frontal y el Diagrama de Bloques.

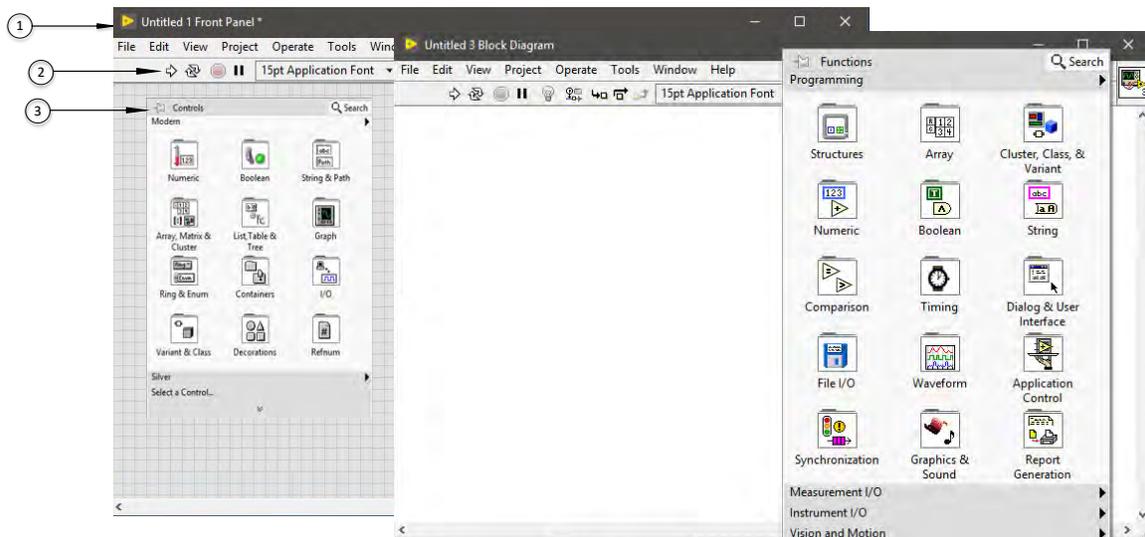


Figura 3.12 (1) Ventana del Panel Frontal, (2) Barra de Herramientas, (3) paleta de Controles.

3.8.2. Paleta de controles.

En la paleta de Controles se puede encontrar los controles e indicadores que utiliza para hacer el diseño del panel frontal. Se Puede tener acceso a la paleta de Controles desde la ventana del panel frontal haciendo clic en **View >> Controls Palette** o al dar clic derecho en cualquier espacio en blanco en la ventana del panel frontal. Como se muestra en las figuras Figura 3.13 y Figura 3.13.

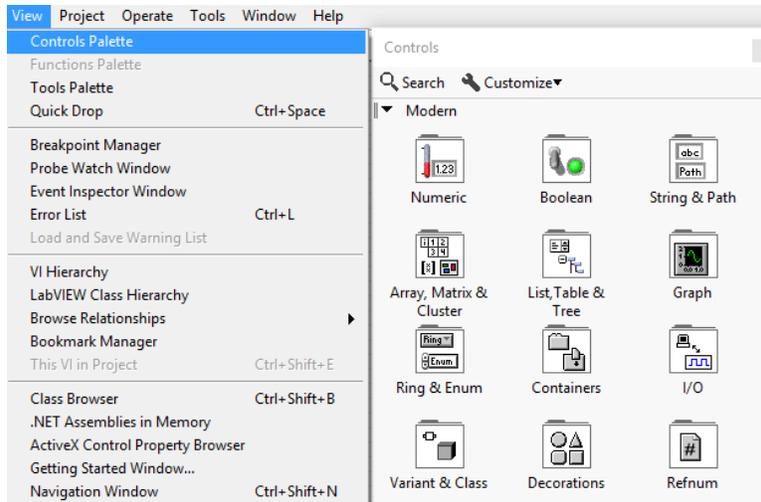


Figura 3.13 Acceso a la paleta de controles.

La paleta de Controles tiene varias secciones; se pueden usar algunas o todas estas dependiendo de las necesidades que se requieran para el diseño de la interfaz del usuario. en la Figura 3.14 se muestra la paleta de Controles con todas las secciones expuestas y la sección “Modern” extendida.

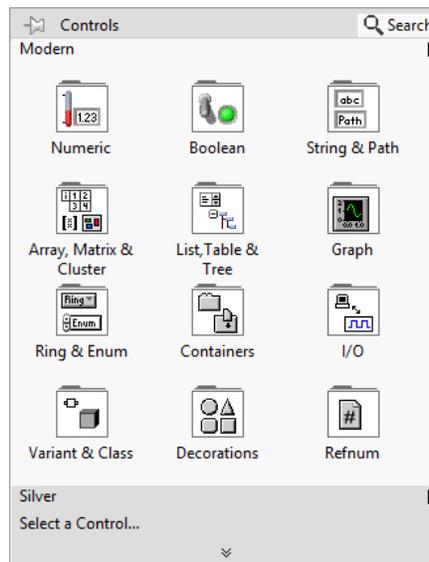


Figura 3.14 Paleta de controles

3.8.2.1 Controles e Indicadores.

En el panel frontal se encuentran los indicadores y controladores para el diseño que se requiera para la interfaz de usuario. Los indicadores y controles son las terminales interactivas de entrada y salida del VI.

Los controles pueden ser representados como perillas, botones, barras deslizantes los cuales simulan dispositivos de entrada de instrumentos físicos al igual que son utilizados para suministrar datos al diagrama de bloques del VI y otros dispositivos de entrada.

Los indicadores simulan dispositivos de salida de instrumentos físicos y muestran los datos que el diagrama de bloques adquiere o genera mediante un hardware especializado. Los indicadores pueden ser representados por gráficas, LEDs, etc.

La Figura 3.15 muestra un VI que tiene dos controles: Número de muestras y el retraso en segundos (s). De igual forma tiene un indicador el cual es una gráfica que muestra la amplitud y el tiempo, llamada Gráficas de temperatura.

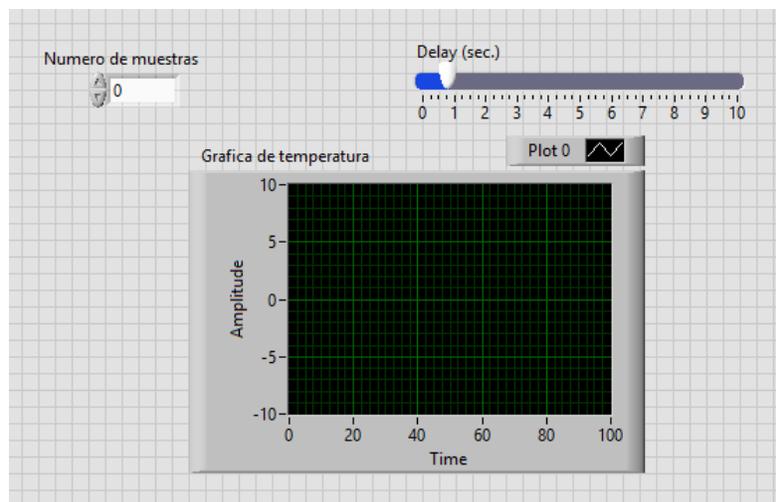


Figura 3.15 Diseño de un VI que contiene indicadores y controles en su interfaz de usuario

En el ejemplo que se muestra en la Figura 3.19 el VI genera valores que se muestran en el indicador basado en el diagrama creado en la ventana de diagrama de bloques. El usuario puede cambiar los valores de entrada para los controles “Número de muestras y Delay (sec.)” que se muestran en la Figura 3.15 y verlos reflejados en el indicador “Gráfica de temperatura”.

Cada tipo de control o indicador está vinculado a un tipo específico de dato que está asociado con él. Por ejemplo, el deslizador horizontal Delay que aparece en la Figura 3.15 tiene como tipo de dato numérico. Los tipos de datos utilizados con mayor frecuencia son numéricos, valor Booleano y cadena de caracteres.

3.8.2.2 Controles e indicadores numéricos

Los tipos de datos numéricos en general son representados por números de varios tipos como un entero, real, con punto decimal, etc. Las herramientas más comunes en la paleta de controles son la herramienta de control numérico y de indicador numérico, como se muestra en la Figura 3.16

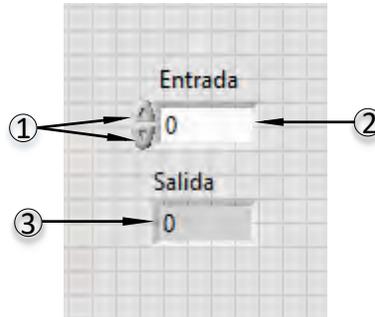


Figura 3.16 (1) Incremento/Decremento del valor numérico, (2) Control, (3) Indicador

Se puede modificar el valor numérico del control “Entrada”, haciendo clic en las botones Incremento/Decremento o haciendo doble clic sobre el numero e ingresar un nuevo valor y presionar la tecla <Enter>.

3.8.2.3 Controles e Indicadores Booleanos

Los tipos de datos booleanos son representados por datos que solamente tiene dos estados posibles, como Verdadero y Falso. La utilización de Los indicadores y controles booleanos son para proporcionar y visualizar valores Booleanos. Las herramientas Booleanas que están representadas en la paleta de controles simulan interruptores, botones y LEDs. En la siguiente figura un interruptor vertical y un LED.

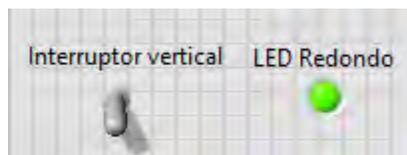


Figura 3.17 Indicador y control Booleano

3.8.2.4 controles e indicadores de cadenas de caracteres

Los tipos de “datos cadena de caracteres” es una secuencia de caracteres ASCII. La herramienta de control de caracteres en la paleta de controles se puede usar

para que el usuario escriba un texto como puede ser una contraseña o nombre de usuario, el indicar de caracteres se usa para mostrar un texto al usuario, en la Figura 3.18 se muestran los elementos más comunes de cadena de caracteres como son las tablas y los cuadros de texto.



Figura 3.18 indicadores y controles de cadena de caracteres.

Cada control o indicador tiene un tipo de datos asociado con él. Los tipos de datos utilizados con mayor frecuencia son numéricos, valor Booleano y cadena de caracteres

3.8.3 Diagrama de Bloques

El diagrama a bloques está constituido por terminales, subVIs, funciones, constantes, estructuras y cables, los cuales funcionan en conjunto con los diferentes objetos que conforman el diagrama a bloques que se va a diseñar. En la Figura 3.19 se muestra la representación de algunos componentes básicos de un diagrama de bloques.

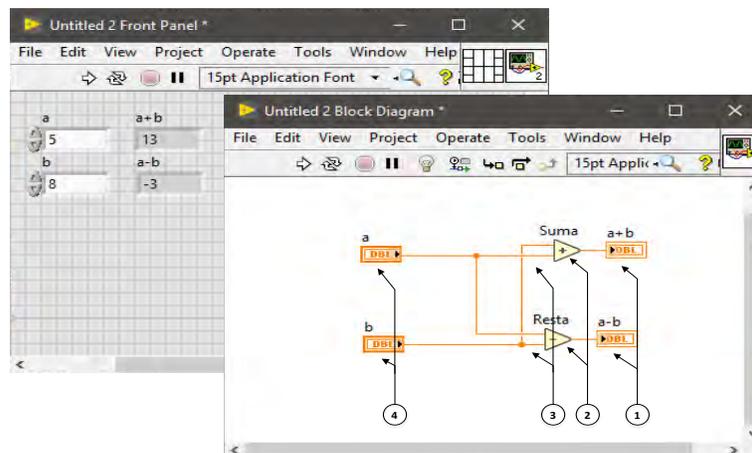


Figura 3.19 (1) Indicadores (2) Cables (3) Nodos (4) Controles.

3.8.3.1 Ventana de Diagrama de Bloques

Cuando se haya terminado de crear el diseño de la ventana del panel frontal, el siguiente paso es hacer la programación de manera gráfica con funciones que permiten controlar los objetos que se añadieron en el panel frontal. En la Figura 3.20 se muestra el código fuente grafico del Diagrama de Bloques.

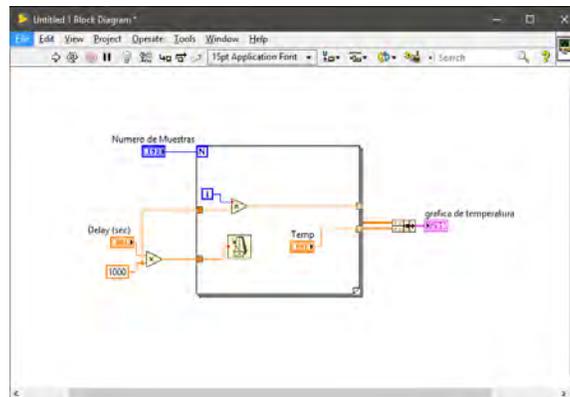


Figura 3.20 Código fuente gráfico.

3.8.3.2 Terminales de Diagrama de Bloques

Los objetos que son agregados en la ventana del Panel Frontal se reflejan en la ventana de Diagrama de Bloque como terminales. Las terminales son puertos de entrada y salida los cuales trabajan en conjunto entre el panel frontal y el diagrama de bloques en el intercambio de información.

Los diferentes tipos de terminales son: terminales de control o indicador y las terminales de nodo. Donde las terminas de control e indicador son las que interactúan directamente entre la ventana del panel frontal y la del diagrama a bloques. Un ejemplo claro se muestra en la Figura 3.19 donde los datos que se le proporcionan a los controles a y b del panel frontal pasan al diagrama de bloques a través de las terminales de control. Entonces los datos son ingresados a las funciones de Suma y Resta. Cuando las funciones de suma y resta terminen los cálculos, obtienen datos nuevos. Los datos nuevos son enviados a la terminal de indicador, donde los datos nuevos son reflejados en los indicadores del panel frontal (a+b y a-b).

Capítulo 4 Diseño del instrumento virtual.

El instrumento virtual fue realizado en el software LabVIEW 2017 myRIO. ya que esta versión del software está dedicada a la programación de controladores myRIO. El instrumento virtual consiste en habilitar dos puertos analógicos del conector A de la tarjeta NI myRIO-1900 para la captura de datos de una señal de corriente y voltaje a través de sensores especializados para medir estos tipos de señales. Las señales procesadas por la tarjeta se visualizarán en la pantalla de una PC y serán almacenadas en una memoria tipo USB colocada en el puerto de la tarjeta.

La tarjeta myRIO se configuró como punto de acceso (AP, por sus siglas en inglés, Access Point) para la transferencia de datos y programación por medio de WiFi. De igual forma se configuró como un servidor FTP para visualizar y descargar los archivos guardados en la memoria USB. Estos archivos contienen los datos en el tiempo de la medición de corriente, voltaje con esta información se calcula la potencia. Los archivos se guardan en un formato tipo CSV los cuales podrán ser abiertos por un visualizador de textos o una hoja de cálculo para su manipulación.

4.1 configuración de la tarjeta myRIO-1900 como AP WiFi

La configuración de la tarjeta myRIO como AP se realiza conectada a la PC por medio de la conexión USB. Una vez que la PC reconozca la tarjeta aparecerá una ventana como la que se muestra en la Figura 4.1.

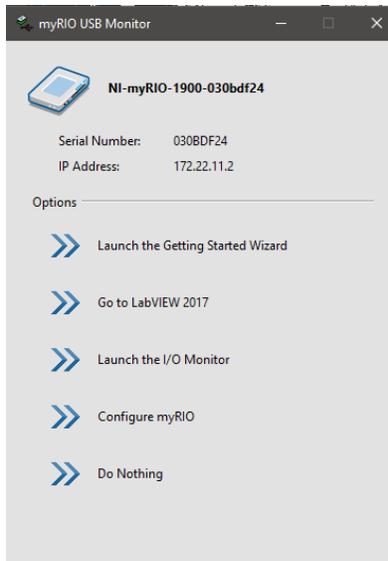


Figura 4.1 ventana de conexión de la tarjeta myRIO-1900 a la PC

Teniendo en cuenta que la tarjeta está conectada a la PC, el siguiente paso es agregarla a la aplicación de NI MAX, la cual es un gestor de dispositivos los cuales funcionan con el software LabVIEW y viene junto con instalación del software.

En la pantalla principal que se muestra en la Fig. 4.2 se puede observar la interfaz NI MAX en donde del lado izquierdo se ven los dispositivos que son reconocidos por el software y listos para utilizarse. Para vincular la tarjeta a esta aplicación se hace clic derecho en la opción **Remote Systems** y aparece un cuadro con tres opciones, donde se selecciona la opción **Create New**, como se muestra en la siguiente Figura 4.2.

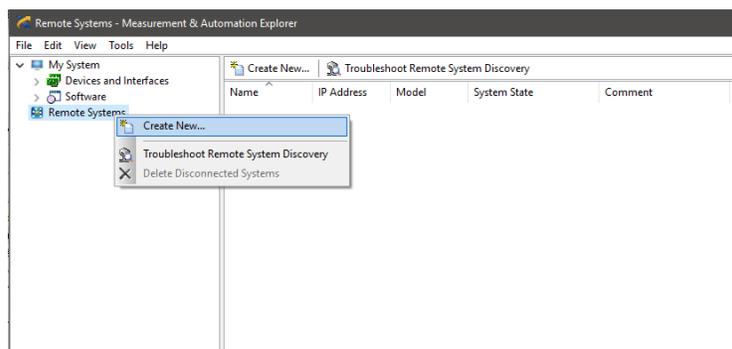


Figura 4.2 Vinculación de la tarjeta myRIO a NI MAX

Al seleccionar la opción **Create New** se desplegará una ventana donde se selecciona la opción **Remote Device (not on the local subnet)** y posteriormente se selecciona el botón **<<Next>>**, como se muestra en la Figura 4.3.

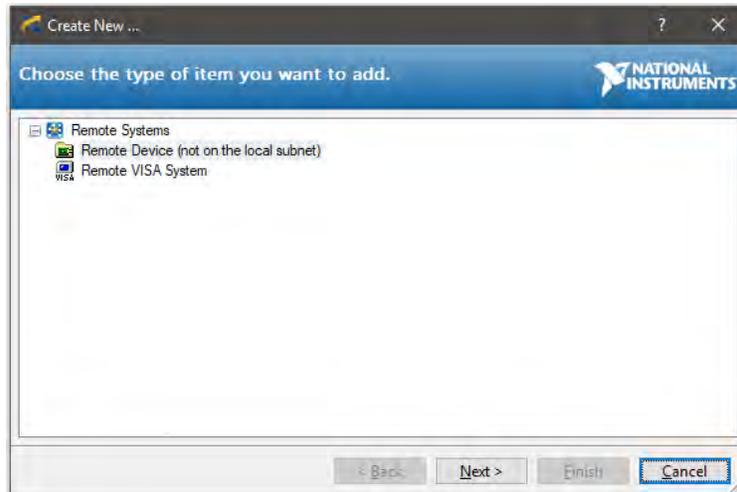


Figura 4.3 selección del tipo de dispositivo para agregar a NI MAX

En la siguiente ventana al presionar el botón **<<Next>>** pide la dirección IP del dispositivo que se agregara, está se obtiene en la ventana que aparece cuando la PC reconoce la tarjeta al conectarla por medio del cable USB, como se muestra en la Figura 4.1 en la cual se puede apreciar la dirección IP del dispositivo. La dirección IP se escribe de la misma forma como se muestra en la figura antes mencionada y se presiona nuevamente el botón **<<Next>>**, una vez presionado el botón se desplegará otra ventana que muestra el nombre, dirección IP y modelo del dispositivo lo cual indica que el dispositivo se agregó correctamente a la aplicación NI MAX tal como se muestra en la Figura 4.4.

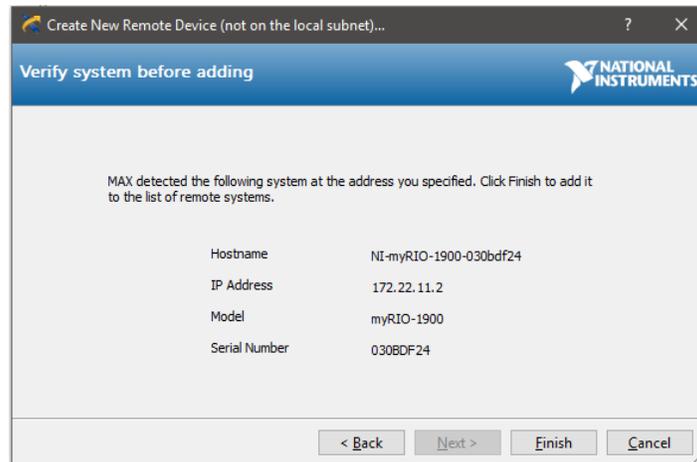


Figura 4.4 Vinculación exitosa de la tarjeta myRIO a NI MAX

Después de visualizar la información del dispositivo, se presiona el botón **<<Finish >>** y se observa que el dispositivo ahora aparece del lado izquierdo en la sección de **Remote Systems**. En la Figura 4.5 se muestra el dispositivo agregado a la aplicación **NI MAX**.

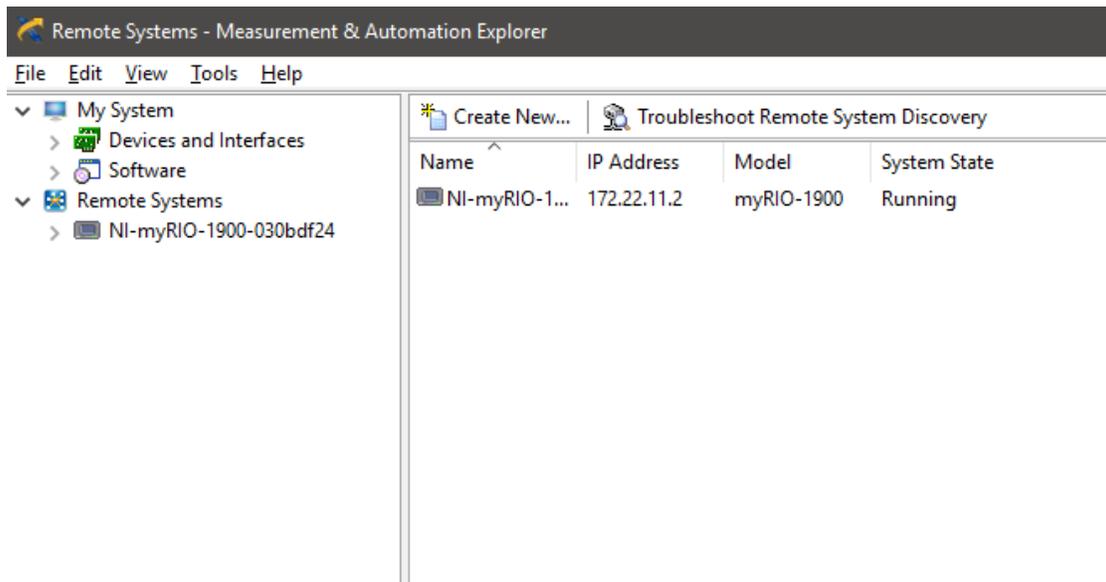


Figura 4.5 Tarjeta agregada a la aplicación NI MAX

Después de agregar la tarjeta a la aplicación, se selecciona el icono de dispositivo myRIO que está ubicado del lado izquierdo, al seleccionarlo se observa en el centro de la ventana, los ajustes del dispositivo, ajustes de inicio y los recursos del sistema. En la Figura 4.6 se muestra los ajustes predeterminados y recursos del sistema de la tarjeta.

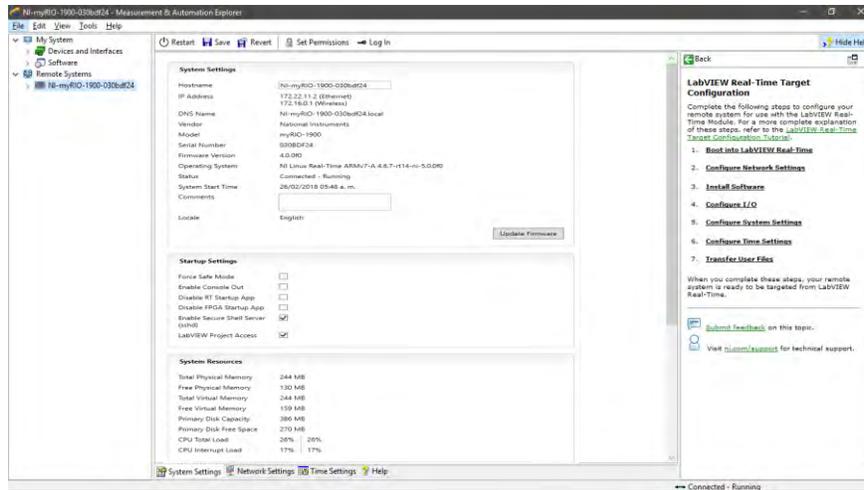


Figura 4.6 Configuración y recursos de la tarjeta NI myRIO-1900

Para configurar la tarjeta myRIO-1900 como AP, se selecciona la opción **Network Settings** que está ubicado en la parte inferior en el centro de la ventana de la aplicación **NI MAX**, esta opción muestra los adaptadores de red del dispositivo, el primario que es conexión por medio del cable USB a la PC y el adaptador inalámbrico. En la Figura 4.7 se muestra la configuración predeterminada de los adaptadores de red de la tarjeta myRIO.

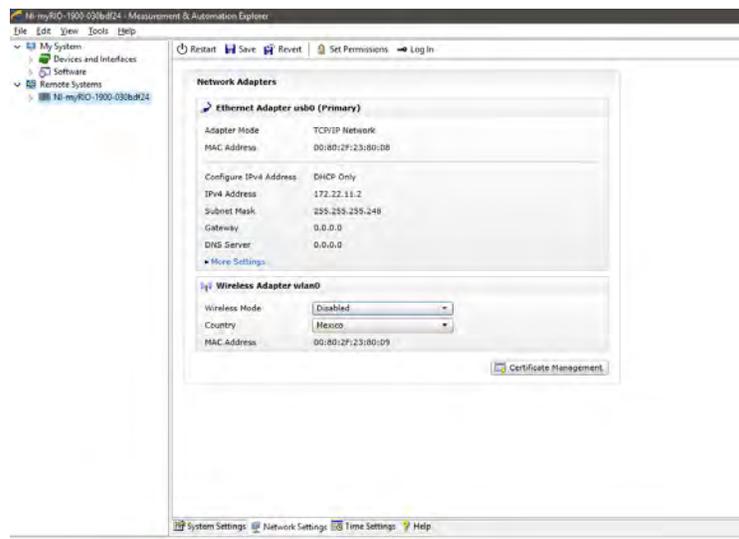


Figura 4.7 Configuración predeterminada de los adaptadores de red de la tarjeta NI myRIO-1900

En la sección de **Wireless Adapter Wlan0** se configura el dispositivo como punto de acceso (AP), para la configuración del dispositivo se hace clic en el menú desplegable de **Wireless Mode** y se selecciona la opción **Create Wireless**

Network. Se despliegan opciones donde se configura la ciudad de ubicación, nombre, seguridad y contraseña para la red, al igual se puede configurar la opción para que la tarjeta asigne direcciones IP mediante DHCP. En la Figura 4.8 se muestra la configuración realizada para que la tarjeta funcione como un punto de acceso.

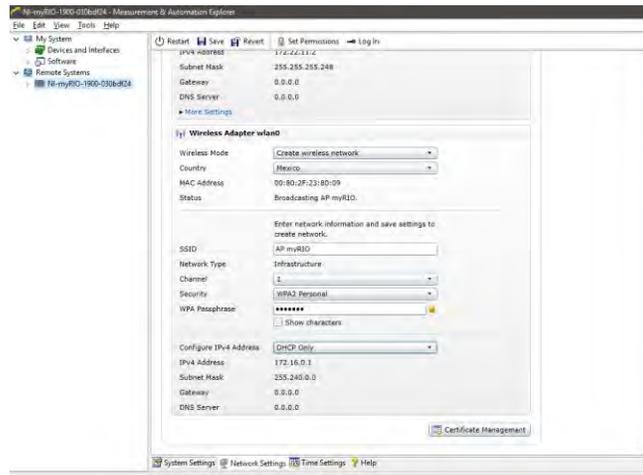


Figura 4.8 configuración para utilizar la tarjeta NI myRIO como punto de acceso

Para iniciar la transferencia de datos y poder programar la tarjeta se tiene que establecer una conexión WiFi con la PC, para esto se busca la red creada en los dispositivos WiFi disponibles desde la PC que donde se va a programar. La red deberá aparecer con el nombre y contraseña asignados en la configuración hecha en la aplicación NI MAX. En la Figura 4.9 se muestra la conexión WiFi establecida entre la PC y la tarjeta myRIO-1900.

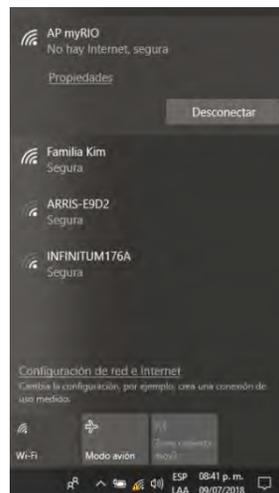


Figura 4.9 conexión WiFi establecida entre la tarjeta NI myRIO-1900 y la PC

4.2 Instalación del servidor FPT en la tarjeta NI myRIO-1900

Para utilizar la tarjeta myRIO-1900 como servidor FTP, primero se deberá instalar el servidor FTP sobre el sistema operativo NI LINUX Real-Time.

A continuación, se ejecuta la aplicación NI MAX. De esta forma en la región de la izquierda de la ventana principal que corresponde a la sección de **Remote Systems** se mostrará la tarjeta que se agregó anteriormente. Se extienden las opciones de la tarjeta NI myRIO y se hace clic derecho sobre la opción **Software** y desplegará una barra con la opción **Add/Remove Software**, como se muestra en la siguiente figura.

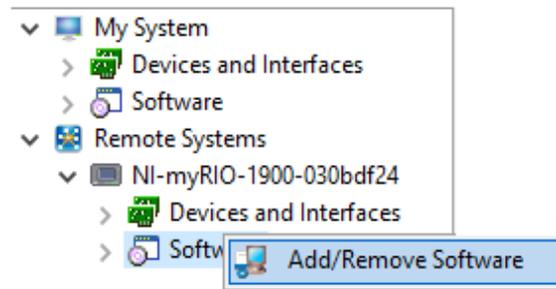


Figura 4.10 Primer paso para instalar el servidor FTP

Al seleccionar esta opción **Add/Remove Software**, aparece una ventana donde pide realizar inicio de sesión con usuario y contraseña. Se presiona el botón **<<Aceptar>>** dejando la contraseña en blanco ya que el sistema viene por default sin contraseña, en la Figura 4.11 se muestra la ventana de inicio de sesión.

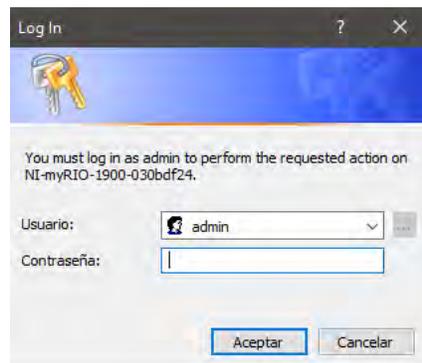


Figura 4.11 Ventana de inicio de sesión como permiso para instalar el servidor FTP.

Al iniciar sesión se abre una ventana en la cual se observan algunas características como la versión de firmware instalado en la tarjeta myRIO y algunos componentes

de software instalados en la tarjeta. A continuación, se selecciona la opción **NI myRIO 17.0 – May 2017** y se oprime el botón **<<Next>>**, como se muestra en la Figura 4.12.

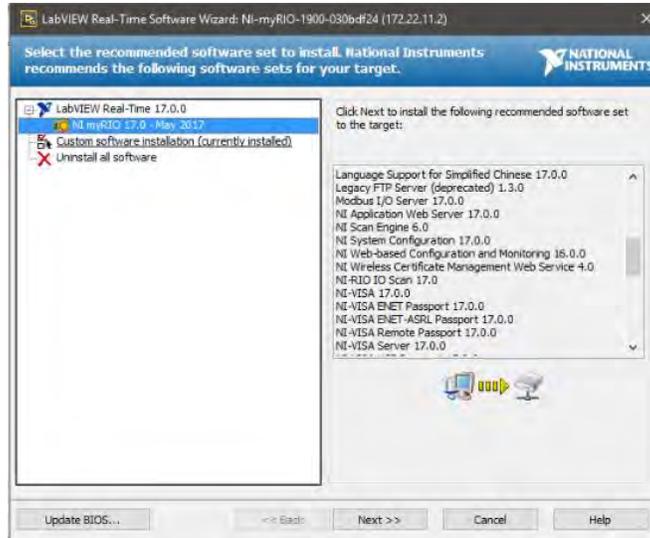


Figura 4.12 segundo paso para instalar el servidor FTP en la tarjeta.

Luego de haber oprimido el botón **<<Next>>** se abre otra ventana, donde se muestran todos los componentes de software que están instalados en la tarjeta. Se extienden las opciones de **Software set Add-ons**. Como se muestra en la Figura 4.13.

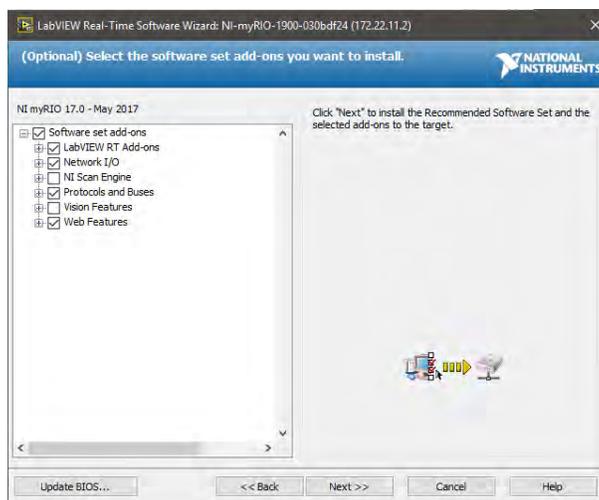


Figura 4.13 opciones que se muestran al desplegar el apartado **Software Add-ons**

En la opción **Software set Add-ons** se ubica el apartado de **Web Features** donde a su vez se ubica la opción **Legacy FTP Server (deprecated) 1.3.0**, el cual es el

componente de software para instalar el servidor FTP, se puede observar que la casilla esta desmarcada lo que indica que no está instalado, para proceder con la instalación se marca la casilla, ver la Figura 4.14.

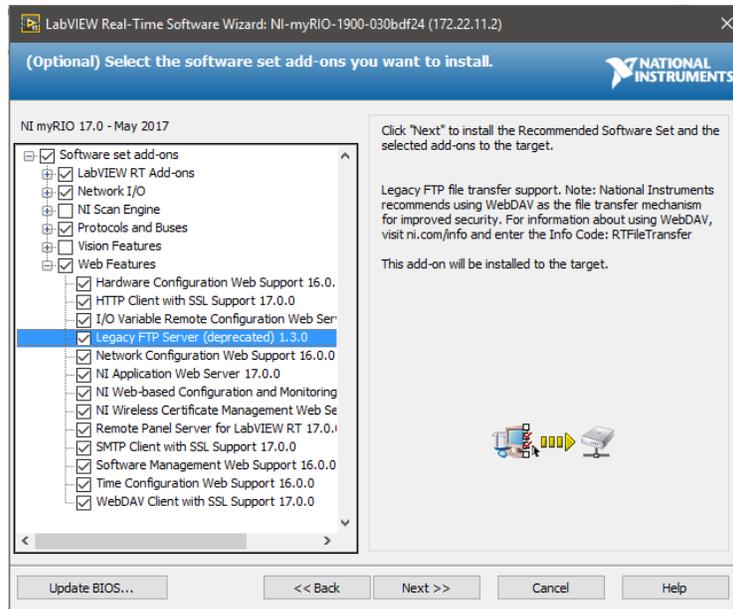


Figura 4.14 tercer paso para instalar el servidor FTP en la tarjeta

se presiona el botón <<Next>> y se abre una nueva ventana donde muestran todos los elementos seleccionados de la ventana anterior, como se muestra en la Figura 4.14, una vez verificado que el componente de software del servidor FTP aparezca seleccionado se presiona el botón <<Next>> para iniciar la instalación, en la Figura 4.15 se puede observar los elementos marcados que se están por instalar.

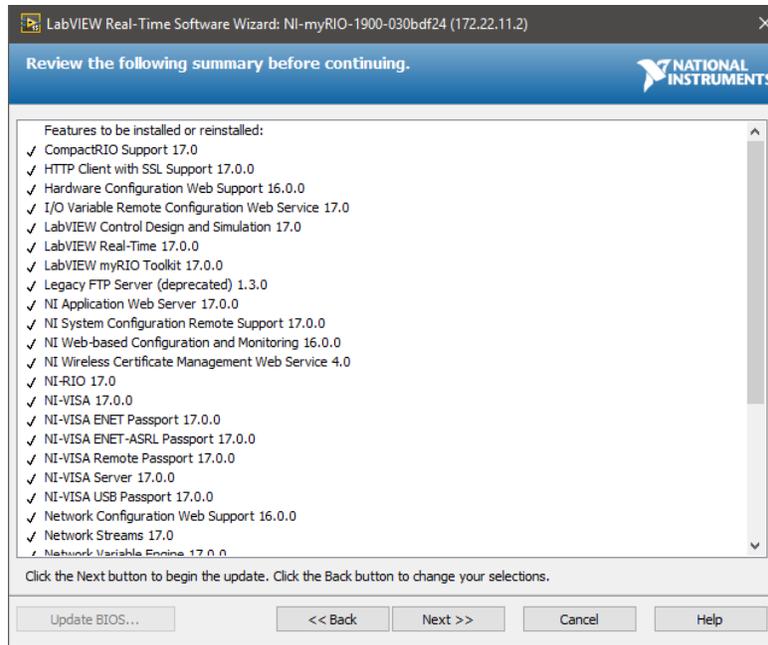


Figura 4.15 aplicaciones seleccionadas para instalar en la tarjeta myRIO

Al finalizar la instalación de los elementos de software seleccionados y el servidor FTP, se puede verificar que se haya instalado correctamente volviendo a entrar a la ventana de **Software set Add-ons** como se muestra en la Figura 4.14. Posteriormente se puede comprobar el funcionamiento correcto de la tarjeta, accediendo desde cualquier navegador y escribiendo en la barra de búsqueda la dirección **ftp://172.16.0.1/** , la cual se enlazará a los directorios de archivos que componen el sistema de la tarjeta myRIO. En la Figura 4.16 se muestra la conexión al servidor FTP desde el navegador WEB Fire Fox.

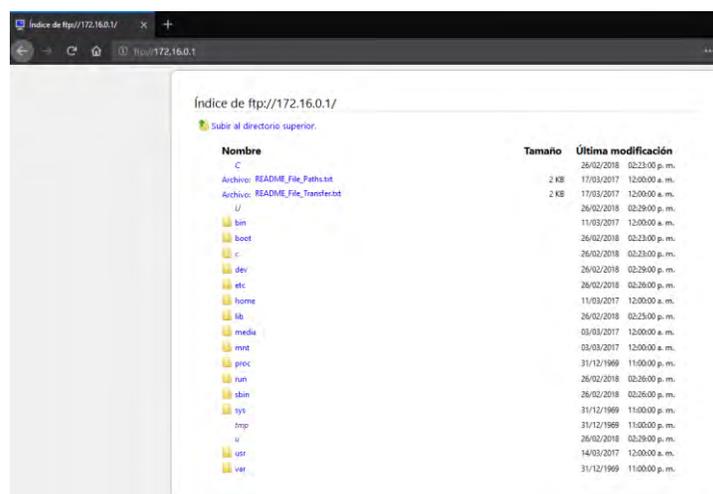


Figura 4.16 conexión establecida al servidor FTP instalado en la tarjeta myRIO-1900

4.3 programación del instrumento virtual a nivel FPGA de la tarjeta myRIO-1900

En el diagrama de bloques del I VI creado a nivel FPGA de la myRIO se crea un ciclo **While Loop** el cual es el encargado de realizar una acción hasta que una condición se cumpla. La condición es un botón tipo booleano nombrado **<<Stop>>**, este sirve para detener las funciones que se realizan dentro del ciclo **While Loop** y detener el programa.

Para hacer la lectura de las señales que se van a adquirir se habilitan los canales analógicos del puerto A de la tarjeta myRIO, en la rama **FPGA Target (RIO0, myRIO-1900)**, en la carpeta **“ConnectorA”>> “Analog”**, se agregan los canales **“Voltaje (ConnectorA/AI0)”** y **Corriente (ConnectorA/AI1)”** al Diagrama de Bloques, estos canales son puestos dentro del ciclo **While Loop**, los canales son representados como nodos en el diagrama de bloques. En la **Figura 4.17** Ciclo While Loop y entradas analógicas en el diagrama de bloques. **Figura 4.17** se muestran los canales de corriente y Voltaje agregados al diagrama de bloques.

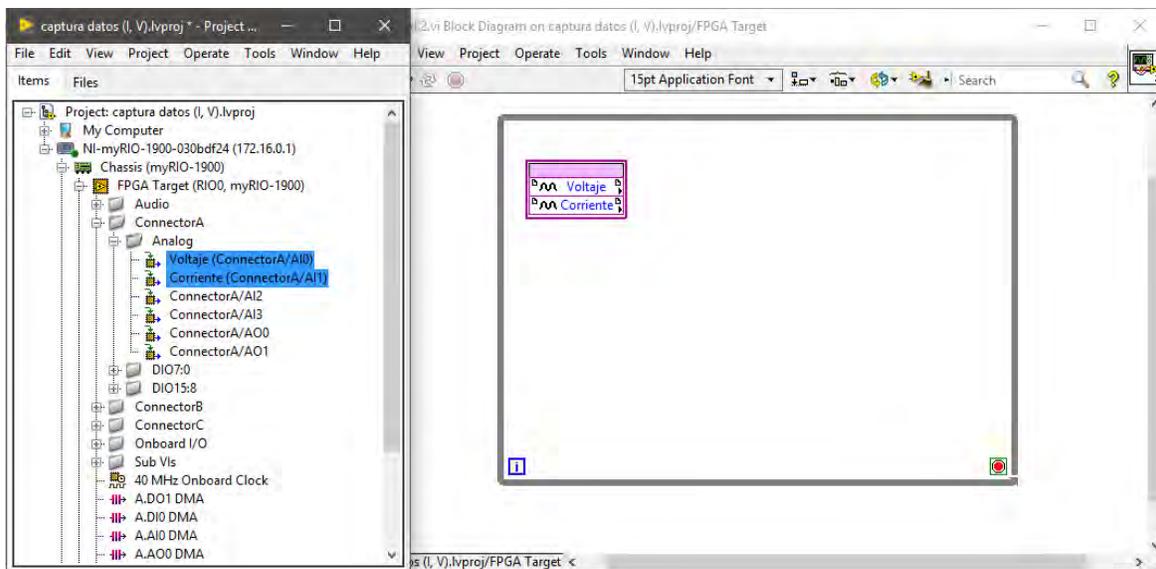


Figura 4.17 Ciclo While Loop y entradas analógicas en el diagrama de bloques

Para visualizar los datos de las señales se crearán indicadores numéricos en cada salida de los nodos de las entradas analógicas y nodos de operaciones, que llevarán como nombre **“V16B, I16B, V12B, I12B, Voltaje y Corriente2**, los indicadores **V16B** y **I16B** se conectarán a las entradas analógicas agregadas de **Voltaje** y **Corriente**.

Los indicadores creados solo servirán para visualizar los datos que están siendo adquiridos por la tarjeta; estos visualizadores son indicadores numéricos, del tipo entero de 16 bits y FXP de 12 bits.

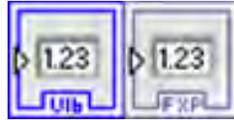


Figura 4.18 Indicadores numéricos tipo Entero y FXP

En las salidas de los conectores analógicos **Voltaje** y **Corriente** se conecta un nodo convertidor en este caso números tipo entero que se convertirán a números de punto fijo (FXP, por sus siglas en inglés, Fixed-Point), como se muestra en la Figura 4.19

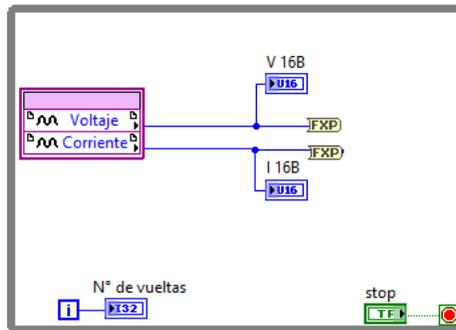


Figura 4.19 conexión del nodo conversor FXP e indicadores numéricos enteros de 16bits

La conversión a este tipo de números se hace para que el ADC pueda representar los valores en bits de los números FXP en números enteros con decimales. El Nodo convertidor y los indicadores numéricos FXP se configuran para que envíen datos de 12bits y se puedan aplicar las ecuaciones (3.1) y (3.2). En la Figura 4.20 configuración del nodo conversor y el indicador FXP numérico para representación de datos de 12 bits. Figura 4.20 muestra la configuración del nodo convertidor FXP y los indicadores numéricos FXP.

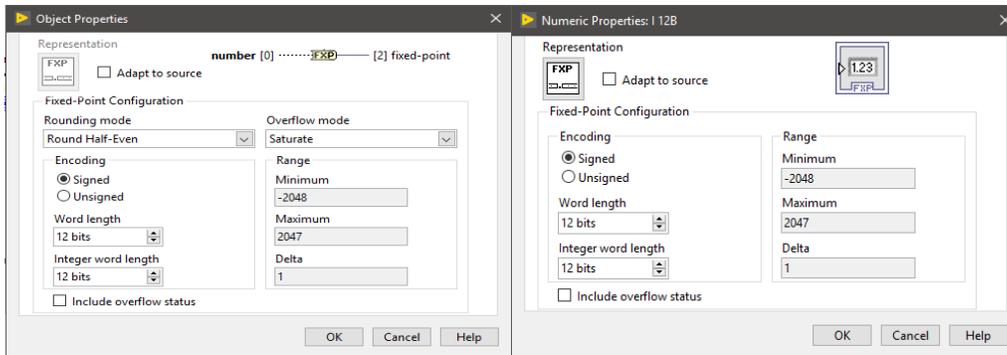


Figura 4.20 configuración del nodo conversor y el indicador FXP numérico para representación de datos de 12 bits

En las salidas de los nodos FXP se conecta el **Express VI “Mean, Variance, and Standard Deviation”**, este es configurado para que calcule el promedio de cincuenta muestras de las señales de voltaje y corriente, el **Express VI** arroja un solo valor por cada número de muestras que le se le configure. En la Figura 4.21 se muestra la configuración y ubicación del Express VI en la paleta de controles.

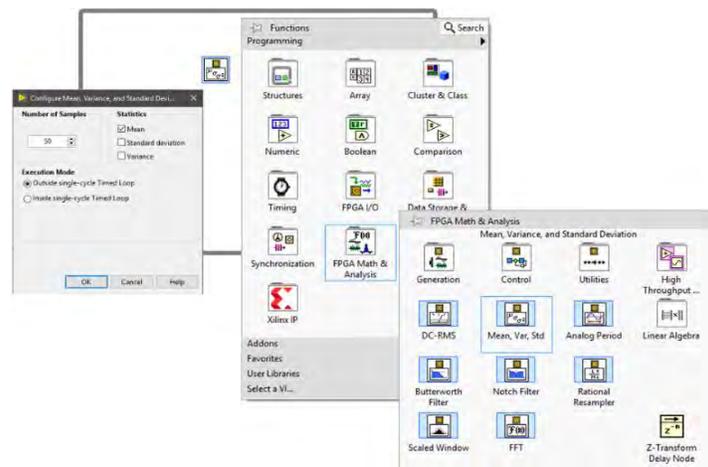


Figura 4.21 Ubicación y configuración del Express VI (Mean, Variance, and Standard Deviation).

En la Figura 4.22 se muestra el diseño del diagrama de bloques con los elementos que se han colocado hasta ahora.

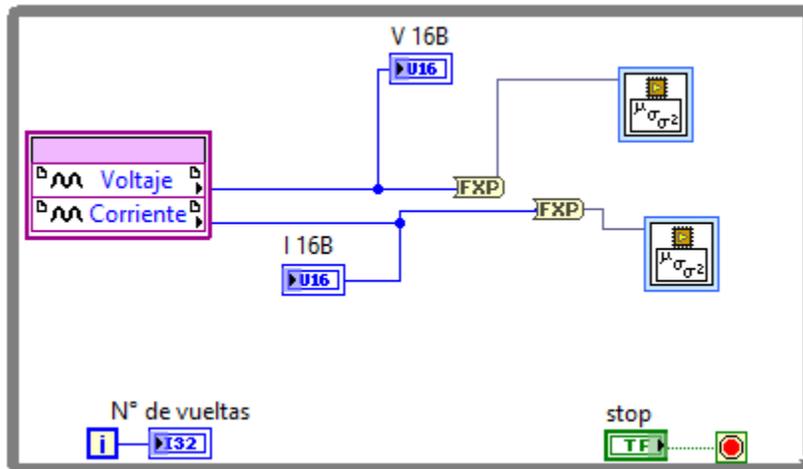


Figura 4.22 conexión del Express VI Express VI (Mean, Variance, and Standard Deviation) al nodo convertidor FXP

En las salidas de los **Express VI** se conecta un indicador con el que se observan los datos convertidos a números de 12 bits. Por otra parte, en la salida del **Express VI** se coloca el nodo **Multiply** que se utilizará para multiplicar los datos de voltaje y corriente por la constante de la ecuación (3.1)

Se agrega un indicador numérico a la salida del nodo **Multiply**, en el cual se van a observar los datos de la señal de Voltaje con su valor real en números decimales.

Pero al indicador numérico colocado en la salida del nodo **Multiply** para visualizar los datos de corriente, en él se visualizan los datos a una escala de mV, ya que la señal procesada por la tarjeta myRIO se está tomando a través de un sensor de corriente el cual tiene una resolución de $50mV / A$, por lo que hay que hacer un segundo nodo **Multiply** que hará una escala a la resolución del sensor para que los valores que se muestren puedan ser leídos fácilmente.

La multiplicación se hace entre los datos convertidos a números decimales y una constante de valor 20, que hace que la escala quede en $1V / 1A$, esto quiere decir, que los valores obtenidos van a ser visualizados en el rango de números enteros y decimales y no en mV, Por lo que indicador numérico es colocado en la salida del segundo nodo **Multiply** para poder visualizar los datos a una escala más grande, como se muestra en la Figura 4.23

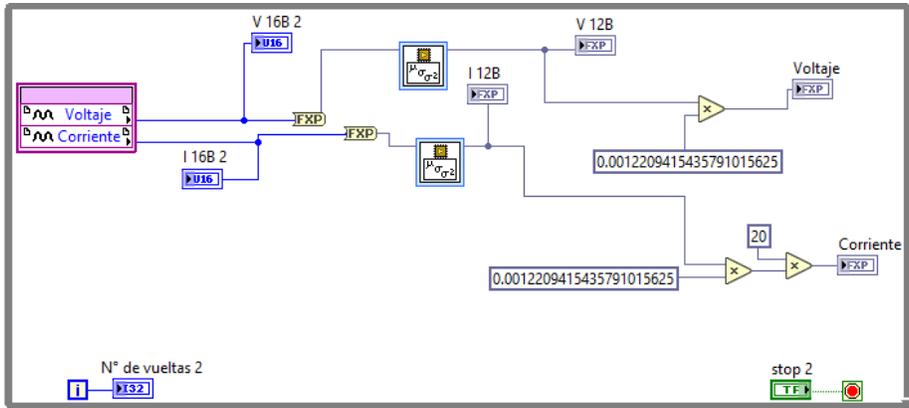


Figura 4.23 multiplicaciones para obtener el valor de corriente y voltaje en números decimales

Se agrega un último nodo **Multiply** para realizar la multiplicación entre los datos de Voltaje y Corriente. En una de las entradas del nodo **Multiply** se conecta el resultado de la multiplicación de voltaje, en la otra entrada se conecta el resultado de la multiplicación de corriente escalado, esta multiplicación se hace para obtener la potencia aplicada al electrolizador. En la Figura 4.24 se muestra el diagrama de bloques completo del programa diseñado para la adquisición de las señales de corriente, voltaje y la potencia aplicada al electrolizador tipo PEM.

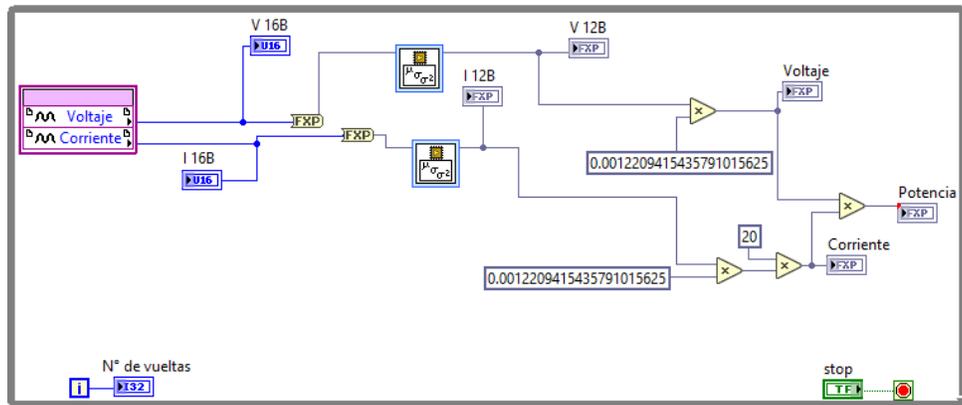


Figura 4.24 Diseño final del diagrama de bloques

4.3.1 Panel frontal del programa a nivel FPGA

El diseño del panel frontal consiste en un botón cuya función es detener el programa cuando se presione y de ocho indicadores donde cada uno muestra las variables de corriente y voltaje medidas con diferentes tipos de indicadores numéricos. Los indicadores llamados **Corriente**, **Voltaje** y **Potencia** son los indicadores más

relevantes del programa ya que allí se muestran los valores de Voltaje, corriente y potencia con números decimales.

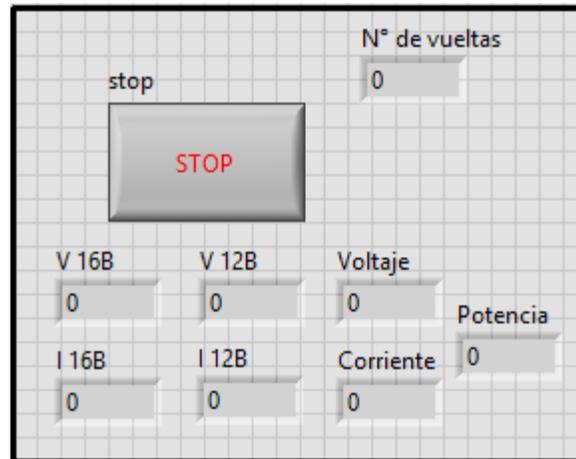


Figura 4.25 Diseño final del panel Frontal

4.4 Desarrollo del instrumento virtual a nivel procesador de la tarjeta myRIO-1900

El desarrollo de la programación del Instrumento virtual para la visualización, almacenamiento y creación de archivos de los datos de las señales enviadas por la FPGA está dividido en dos partes. La primera parte consiste en el desarrollo de la programación para la visualización de las señales enviadas por la FPGA en visualizadores tipo gráfica, medidores y visualizadores numéricos. Por otro lado, la segunda parte consiste en el desarrollo de la programación para el almacenamiento y creación de archivos de datos.

4.4.1 Programación para visualización de datos

Se crea un nuevo VI a nivel procesador y agregan los nodos **Open FPGA VI Reference**, **Read/Write Control**, **Close FPGA VI Reference** y un ciclo **Time Loop**. Al diagrama de bloques, en la Figura 4.26 se muestra los nodos y el ciclo **Time Loop** que se agregaron al diagrama de bloques en la paleta de controles.

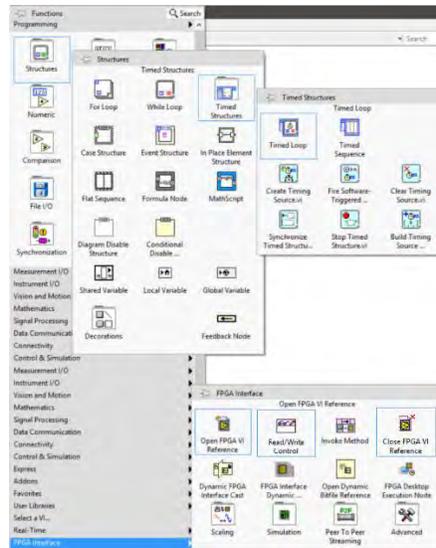


Figura 4.26 ciclo Time Loop y Nodos para abrir, leer y cerrar un VI referenciado de la FPGA

el nodo **Open FPGA VI Reference**, como su nombre lo indica hace referencia a un VI que está a nivel FPGA, esto quiere decir que abrirá y ejecutara un VI que está a nivel FPGA al igual enviara los datos que están siendo procesados en al FPGA hacia el procesador de la tarjeta myRIO, en este caso se hará la referencia al VI que se creó anteriormente a nivel FPGA, para enviar los datos de voltaje, corriente y potencia. Para hacer la referencia al VI que está a nivel FPGA se abre la configuración del nodo **Open FPGA VI Reference**, se marca la opción VI, se busca y se selecciona el VI al que se quiere referenciar, como se muestra en la Figura 4.27

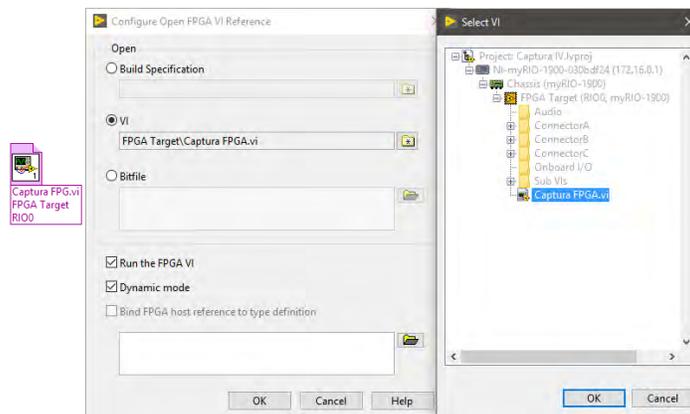


Figura 4.27 configuración del nodo Open FPGA VI Reference.

Dentro del ciclo **Time Loop** se inserta el nodo **Read/Write Control**. Este nodo va a recibir los datos de las señales de voltaje, corriente y potencia que se están enviando la FPGA por medio del nodo de **Open FPGA VI Reference** mientras que

el ciclo **Time Loop** servirá para realizar el monitoreo de dichos datos. El nodo **Close FPGA VI Reference** va colocado fuera del ciclo y servirá para cerrar y suspender la transferencia de datos enviados al nodo **Open FPGA VI Reference** cuando se oprima el botón **<<Stop>>** Finalmente se insertan los datos de voltaje, corriente y potencia que van a ser leídos en el nodo **Read/Write Control** para visualizarlos y se unen las salidas de los nodos siguiendo una estructura de abrir, leer y cerrar, como se muestra en la Figura 4.28.

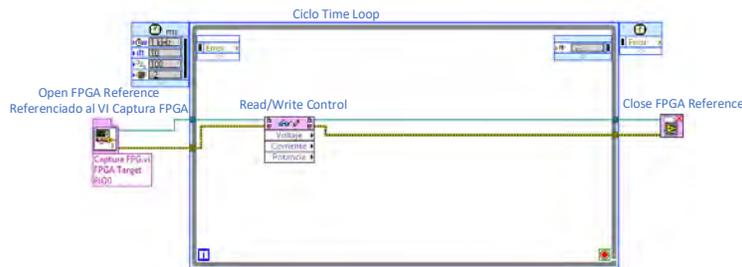


Figura 4.28 conexión de los nodos para abrir, leer y cerrar el VI referenciado

Después de armar la estructura de abrir, leer y cerrar el Vi referenciado, en las salidas del nodo **Read/Write Control** (voltaje, corriente y Potencia) se agrega un nodo convertidor de números de punto fijo a números de punto flotante con doble precisión (DBL), esta operación se hace para que en las gráficas que se insertan en el panel frontal se puedan visualizar los datos enviados de la FPGA.

En la Figura 4.29 se muestra la conexión de los nodos conversores a las salidas del bloque **Read/Write Control** (voltaje, corriente y Potencia).



Figura 4.29 conexión de los nodos conversores a las salidas del bloque **Read/Write Control**.

En las salidas de cada nodo convertidor se conecta un indicador numérico, una gráfica y un medidor, estos indicadores servirán para la visualización de los datos que están siendo enviados por la FPGA, los indicadores se agregan desde la paleta de controles del panel frontal. En la Figura 4.30 se muestra cada uno de los componentes mencionados desde la paleta de controles del panel frontal.



Figura 4.30 Indicadores numéricos y graficas agregadas desde el panel frontal

La Figura 4.31 muestra la conexión de los indicadores numéricos y graficas que se agregaron desde el panel frontal y el diseño final del algoritmo programado para la visualización de los datos enviados por la FPGA.

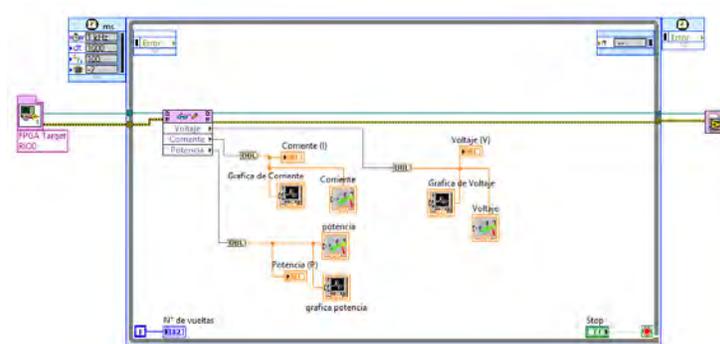


Figura 4.31 Algoritmo para la Visualización de datos enviados por la FPGA

4.4.2 Programación para creación de archivos y almacenamiento de datos

El algoritmo que se muestra en la Figura 4.35 crea archivos y almacena los datos de corriente, Voltaje y Potencia enviados por la tarjeta myRIO en un archivo con formato .CSV, los archivos creados se visualizan en programas como pueden ser un editor de texto o una hoja de cálculo. Los archivos son almacenados en una memoria USB conectada al puerto USB 2.0 de la tarjeta myRIO.

4.4.2.1 Desarrollo del algoritmo para la creación y almacenamiento de datos.

Al exterior del ciclo **Time Loop** se inserta un nodo **Build Array** con cuatro entradas, en las últimas entradas se conectan las salidas de los datos de Voltaje, Corriente y Potencia. Este arreglo sirve para que los datos enviados por la FPGA sean fáciles de manipular al momento de ser capturados en el archivo.

En el nodo de iteraciones que está dentro del ciclo **Time Loop** se conecta a un nodo convertidor **DBL** en la parte exterior del ciclo, los valores enviados por nodo iteraciones el cual se conectó al convertidor **DBL** se dividen por una constante de valor “100” por medio de un nodo **Divide**, la salida de este nodo **Divide** se conecta a la primera entrada del nodo **Build Array** agregado anteriormente. La división se hace para convertir el valor de las iteraciones del ciclo en segundos, basado en los diez milisegundos de captura configurados en el ciclo **Time Loop**.

La salida del nodo **Build Array** se conecta a la entrada “**2D data**” del **Express VI (Write Delimited Spreadsheet)**, este Express VI convierte los datos enviados por el nodo **Build Array** (tiempo de captura, corriente, Voltaje y potencia) a texto plano para hacer la escritura de ellos en el archivo con formato .CSV. En la Figura 4.32 se muestran los componentes utilizados por el algoritmo para creación de archivos en la paleta de controles.

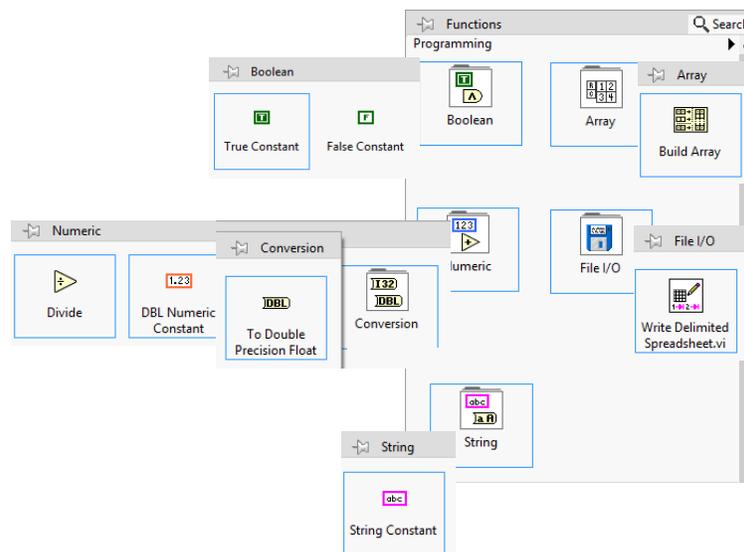


Figura 4.32 primera parte de componentes usados para el algoritmo de creación de archivos.

En la entrada “**Transpose?**” del **Express VI** que se muestra en la Figura 4.33 se conecta una constante del tipo booleana con valor “Verdadero” (T), el cual indica que los datos escritos en los archivos se escriban en cuatro columnas distintas. Por su parte en la entrada **delimiter** se conecta una constante de tipo string de carácter coma (,), esta sirve para agregar un salto de línea en cada columna de los datos capturados en el archivo. En la Figura 4.33 se muestra señalados en recuadros

rojos los componentes usados para el algoritmo de creación de archivos en el Diagrama de Bloques.

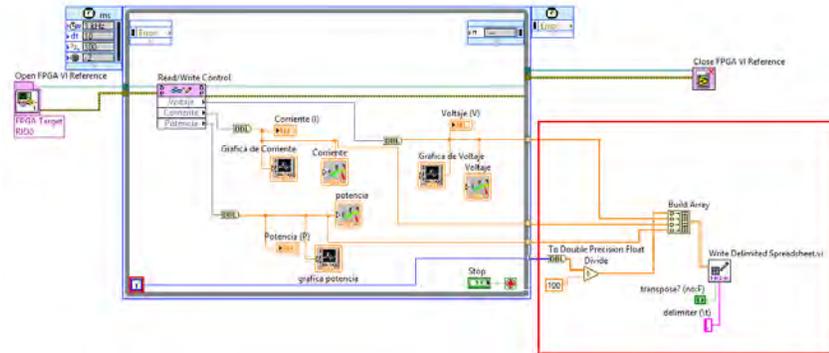


Figura 4.33 Primera parte del algoritmo para la creación de archivos en el Diagrama de Bloques

En la parte exterior del ciclo Time Loop se agrega un nodo **Build Path**, el cual sirve para crear y mandar un archivo a un directorio específico para su almacenamiento, entre la conexión de la salida **Appended Path** del nodo **Build Path** a la entrada **file Path (dialog if empty)** del **Express VI (Write Delimited Spreadsheet)**, se crea un indicador el cual sirve para visualizar la ruta donde será guardado el archivo.

Para la salida **Name o Relative Path** del nodo **Build Path** se creó un nodo **concatenate string** el cual tiene dos entradas. La primera entrada se conecta a un indicador tipo string, el cual sirve para crear el nombre del archivo que se va a almacenar, en la segunda entrada se crea una constante del tipo string en esta se escribe el formato del archivo “.CSV”, en la Figura 4.34 se muestra marcado en un cuadro rojo los componentes y conexiones del nodo **Build Path** para el algoritmo de creación de archivos.

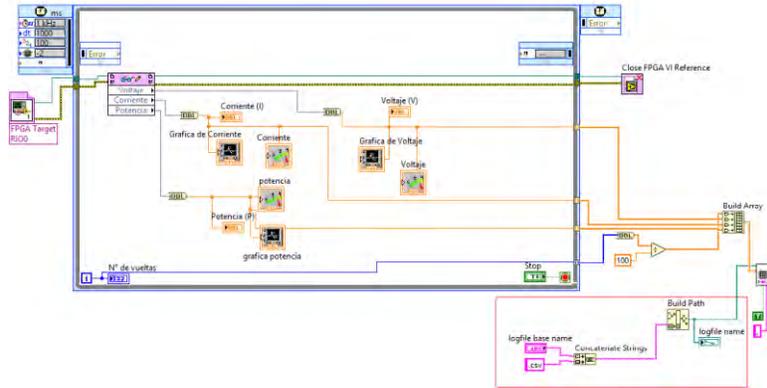


Figura 4.34 Segunda parte del algoritmo para la creación de archivos

Para hacer el almacenamiento de los archivos creados, en la entrada **base Path** del nodo **Build Path**, se conecta una constante **file Path**, esta lleva como nombre la unidad de almacenamiento o ruta donde se va a guardar el archivo, en este caso se nombró “/u” ya que es el nombre que le da por default la tarjeta myRIO a las memorias USB conectadas en su puerto. El conector donde se creó la constante **file Path** se conecta el nodo **List folder** y en la entrada llamada **file names** del nodo **List folder** se conecta un indicador, en el visualizan el nombre y formato del archivo creado, en la entrada **folder names** del mismo nodo se conecta una constante, esta indica el directorio donde está contenido el archivo en la memoria USB, en la Figura 4.35 se muestra la conexión de los últimos componentes mencionados y el diseño final del diagrama de bloques.

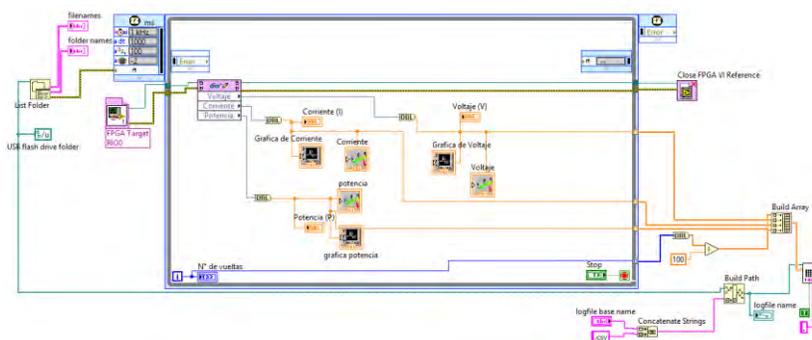


Figura 4.35 Diseño final del diagrama de bloques

4.4.3 panel frontal del programa a nivel procesador.

En el panel frontal se muestran los indicadores que se agregaron anteriormente, como se muestran en la Figura 4.30. En los indicadores se pueden visualizar los

datos enviados de las señales de voltaje, corriente y potencia. De igual forma se podrá cambiar el nombre del archivo creado, esto quiere decir que se podrá almacenar múltiples archivos con diferentes muestras. En la Figura 4.36 se muestra el diseño final del panel frontal.

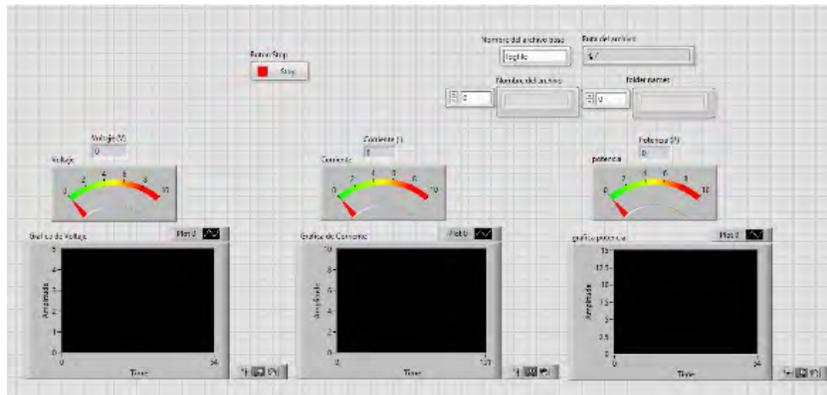


Figura 4.36 Diseño final del panel frontal

4.5 Simulación de señales a medir.

Antes de conectar el sensor de corriente y la tarjeta myRIO se hizo la simulación de las señales a medir para comprobar que el dispositivo de adquisición de datos este registrando correctamente las señales enviadas de voltaje y corriente.

La señal de corriente se simuló conectando el sensor a un arreglo de resistencias de potencia conectadas en serie a una fuente variable de voltaje. Para la señal de voltaje se conectó una de las salidas de 5V del conector A de la tarjeta myRIO.

En cuanto a los resultados de las señales simuladas medidas, la tarjeta capturó correctamente los datos de las señales con los parámetros necesarios. En la Figura 4.37 se muestra el prototipo de la simulación de las señales, así como los resultados adquiridos.

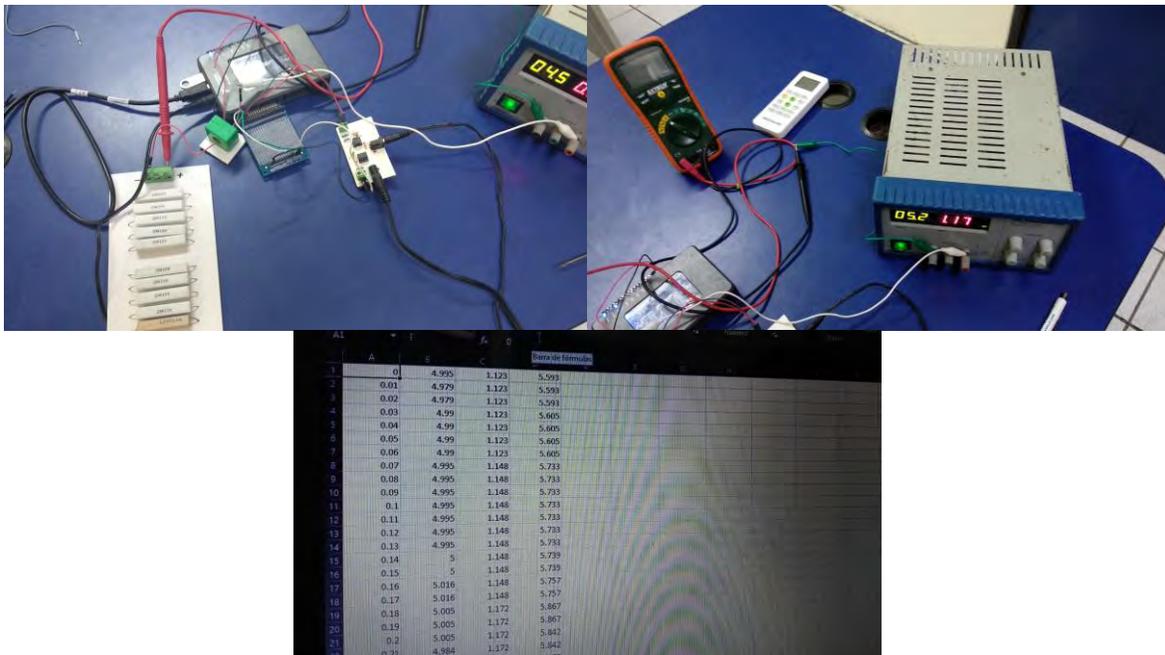


Figura 4.37 simulación para la adquisición de datos de las señales de corriente y voltaje, y resultados adquiridos.

Capítulo 5 Resultados experimentales.

Para caracterizar el comportamiento de una de las celdas internas de 100W tipo PEM del electrolizador, se capturaron y almacenaron en distintos archivos muestras de la parte del comportamiento de la potencia eléctrica aplicada al electrolizador para alcanzar distintos puntos de operación. Los datos de medición de tiempo, voltaje, corriente y potencia almacenados se pueden visualizar en un software de hoja de cálculo. En la hoja de cálculo se visualizan los datos mencionados en una serie de columnas donde en la primera columna se encuentran el tiempo de medición, en la segunda el voltaje, en la tercera corriente y en la cuarta a potencia eléctrica aplicada al electrolizador.

Los datos registrados se graficaron para caracterizar el comportamiento de corriente de la celda, voltaje de la celda y potencia eléctrica con respecto tiempo, hasta alcanzar un punto de operación del electrolizador.

Se realizaron mediciones experimentales a intervalos de 15 minutos por cada punto de operación programado en el electrolizador. Los puntos de operación se programaron para que el electrolizador operará a presiones de 2, 4 y 6 Bares. Los archivos con los datos capturados para distintos puntos de presión tienen un tamaño promedio de 3260 kB con 90000 datos aproximado cada uno.

Para tener una gráfica representativa del comportamiento de los datos adquiridos de voltaje, corriente y potencia eléctrica, se tomaron 200 datos de los 90000 capturados, que representan el tiempo que tarda el electrolizador para llegar al punto de operación programado para la producción de hidrogeno.

La caracterización de la celda tipo PEM se utilizó también para verificar el funcionamiento de los sensores de medición de corriente y voltaje que están integrados de fábrica en el electrolizador. La medición de las variables eléctricas se realizó con la tarjeta NI-myRIO y con la ayuda de un osciloscopio y una sonda de medición de corriente se verificó la medición de los datos adquiridos.

5.1 Medición de voltaje

En las siguientes imágenes se representan los cambios de voltaje con respecto al tiempo de la celda tipo PEM de 100W hasta alcanzar los puntos de operación establecido en 2, 4 y 6 Bares. El eje X corresponde a los datos medidos en 15 minutos y el eje Y representa una fracción de los datos de voltaje de la celda medidos en Volts (V). La Figura 5.1 muestra el comportamiento del voltaje aplicado al electrolizador, donde se observa que el voltaje disminuye conforme se alcanza la presión de 2 bares establecida como punto de operación para el hidrógeno generado.

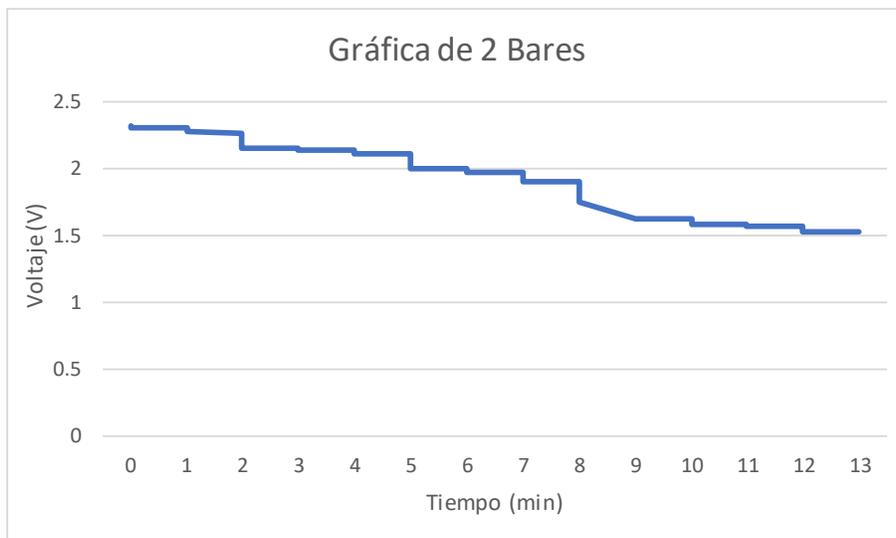


Figura 5.1 gráfica de voltaje para el punto de control de 2 bares asignado al electrolizador.

La Figura 5.2 muestra el comportamiento del voltaje aplicado al electrolizador, donde se observa que el voltaje disminuye conforme se alcanza la presión de 4 Bares establecida como punto de operación para el hidrógeno generado.

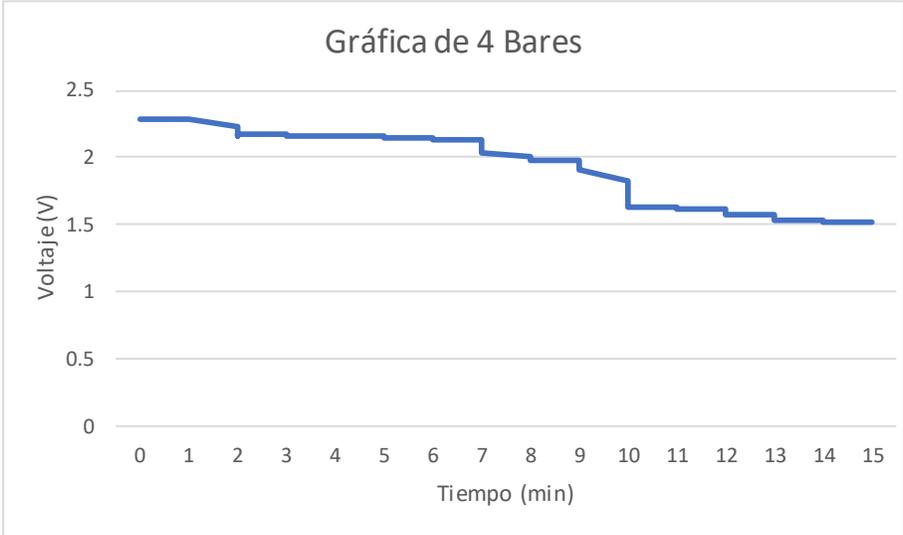


Figura 5.2 gráfica de voltaje para el punto de control de 4 bares asignado al electrolizador.

La Figura 5.3 muestra el comportamiento del voltaje aplicado al electrolizador, donde se observa que el voltaje disminuye conforme se alcanza la presión de 6 Bares establecida como punto de operación para el hidrógeno generado.

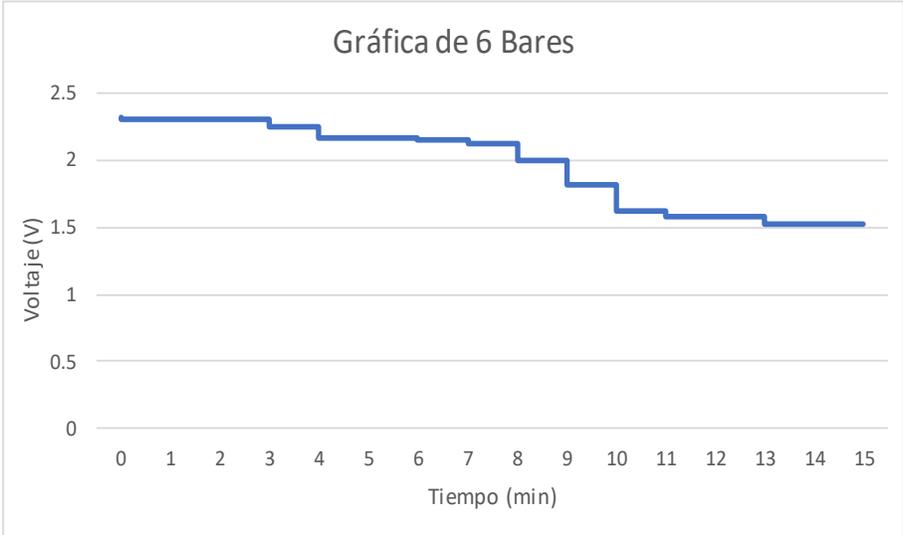


Figura 5.3 gráfica de voltaje para el punto de control de 2 bares asignado al electrolizador.

5.2 Medición de Corriente

En las siguientes imágenes se representan los cambios de corriente con respecto al tiempo de la celda tipo PEM de 100W hasta alcanzar los puntos de operación establecido en 2, 4 y 6 Bares. El eje X corresponde a los datos medidos en 15 minutos y el eje Y representa una fracción de los datos de corriente de la celda medidos en Amperes (A). La Figura 5.4 muestra el comportamiento de la corriente aplicada al electrolizador, donde se observa que la corriente disminuye conforme se alcanza la presión de 2 Bares establecida como punto de operación para el hidrógeno generado.

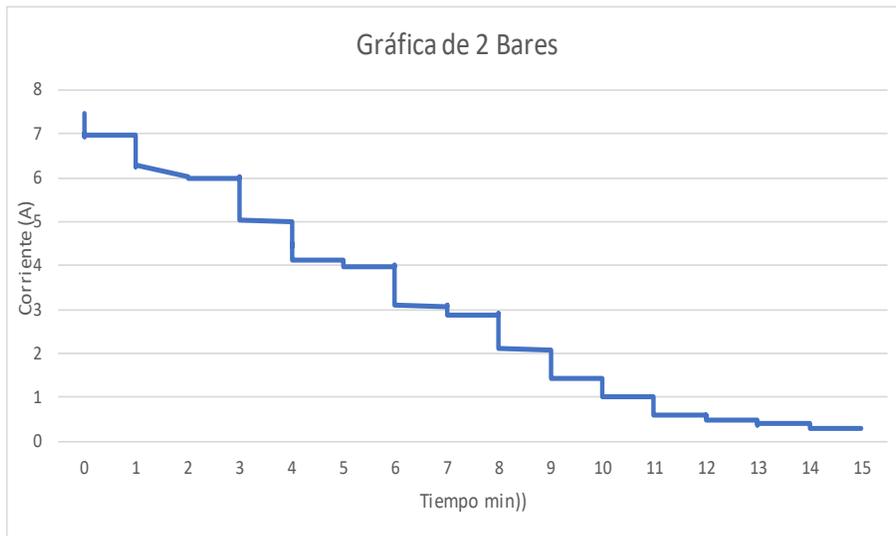


Figura 5.4 gráfica de corriente para el punto de control de 2 bares asignado al electrolizador.

La Figura 5.5 muestra el comportamiento de la corriente aplicada al electrolizador, donde se observa que la corriente disminuye conforme se alcanza la presión de 2 Bares establecida como punto de operación para el hidrógeno generado.

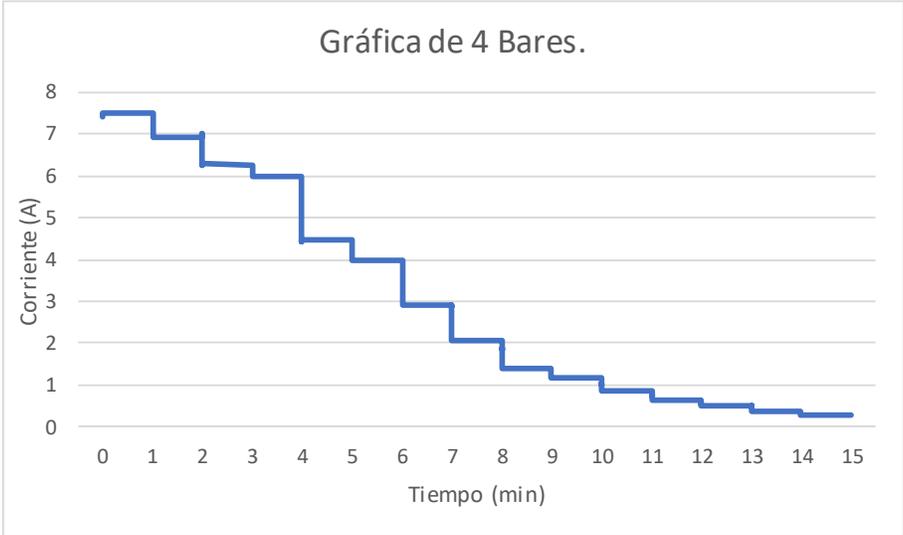


Figura 5.5 gráfica de corriente para el punto de control de 4 bares asignado al electrolizador.

La Figura 5.6 muestra el comportamiento de la corriente aplicada al electrolizador, donde se observa que la corriente disminuye conforme se alcanza la presión de 2 Bares establecida como punto de operación para el hidrógeno generado.



Figura 5.6 gráfica de corriente para el punto de control de 6 bares asignado al electrolizador.

5.3 Medición de potencia eléctrica

En las siguientes imágenes se representan los cambios de potencia eléctrica con respecto al tiempo de la celda tipo PEM de 100W hasta alcanzar los puntos de operación establecido en 2, 4 y 6 Bares. El eje X corresponde a los datos medidos en 15 minutos y el eje Y representa una fracción de los datos de potencia eléctrica de la celda medidos en Watts (W). La Figura 5.7 muestra el comportamiento de la potencia eléctrica aplicada al electrolizador, donde se observa que la potencia eléctrica disminuye conforme se alcanza la presión de 2 Bares establecida como punto de operación para el hidrógeno generado.

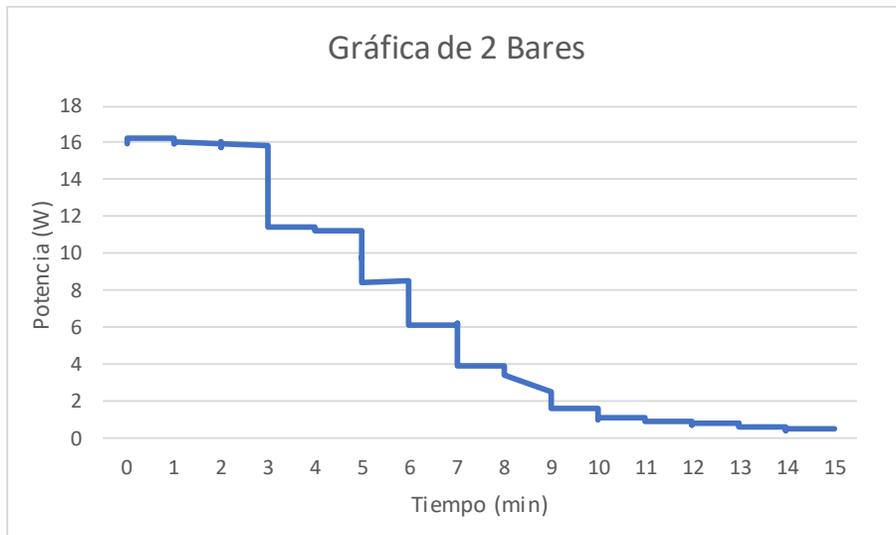


Figura 5.7 gráfica de potencia para el punto de control de 2 bares asignado al electrolizador.

La Figura 5.8 muestra el comportamiento de la potencia eléctrica aplicada al electrolizador, donde se observa que la potencia eléctrica disminuye conforme se alcanza la presión de 2 Bares establecida como punto de operación para el hidrógeno generado.

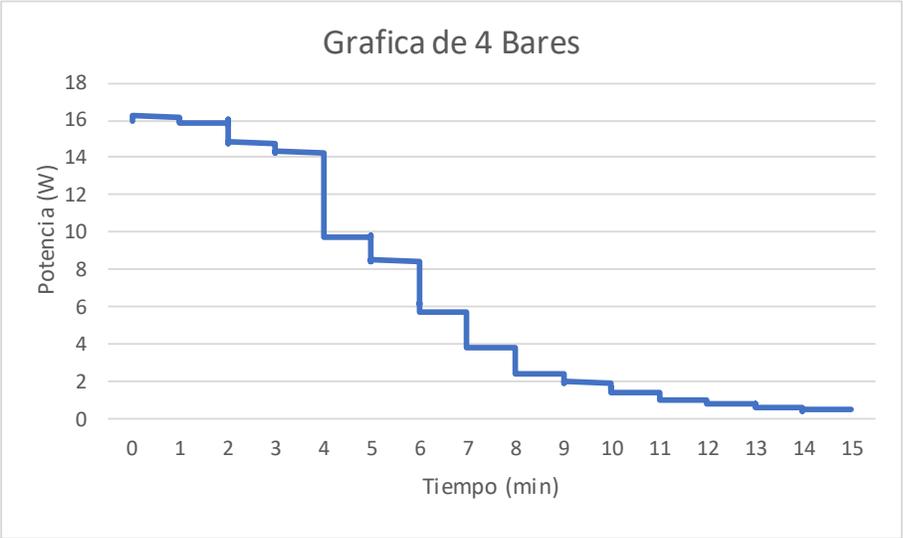


Figura 5.8 gráfica de potencia para el punto de control de 4 bares asignado al electrolizador.

La Figura 5.9 muestra el comportamiento de la potencia eléctrica aplicada al electrolizador, donde se observa que la potencia eléctrica disminuye conforme se alcanza la presión de 2 Bares establecida como punto de operación para el hidrógeno generado.



Figura 5.9 gráfica de potencia para el punto de control de 6 bares asignado al electrolizador.

Conclusiones

El sistema creado para la adquisición de datos para un electrolizador tipo PEM cumple satisfactoriamente lo planteado.

Se utilizó la técnica de instrumentación virtual para la programación de la tarjeta NI myRIO-1900, se creó un sistema de acorde a las necesidades planteadas en un principio. El diseño del instrumento virtual se construyó en el software LabVIEW el cual puede capturar, visualizar y almacenar datos procesados por la tarjeta myRIO

Se ha configurado la tarjeta de adquisición de datos para que trabaje como punto de acceso WiFi, por lo que se pueden enviar y recibir datos de la tarjeta hacia otros dispositivos o viceversa, también se puede hacer la programación de la tarjeta remotamente, al igual se instaló un servidor FTP para tener acceso a los archivos de las mediciones echas por la tarjeta myRIO.

Con los datos de voltaje y corriente obtenidos de las mediciones se puede determinar la energía consumida por el electrolizador en cualquier instante de tiempo. También los datos medidos son registrados simultáneamente en tiempo real, lo que permite determinar de manera efectiva el comportamiento del consumo de energía en el electrolizador

con los datos de potencia medidos se puede determinar la potencia instantánea que requiere el electrolizador para su operación, lo cual es fundamental en el caso de que se quiera emplear como medio de almacenamiento de energía con sistemas de energía renovable como son las celdas fotovoltaicas o generadores eólicos.

Al contar con el servidor FTP instalado en la tarje myRIO se tiene acceso de forma remota vía WiFi a los archivos con la información capturada de voltaje, corriente y potencia de la celda PEM por lo que facilita el acceso a la información para la caracterización del electrolizador.

Trabajos Futuros

Una de las características que me hubiera gustado implementar para mejorar el instrumento virtual es el poder habilitar el diseño del panel frontal para poderlo visualizar en un servidor Web y así poder hacer el sistema desarrollado completamente remoto para que cualquier usuario pueda acceder a monitorear los datos que están siendo procesados por la tarjeta, esto se puede mediante un enlace Web al panel frontal. Otra característica que se puede implementar para trabajos futuros complementar la instrumentación virtual realizada para hacer mediciones de presión.

Bibliografía.

- [1] J. I. S. Maldonado and I. A. V. Vergara, "Prácticas De Instrumentación Virtual Para El Laboratorio De Medición E Instrumentación," Universidad Nacional Autónoma De México, 2010.
- [2] L. A. S. Ramírez, "Desarrollo De Un Sistema De Monitoreo En Tiempo Real Para Un Electrolizador Oxihidrogeno.," Instituto Tecnológico De Chetumal, 2016.
- [3] J. C. R. Lara, "Implementación De Un Sistema Embebido De Algoritmos De Seguimiento De Máximo Punto De Potencia Para Paneles Fotovoltaicos," Universidad De Quintana Roo, 2017.
- [4] I. S. Engineering, "Ingeniería De Los Sistemas Embebidos," 2011.
- [5] Mendoza Santos Edwin Fernando, "Estudio Comparativo De Acero Inoxidable 304 Y 316 L Para Ser Utilizados Como Material De Electrodo En Electrolizadores Alcalinos," Universidad De Ciencias Y Artes De Chiapas, 2016.
- [6] N. I. Corporation, *Labview Core 1 Manual De Curso*. 2011.
- [7] Masterhacks, *Manual Básico De Programación En Labview*. 2013.
- [8] G. E. Acosta Quishpe And C. R. Salgado Rojas, "Diseño Y Construcción De Un Electrolizador De Corriente Continua De 200 Watios Con Energía Solar E Instrumentación Necesaria Para Medir La Producción De Hidrógeno," Escuela Politécnica Nacional, 2010.
- [9] Y. Miguel And B. Cepeda, "Sistema De Supervisión Variables Físico-Químicas De Un Sistema De Producción Energía Basado En Un Arreglo De Celdas De Combustible PEM," Universidad Distrital Francisco José De Caldas, 2017.
- [10] T. L. Eugenio, "Electrólisis PEM De Baja Temperatura Y Sus Aplicaciones Industriales," Universidad De Sevilla, 2015.
- [11] J. A. S. Guañuna, "Modelado, Simulación Y Optimización De La Producción De Hidrógeno En Un Electrolizador A Partir De Vapor Sobrecalentado De Agua Mediante La Herramienta LabVIEW De National Instruments," Escuela Politécnica Nacional, 2016.
- [12] J. C. Sánchez, *Tutorial LabVIEW*. 2011.

- [13] D. R. A. D. Ña, M. Del, And C. A. C. Lemente, *El Hidrógeno Como Almacén Energético. Aplicación De La Pila De Combustible Reversible Polimérica*. 2010.
- [14] P. M. G. Puente, “Desempeño Electroquímico De Partículas Ru-Ir-Co-O Soportadas En $\text{SnO}_2\text{-Sb}$ Para La Reacción De Evolución De Oxígeno En Un Electrolizador Tipo PEM,” Instituto Politécnico Nacional, 2018.
- [15] H. Roncancio And H. Cifuentes, *Tutorial De LabVIEW*, Vol. 1. 2000.
- [16] A. Delgado Martín, “Refrigeración De Alternadores Con Hidrógeno Producido Mediante Electrolizadores Tipo PEM.,” *An. Mecánica Y Electr.*, Vol. 82, No. 2, Pp. 17–22, 2005.
- [17] P. R. C. Director, “Análisis De Sistemas De Producción Y Utilización De Hidrógeno. Aplicación A Power To Gas.,” Universidad De Zaragoza, 2015.
- [18] D. P. Calleja, “Creación De Un Sistema De Control En LabVIEW Para Un Banco De Ensayos,” Universidad Carlos III De Madrid, 2010.
- [19] M. A. Niño Galeano, “Estudio De Generación De Energía Con Celdas De Combustible Con Membrana Poly Diallyldimethylammonium Chloride (Pdadm) Como Polímero De Intercambio Protónico,” Universidad De Puerto Rico, 2010.
- [20] L. J. O. Serrano And D. A. G. Páez, “Diseño E Implementación De Un Convertidor Utilizando Celdas De Hidrógeno Como Fuente De Energía,” Pontificia Universidad Javeriana, 2014.
- [21] V. G. Carrascal, “Gestión De Energía Eléctrica Con Controladores En Tiempo Real,” Universidad De Valladolid, 2016.
- [22] Iñigo J. Oleagordia Aguirre, “Empleo Del Ordenador Personal En La Instrumentación De Panel.”
- [23] W. A. Gálvez, “Detección Y Control Del Estado De Una Pila PEM Para Funcionamiento Óptimo: Arquitectura De Agentes De Percepción Y Control,” Universidad Carlos III De Madrid, 2013.
- [24] J. A. Asensio, J. Peña, D. Pérez-Coll, J. C. Ruiz-, D. Marrero-López, P. Núñez, B. Ballesteros, S. Borrós, C. De Investigación, C. I. N. Csic-Icn, C. Uab, And B. Barcelona, “Pilas De Combustible De Membrana Polimérica,” *Afinidad Lxviii*, Vol. 554, Pp. 246–258, 2011.

- [25] L. A. M. Ruiz, "Síntesis De Películas De Nanotubos De Carbono Para Producción De Hidrógeno Por Electrolisis Y Emisión De Campo," Instituto Politécnico Nacional, 2009.
- [26] E. Chihuahua, Programación Básica En LabVIEW. 2004.
- [27] J. D. Velásquez And J. F. Quiceno, "Diseño De Un Sistema De Generación De Hidrógeno Por Electrólisis," Universidad Tecnológica De Pereira, 2013.
- [28] C. Segura And M. Bernal, "LabVIEW Y La Instrumentación Virtual Aplicados A La Docencia Y La Investigación En Ciencias Básicas," *Elementos*, 2013.
- [29] J. García Méndez, "Diseño Y Construcción De Un Electrolizador De Agua Para Obtención De Oxihidrógeno Como Combustible," Instituto Politécnico Nacional, 2013.
- [30] N. Instruments, "User Guide and Specifications Ni Myrio-1900".

Anexos

creación de un proyecto nuevo en LabVIEW

Para crear un nuevo proyecto primero se debe iniciar el programa LabVIEW. Una vez iniciado el programa, se despliega la ventana de carga como la que se mostró en la Figura 3.10. Después que el programa se haya cargado completamente se despliega la ventana de inicio de LabVIEW. En esa ventana se hace clic en la opción “**File**” que se encuentra en la esquina superior izquierda y se despliega un cuadro de dialogo donde se selecciona la opción **Create Project**, como se muestra en la siguiente figura.

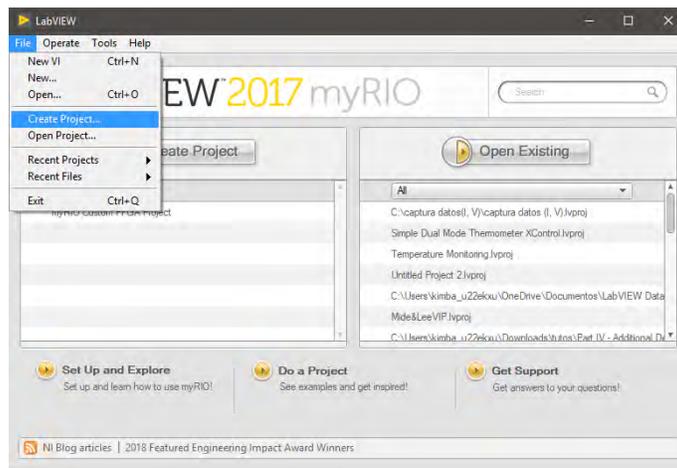


Figura de anexo 1 Ventana de inicio de LabVIEW

Al dar clic en la opción **Create Project** se abre una venta donde se seleccionan las opciones **myRIO>>myRIO Custom FPGA Project**, después se presiona el botón **<<Next>**. En la ventana que aparece después, se configura el nombre del proyecto, el directorio donde guarda él proyecto y el tipo conexión que usa para la transferencia de datos y programación de la tarjeta myRIO.

En el apartado **Target** de la ventana de configuración se marca la opción **Connected Over WiFi** ya que la tarjeta myRIO se configuro para trabajar como Punto de Acceso para hacer la programación y transferencia de datos por medio de WiFi, en el campo **Project Name** se pone el nombre del proyecto (**Captura VI**) y en el campo **Project Root** se selecciona el directorio donde se guardara el proyecto

creado, en este caso se guarda en la raíz del disco duro en la carpeta **(Captura IV)** que lleva el mismo nombre que el proyecto, es más eficiente guardar los archivos de un proyecto en la raíz del disco duro primario de una PC ya que a la hora de compilar el programa la búsqueda de archivos necesarios para llevar a cabo el proceso de compilación es más rápida por lo que el proceso tarda menos, una vez que se haya llenado los campos y opciones requeridos para la creación del proyecto se hace clic en el botón **<<Finish>>**. En la siguiente figura se muestra la configuración de los campos y opciones requeridas para la creación del proyecto.

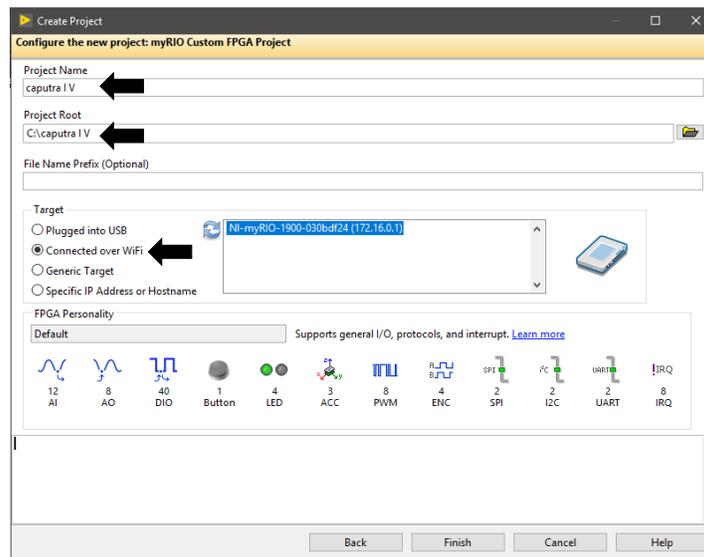


Figura de anexo 2 Configuración y opciones requeridas para la creación de un proyecto en LabVIEW

La ventana que se despliega después de presionar el botón **<<Finish>>** es la ventana del proyecto creado, en la ventana se aprecia cómo están organizados los componentes de la tarjeta myRIO-1900, en la sección de **Chassis (myRIO-1900)** y de **FPGA Target (RIO0, myRIO-1900)**, algunos de los elementos que contiene esta ventana son: los conectores A, B y C con sus respectivas entradas y salidas analógicas y digitales, el conector de audio, acelerómetro, LEDs, etc. En la siguiente figura se muestran los elementos que componen la tarjeta en la ventana del proyecto.

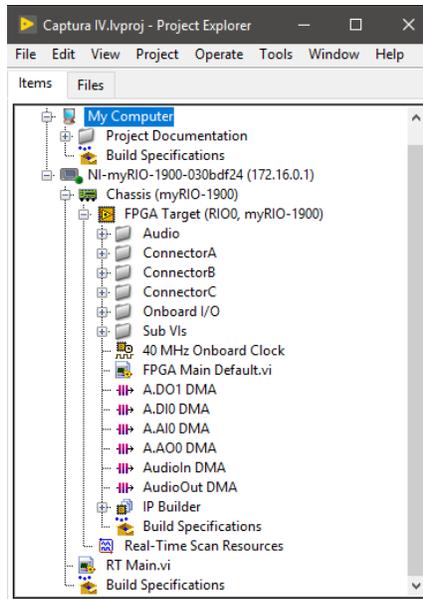


Figura de anexo 3 Componentes del proyecto creado

Una vez creado el proyecto se crea un nuevo VI a nivel FPGA de la myRIO, el nuevo VI se crea dando clic derecho sobre **FPGA Target (RIO0, myRIO-1900)** seleccionando **NEW>>VI**, como se muestra en la siguiente imagen.

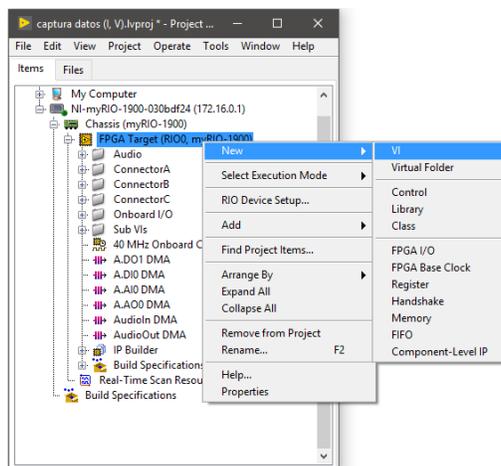


Figura de anexo 4 Creación de un VI en la FPGA de la tarjeta myRIO-1900

Después de haber creado el VI, se renombran las entradas analógicas “AI0 y AI1” del “conector A”. se extienden las opciones de la carpeta **ConnectorA>>Analog** en esta están contenidas las entradas analógicas del conector A, las cuales están referenciadas a un pin físico de la tarjeta myRIO, para renombrar las entradas analógicas se hace doble clic sobre **ConnectorA/AI0** y se abre una ventana en el campo de texto **Name**, se borra lo que contiene y se reemplaza con el texto **Voltaje**,

para el **ConnectorA/AI1** igual se renombra el campo de texto lo escribiendo “**Corriente**”, como se muestra en la siguiente figura.

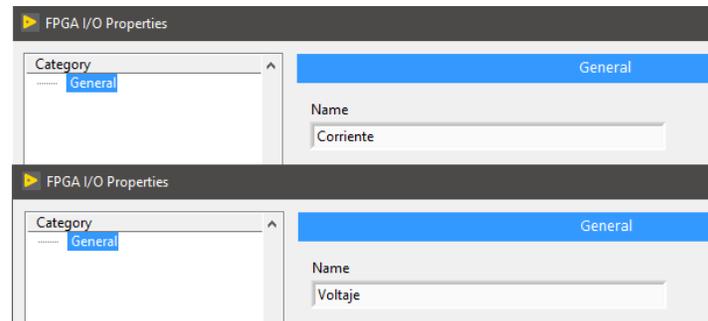


Figura de anexo 5 cambio de nombre a las entradas analógicas AI0 y AI1 de la tarjeta NI myRIO-1900

El cambio de nombre de los pines se hace para distinguir al momento de ser agregados al diagrama de bloques y empezar a programar con que pines del conector A está trabajar y que señal analógica va a ser procesada por los pines habilitados del conector, en la siguiente figura se muestra las entradas analógicas AI0 y AI1 renombradas en el proyecto.

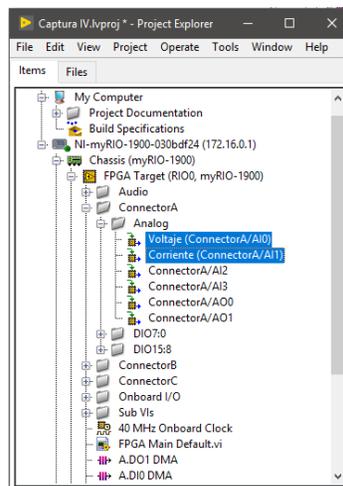


Figura de anexo 6 Entradas analógicas AI0 y AI1 renombradas en el proyecto

Cuando la tarjeta y la PC tengan establecida la conexión WiFi como se mostró en la Figura 4.9 es necesario hacer la conexión de la tarjeta con el programa LabVIEW para poder hacer el diseño del panel frontal y el diagrama de bloques, la conexión se establece haciendo clic derecho en el dispositivo de la tarjeta agregado al proyecto (**NI-myRIO-1900-030bdf24**) donde se despliega un cuadro de opciones y

se presiona la opción **Connect**. En la siguiente figura se muestra la diferencia entre cuando la tarjeta está conectada y cuando no al software LabVIEW.

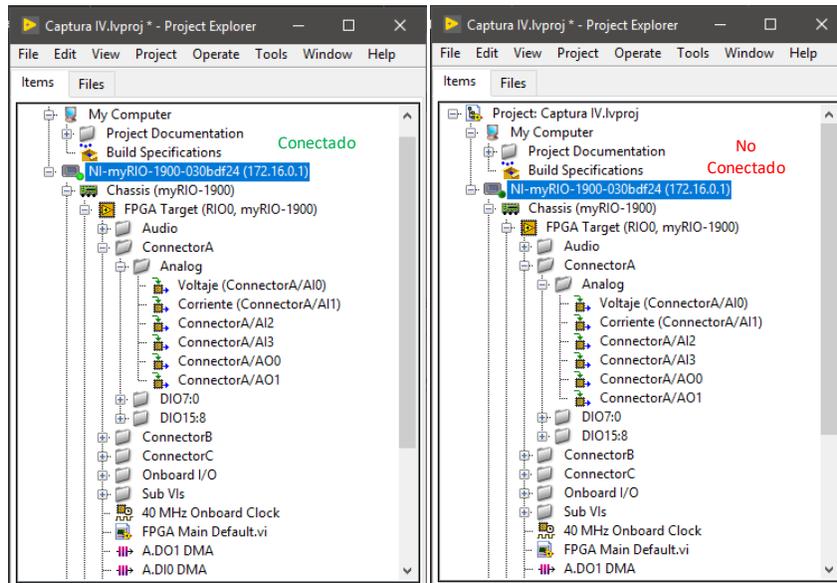


Figura de anexo 7 conexión entre la tarjeta myRIO y el Software LabVIEW

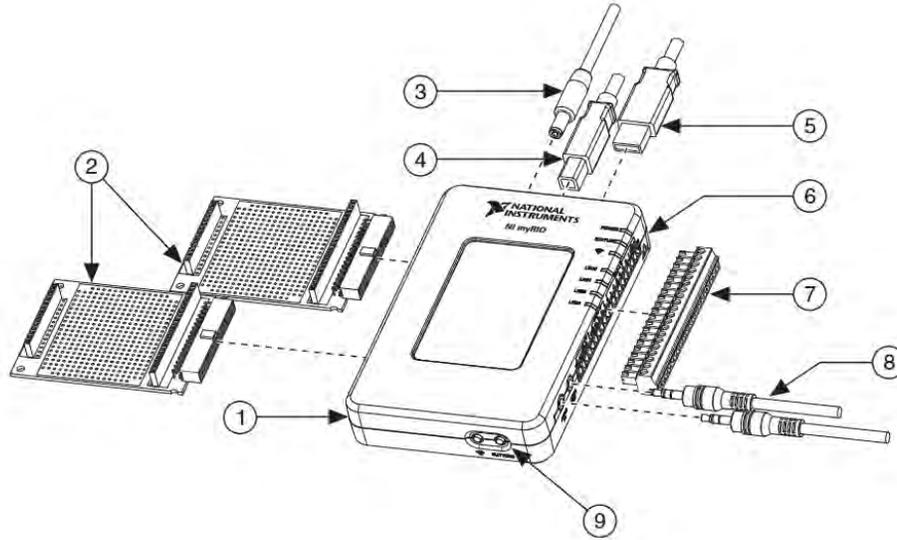


Imagen 3.5 Componentes externos de la tarjeta myRIO/1900

1. NI myRIO-1900
2. Puertos de expansión MXP
3. Cable de alimentación
4. Cable USB del dispositivo
5. Cable USB Host
6. LEDs
7. Minipuerto del sistema de terminales (MSP)
8. Cables de audio E/S
9. Botones

Especificaciones de la tarjeta NI myRIO-1900

Procesador

Tipo de procesador.....	Xilinx Z-7010
Velocidad del procesador.....	667 MHz
Núcleos del procesador.....	2

Memoria

Memoria no volátil.....	256 MB
Memoria DDR3.....	512 MB
Frecuencia de reloj DDR3.....	533 MHz
Anchura del bus de datos DDR3.....	16 bits

FPGA

Tipo de FPGA.....	Xilinx Z-7010
-------------------	---------------

Características inalámbricas

Modo de radio.....	IEEE 802.11 b, g, n
Banda de frecuencia.....	ISM 2.4 GHz
Ancho de banda.....	20 MHz
Canales.....	USA 1-11, International 1-13
TX power.....	+10 dBm Max (10 mW)
Rango inalámbrico	(línea de visión) hasta 150 m
Dirección de la antena.....	Omnidireccional
Seguridad.....	WPA, WPA2, WPA2-Enterprise

Puertos USB

Puerto de host USB.....	USB 2.0 Hi-Speed
Puerto de dispositivo USB.....	USB 2.0 Hi-Speed

Entrada analógica

Tasa de muestreo.....	500 kS/s
Resolución.....	12 bits
Protección contra sobretensiones.....	±16 V
Conectores MXP	
Configuración.....	cuatro canales single-ended por conector
Impedancia de entrada.....	>500 kΩ a 500 kS/s

	1 M Ω Potencia encendido
	4.7 k Ω Potencia apagado
Impedancia recomendada.....	3 k Ω o menor
Rango nominal.....	0 V a +5 V
Precisión absoluta.....	± 50 mV
Ancho de banda.....	>300 kHz
Conectores MSP	
Configuración.....	Dos canales diferenciales
Impedancia de entrada.....	Hasta 100 nA fuga de potencia de encendido
	4.7 k Ω Potencia apagado
Rango nominal.....	± 10 V
Voltaje de funcionamiento	
(señal + modo común).....	± 10 V de AGND
Precisión absoluta.....	± 200 m
Ancho de banda.....	20 kHz mínimo, >50 kHz normal
Entrada de audio	
Configuración.....	Una entrada estéreo
Impedancia de entrada.....	10 k Ω en DC
Rango nominal.....	± 2.5 V
Ancho de banda.....	2 Hz a >20 kHz

Salida analógica

Tasas de actualización máxima

Todos los canales AO en los conectores MXP..... 345 kS/s

Todos los canales AO en los conectores MSP

y en los canales de salida de audio..... 345 kS/s

Resolución..... 12 bits

Protección contra sobrecarga..... ± 16 V

Voltaje de encendido..... 0 V Después de la inicialización de la FPGA

Conector MXP

Configuración..... Dos canales single-ended por conector

Rango..... 0 a 5V

Precisión absoluta..... 50 mV

Corriente..... 3 mA

Velocidad de subida..... 3 V/ μ S

Conector MSP

Configuración	Dos canales single-ended por conector
Rango.....	± 10 V
Precisión absoluta.....	± 200 mV
Corriente	2 mA
Velocidad de subida.....	3V/ μ S
Salida de audio	
Configuración	Una salida de audio
Impedancia de salida	100 Ω en serie con 22 μ F
Ancho de banda.....	70 Hz a >50 kHz dentro de la carga de 32 Ω
.....	2 Hz a >50 kHz dentro de una carga de alta impedancia

Entradas y Salidas Analógicas

Número de líneas

Conectores MXP.....dos puertos de 16 líneas DIO (un puerto por conector)
una línea de UART.RX y uno UART.TX por conector

Conector MSP

un puerto de ocho líneas DIO

Control de dirección.....Cada línea DIO es individualmente programable
como entrada o salida

Nivel lógico.....5 V entrada compatible con Low Voltage

Transistor-Transistor Logic (LVTTTL, por sus siglas en ingles); 3.3 V LVTTTL de salida

Niveles lógicos de entrada

Entrada de Bajo Voltaje,(V_{IL} , por sus siglas en inglés).....0 V min; 0.8 V Max

Entrada de Alto Voltaje,(V_{IH} , por sus siglas en inglés).....2.0 V min; 5.25 V Max

Niveles lógicos de salida

Salida de Alto Voltaje,(V_{OH} , por sus siglas en inglés) adquisición 4 mA.....2.4 V min; 3.465
V Max

Salida de Bajo Voltaje,(V_{OL} , por sus siglas en inglés) reducción 4 mA 0 V min;
0.4 V Max

Ancho de pulso mínimo..... 20 ns

Frecuencias máximas para funciones digitales secundarias

SPI

4 MHz

PWM..... 100 kHz

Cuadratura de entrada del codificador..... 100 kHz

I²C..... 400 kHz

Líneas UART

Velocidad máxima de baudios..... 230,400 bps

Bits de datos..... 5, 6, 7, 8

Bits de parada..... 1, 2

Igualdad..... Impar, par, marca, Espacio

Flujo de control.....XON/XOFF

Acelerómetro

Número de ejes.....3

Rango..... ±8 g

Resolución..... 12 bits

Frecuencia de muestreo..... 800 S/s

Ruido..... 3.9 $m g_{rms}$, normal a 25 °C

Potencia de salida

+5 V de potencia de salida

Voltaje de salida 4.75 V a 5.25 V

Corriente máxima por cada conector..... 100 mA

+3.3 V potencia de salida

Voltaje de salida..... 3.0 V a 3.6 V

Corriente máxima por cada conector..... 150 mA

+15V potencia de salida

Voltaje de salida.....+15 V a +16 V

Corriente máxima32 mA (16 mA durante el encendido)

-15 V potencia de salida

Voltaje de salida.....-15 V a -16 V

Corriente máxima.....32 mA (16 mA durante el encendido)

Potencia máxima combinada de +15V y -15V de potencia de salida.....500 mW