

E.T.S. de Ingeniería Industrial, Informática  
y de Telecomunicación

# Desarrollo del Software de un Sistema de Adquisición de Datos para Ensayos de Fatiga de Palas de Aerogenerador



Máster Universitario  
en Ingeniería Industrial

Trabajo Fin de Máster

Ane Pérez Ibarrola

Idoia San Martín Biurrun y David Astrain Ulibarrena

Pamplona, 30 de junio de 2022

upna

Universidad Pública de Navarra  
Nafarroako Unibertsitate Publikoa

# Resumen

Eliminar la dependencia energética es primordial, por lo que cada vez se desarrollan más las energías renovables como la eólica. En una turbina eólica, un aumento de área barrida por el rotor proporciona mayor potencia generada, con lo que cada vez se fabrican palas de aerogeneradores de mayor tamaño. En este contexto, el Laboratorio de Ensayo de Aerogeneradores de CENER ha realizado una ampliación que permitirá realizar ensayos de acuerdo con la normativa internacional de palas de mayores dimensiones. Con el objetivo de anticiparse a posibles daños, se pretenden monitorizar la estructura de las bancadas reforzadas. Se ha implementado un Sistema de Adquisición de Datos empleando un Hardware de 'National Instruments' compuesto por módulos de entrada C y chasis Ethernet. Tras definir las características y limitaciones del sistema, se ha desarrollado un Software en LabVIEW que permite adquirir, procesar de acuerdo con la normativa de ensayos de fatiga y guardar los datos en formato binario.

**Palabras clave:** Ensayo de Palas de Aerogeneradores, SAD, Software, LabVIEW.

# Abstract

Abolishing energy reliance is becoming essential, which is why renewable energies such as wind power are growing. When increased the rotor-swept area, the wind turbine is able to generate more power, thus, wind turbine blade size has escalated recently. In this context, CENER's Wind Turbine Test Laboratory (LEA) has been enlarged to allow testing of larger blades according to international standards. In order to anticipate possible damage, the aim is to monitor the structure of the reinforced test rigs. A Data Acquisition System has been implemented using 'National Instruments' Hardware consisting of C series input modules and Ethernet chassis. Upon setting the features and constraints of the system, a LabVIEW Software has been developed to acquire data, process it according to fatigue test standards and save it in binary format.

**Key words:** Wind Turbine Blade Test, DAQ Software, LabVIEW.



# Contenido

1	PRESENTACIÓN.....	5
1.1	Introducción .....	5
1.2	Motivación.....	6
1.3	Objetivo general y descripción del proyecto.....	7
2	MARCO TECNOLÓGICO .....	8
2.1	Ensayos de Fatiga de Palas de Aerogenerador .....	8
2.2	Sistema de Adquisición de Datos .....	12
3	PROYECTO DE APLICACIÓN .....	14
3.1	Definición del Sistema Físico .....	14
3.2	Adecuación del Hardware al sistema .....	14
3.3	Elección del software .....	18
4	DISEÑO FUNCIONAL .....	19
4.1	Consideraciones generales.....	19
4.2	Diseño del programa .....	21
5	CONCLUSIONES .....	26
6	BIBLIOGRAFÍA.....	27
7	ANEXO .....	29
1	MANUAL DE USUARIO .....	31
2	MANUAL DE PROGRAMACIÓN .....	48



# Índice de figuras

Figura 1 - Generación eólica Nacional 2010-2021 [1].	5
Figura 2 - Partes aerogenerador [2]	6
Figura 3 - Principales elementos de la pala del aerogenerador [6]	8
Figura 4 - Nave de ensayo de palas CENER [9]	9
Figura 5 - Definición de deformación	10
Figura 6 - Galga extensiométrica [11]	10
Figura 7 - Puente de Wheatstone	11
Figura 8 – Componentes de un sistema de adquisición	12
Figura 9 – Efecto de la frecuencia de muestreo	15
Figura 10 - Conexión TSN Ethernet en cascada [14]	17
Figura 11 - Efecto del cerado de galga	21
Figura 12 - Esquema general del programa	22
Figura 13 - Dispositivo simulado en NI MAX	23
Figura 14 - Diagrama de bloques de creación de tareas de corriente	24
Figura 15 - Estructura de canales y tareas	24
Figura 16 - Gráfica de datos adquiridos y procesados	25
Figura 17 - Conexión de los cables Ethernet entre chasis	33
Figura 18 - Conexión con Hardware empleando NI MAX	34
Figura 19 - Primera conexión con Hardware	35
Figura 20 - Error en primera conexión con hardware	35
Figura 21 - Carpeta con los archivos para ejecutar el programa	36
Figura 22 - Pantalla de inicio del programa	36
Figura 23 - Pantalla de configuración	37
Figura 24 - Pantalla para crear una configuración	39
Figura 25 - Pantalla para modificar la configuración	40
Figura 26 - Pantalla de acceso a la adquisición	41
Figura 27 - Pantalla de espera durante la creación de tareas	41
Figura 28 - Ventana emergente para configurar la ruta de guardado	42
Figura 29 - Ventana emergente para configurar el archivo de guardado	42
Figura 30 - Pantalla de inicio de adquisición	43
Figura 31 - Pantalla durante la adquisición	43
Figura 32 - Error de creación de tareas	45
Figura 33 - Configuración de canales errónea	46
Figura 34 - Error de guardado	47
Figura 35 - Organización del proyecto	48
Figura 36 - Navegación entre las funcionalidades del programa	49
Figura 37 - Bucle 'event case' anidado a un bucle 'while'	50
Figura 38 - Bucles productor y consumidor	51
Figura 39 - Estructura general de datos	52
Figura 40 - Diagrama de flujo de la función principal del programa	53



Figura 41 - Diagrama de flujo de la función de configuración.....	54
Figura 42 - Panel frontal de la función de crear de configuración .....	55
Figura 43 - Diagrama de flujo de la función crear configuración .....	56
Figura 44 - Archivo de configuración en lector de texto .....	57
Figura 45 - Diagrama de flujo de la función modificar configuración .....	57
Figura 46 - Diagrama de flujo de la función de acceso a la adquisición .....	58
Figura 47 - Diagrama de flujo de la función de creación de tareas.....	59
Figura 48 - Diagrama de flujo de la función de inicio a la adquisición .....	60
Figura 49 - Diagrama de flujo de la función de adquisición .....	62
Figura 50 - Diagrama de bloques de la función lectura archivo de configuración .....	63
Figura 51 - Diagrama de bloques de la función Generar estructura .....	63
Figura 52- Estructura de atributos globales .....	64
Figura 53 - Estructura atributos canales.....	64



## 1 PRESENTACIÓN

### 1.1 Introducción

La Unión Europea combate el cambio climático con medidas y acuerdos ambiciosos. Pasando por una reducción considerable en las emisiones de CO<sub>2</sub> en 2030, el objetivo definido por el *Pacto verde europeo* es llegar a 2050 sin emisiones de gases de efecto invernadero. Para poder alcanzar esa meta es esencial sustituir los combustibles fósiles por fuentes de energía limpias.

La energía eólica usa la fuerza del viento para accionar una turbina y generar así electricidad. Es una de las principales fuentes de energías renovables en España y su contribución en el porcentaje de la generación de energía nacional ha ido aumentando hasta llegar en el año 2021 al 23.5%.

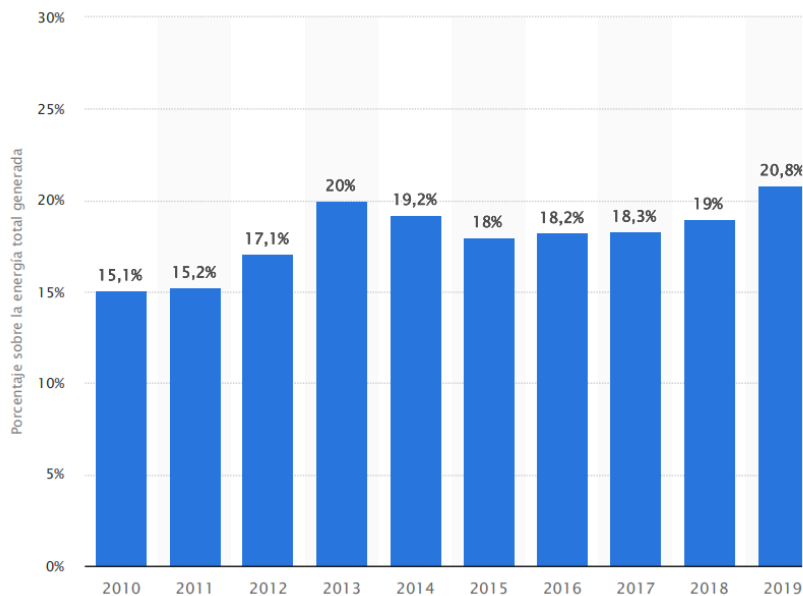


Figura 1 - Generación eólica Nacional 2010-2021 [1].

El objetivo de un aerogenerador es obtener energía eléctrica a partir de energía cinética del viento. El dispositivo consta de las partes que se pueden ver en la Figura 2.

Las palas montadas en el rotor son las encargadas de aprovechar la energía cinética del viento gracias a su perfil aerodinámico. El viento las hace girar en torno al buje y acciona un eje mecánico colocado dentro de la góndola. Para evitar el choque con el suelo en el giro y conseguir un viento lo menos afectado posible por el terreno, el rotor está montado encima de una torre que le da altura. Los aerogeneradores más comunes en la industria son los de tres palas ya que con ellos se consigue mayor eficiencia sin perder estabilidad.

A través de una multiplicadora se transfiere esa energía mecánica al eje de alta velocidad. Éste está conectado al generador, que transforma la energía de giro en electricidad. Dentro de



la góndola se encuentran además un freno mecánico que se emplea para paradas de mantenimiento o cuando la velocidad del viento supera un límite.

Existen además dos mecanismos de orientación que ayudan a obtener la máxima energía del viento. Las palas giran respecto a su eje con el control de pitch y la torre se orienta con el motor Yaw. Para determinar la dirección y calcular la velocidad del viento los aerogeneradores cuentan con una veleta y un anemómetro encima de la góndola.

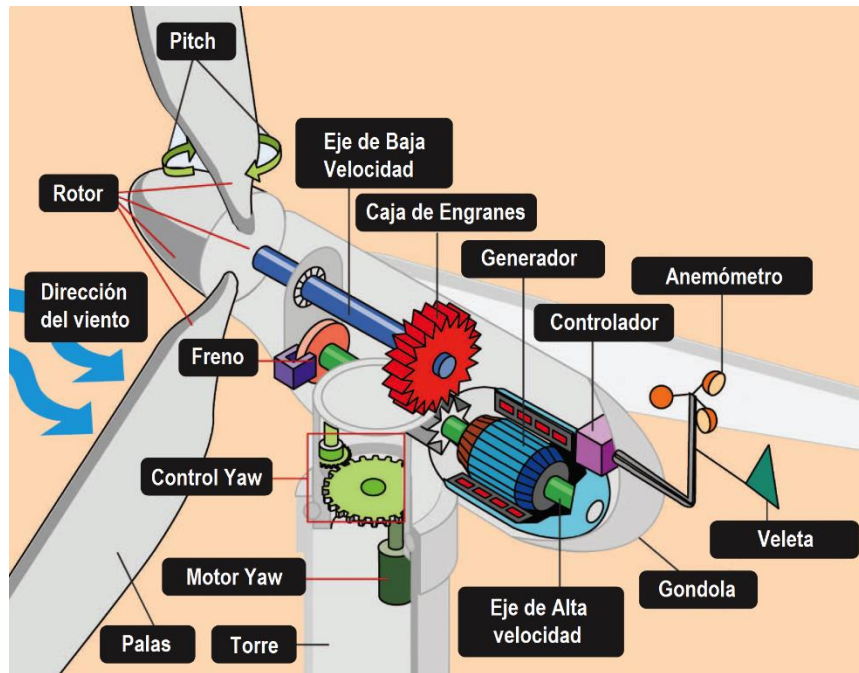


Figura 2 - Partes aerogenerador [2]

A lo largo de su ciclo de vida, el aerogenerador está expuesto a una serie de situaciones que se pueden traducir en cargas máximas. El diseño de los diferentes componentes está realizado para que puedan soportar esas situaciones extremas.

Para poder montar un aerogenerador en un parque eólico su diseño debe estar certificado. Esa certificación debe ser expedida por una 'Entidad Certificadora' ajena al proceso de diseño.

El Centro Nacional de Energías Renovables (CENER) lleva a cabo investigación técnica y ofrece soporte tecnológico. Entre sus instalaciones cuenta con laboratorios acreditados por la Entidad Nacional de Acreditación (ENAC) para realizar ensayos de acuerdo con la norma internacional.

## 1.2 Motivación

El Laboratorio de Ensayos de Aerogeneradores situado en Sangüesa, dispone de una nave con dos bancadas para el ensayo de palas completas de aerogenerador. Recientemente, para adaptarse al crecimiento continuo del tamaño de palas, se ha realizado una ampliación de la nave. Al adaptar la nave a mayores tamaños de pala, las cargas que tiene que soportar cada una





de las bancadas ha aumentado y, por ello, se ha contratado una entidad externa para diseñar un refuerzo y se ha implementado.

Se quiere realizar el seguimiento de *vida a fatiga* de esa expansión. Esta monitorización estructural permitirá anteponerse a posibles fallos de diseño. Con los datos obtenidos es posible observar si los esfuerzos de la estructura superan los cálculos proyectados y tomar medidas al respecto antes de que ocurra un daño permanente o irreparable.

Para ello, en las barras críticas de la armadura del hormigón se han colocado una serie de galgas extensiométricas y se realizará una monitorización sincronizada de esas galgas durante los *ensayos de fatiga*. Para llevarlo a cabo es necesario un equipo de adquisición de datos y un software que lo controle.

### 1.3 Objetivo general y descripción del proyecto

**El propósito de este proyecto es producir un *ejecutable* que controle la adquisición de los sensores colocados en la bancada de un *ensayo de fatiga* de palas de aerogenerador.**

Se ha desarrollado en el Centro Nacional de Energías Renovables, implementándolo en un hardware de adquisición ya disponible en la empresa. El Software se ha realizado en un entorno de programación gráfica, LabVIEW, debido a la adecuación del sistema. Sin embargo, no había ningún miembro del grupo de trabajo con conocimientos suficientes como para guiar la programación. Por lo que la fase inicial del proyecto ha consistido en documentarse sobre el lenguaje y realizar ejemplos proporcionados por diversos libros [3]–[5]. Una vez obtenidos el conocimiento suficiente como para llevar a cabo una aplicación de gran tamaño, se ha procedido con los pasos documentados en esta memoria.

A lo largo de este documento se pueden encontrar tres apartados principales:

- Introducción teórica de los contenidos que se han trabajado en este proyecto
- Análisis previo de las necesidades del sistema
- Descripción del producto final

En el contenido agregado se encuentra el Manual de Usuario y el Manual de Programación como guía de utilización y modificación del programa.

Se complementa además con un segundo documento llamado ‘Validación e Instalación del Software de un Sistema de Adquisición de Datos para Ensayos de Fatiga de Palas de Aerogeneradores’ en el cual se puede encontrar la puesta en marcha del sistema completo en el destino y una validación del software.

## 2 MARCO TECNOLÓGICO

Para tener una mejor comprensión del desarrollo del software, es importante tener unos conceptos básicos sobre los términos principales del proyecto.

### 2.1 Ensayos de Fatiga de Palas de Aerogenerador

En este apartado se realiza una breve introducción a las palas de un aerogenerador, el ensayo de palas y su normativa. Finalmente, nos centraremos en los *ensayos de fatiga* que es donde se sitúa el software desarrollado en este proyecto.

**La pala del aerogenerador** es la encargada de convertir la energía cinética del viento en energía mecánica. Las palas de aerogenerador suponen entorno a un 22% del coste del mismo y se fabrican en moldes diseñados específicamente para ello a partir de material compuesto. Normalmente suele ser una mezcla entre fibra de vidrio y resina epoxi. También pueden emplearse en su lugar poliéster y se puede añadir fibra de carbono para incrementar la resistencia y la rigidez.

Cuando se diseña una pala se intenta conseguir el perfil más aerodinámico posible, para obtener máxima sustentación y generar máximo par de giro para accionar el generador. Se debe mantener de cualquier forma la estabilidad estructural. Por ello, se emplean núcleos para aumentar el espesor del laminado y así aumentar el momento de inercia con el menor aumento de peso posible. Éstos pueden ser de madera de balsa, de PVC o de nomex.

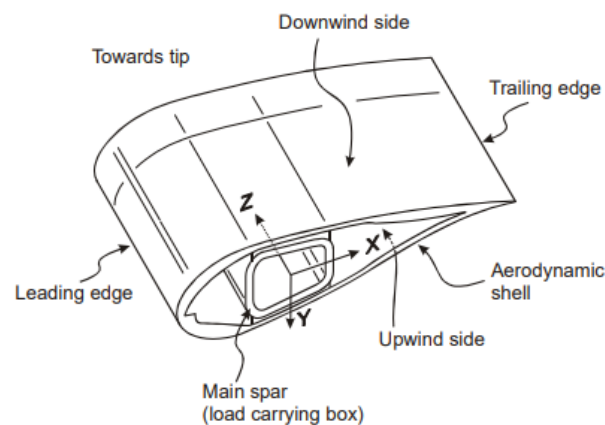


Figura 3 - Principales elementos de la pala del aerogenerador [6]

En la Figura 3 se describen los principales elementos del perfil de la pala del aerogenerador. En este caso concreto, la estructura portadora de carga se trata de una viga cajón, pero también existen 'spar caps' o estructura portante con costillas. La parte superior se le denomina zona de succión y la inferior la de presión. La dirección entre éstas se le denomina 'Flap'. En cambio, la dirección, entre el borde de ataque y de salida, de 'leading' a 'trailing', se le llama 'Edge'.

De acuerdo con el estándar realizado por la Comisión Electrotécnica Internacional (IEC), cuando un fabricante de palas de aerogenerador realiza cambios considerables en el diseño,



debe realizar una serie de ensayos a través de una entidad certificadora. Los requisitos que tiene que cumplir ese ensayo están descritos en el apartado **IEC TS 61400 – 23**. El objetivo final del ensayo es validar las propiedades estructurales de una pala en tamaño real. En definitiva, que a lo largo de su ciclo de vida va a ser capaz de soportar las cargas para las que está diseñada. Previamente los fabricantes han podido realizar ensayos de acuerdo con el estándar para validar su diseño, pero la certificación se debe realizar a través de un organismo independiente. [7], [8]



Figura 4 - Nave de ensayo de palas CENER [9]

Se contemplan diferentes ensayos en el estándar, los cuales deben realizarse en un orden concreto. En primer lugar, se obtienen propiedades de la pala como el peso, el centro de gravedad, distribución de rigidez o las frecuencias de resonancia. A continuación, se realiza un *ensayo estático*, seguido de un *ensayo a fatiga*, y finalmente, un *ensayo estático post fatiga*.

En los ensayos estáticos se expone la pala a las cargas máximas que se pretende que trabaje a lo largo de sus años de operación. Por otro lado, el ensayo a fatiga replica el daño que sufre la pala de aerogenerador a lo largo de su ciclo de vida. Dado que las condiciones extremas de trabajo pueden darse al final de la vida útil del aerogenerador, se realiza un ensayo estático post fatiga comparando la rigidez obtenida en ambos casos.

Para validar el diseño y la fabricación de la pala se asume que los parámetros del ensayo se han calculado de acuerdo con estándares de diseño como IEC 61400-1, que especifica cómo realizar el diseño para garantizar la seguridad de los aerogeneradores.

En un **ensayo a fatiga** se somete la pala del aerogenerador a cargas dinámicas que equivalen a los esfuerzos que está expuesta en su día a día a lo largo de sus 20-25 años de vida. Para poder mover esas estructuras tan grandes y pesadas, se las hace vibrar a su frecuencia de resonancia.

El fabricante de la pala debe proporcionar detalles técnicos de la misma como la longitud, peso estimado, centro de gravedad, detalles de la raíz y puntos de manipulación.

El ensayo de fatiga consiste en lo siguiente:

Se iza la pala colocándola en la bancada, una estructura estática, y en una sección determinada por el diseñador de la pala se coloca un actuador móvil. Este actuador se desplazará a la frecuencia de resonancia de la pala y de esta manera se expone la pala a efectos de compresión y tensión. Se realiza este ensayo tanto en la dirección de 'Flap' como 'Edge' y se alcanzan entre 1 millón y 5 millones de ciclos por lo que la duración total del ensayo se puede prolongar hasta varios meses.

En el caso de que el ensayo haya sido satisfactorio, el fabricante de pala puede decidir llevarla hasta el punto de fractura. De esta forma se puede evaluar qué margen tiene la pala para futuras mejoras en el diseño e identificar los puntos débiles.

Un elemento clave en el ensayo es, sin duda, la **galga extensiométrica** [10]. Es el instrumento empleado comúnmente para medir deformación. Cuando a un objeto se le aplica una fuerza, éste se va a elongar en ese mismo sentido. La deformación se define como el ratio entre la variación de longitud y la longitud original.

$$\varepsilon = \frac{\Delta L}{L} \quad (1)$$

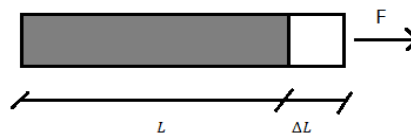


Figura 5 - Definición de deformación

La galga extensiométrica se basa en el cambio de resistencia en proporción con la deformación del objeto. En la dirección vertical se coloca hilo en forma de rejilla, de esta manera se consigue la máxima variación de resistencia.

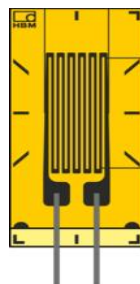


Figura 6 - Galga extensiométrica [11]

Para caracterizar la relación entre el cambio en la Resistencia eléctrica y la deformación se emplea el factor de galga (GF). Habitualmente en las galgas extensiométricas comerciales este parámetro toma un valor muy próximo a 2 y será siempre proporcionado por el fabricante.

$$GF = \frac{\Delta R/R}{\Delta L/L} = \frac{\Delta R/R}{\varepsilon} \quad (2)$$



Las variaciones de resistencia que se dan son muy pequeñas en magnitud respecto al valor en reposo de la galga. Por ello, es necesario acondicionar la señal obtenida empleando un puente de Wheatstone.

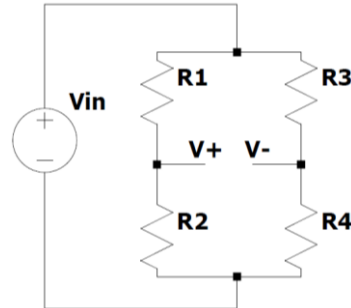


Figura 7 - Puente de Wheatstone

Como se observa en la Figura 7 está formado por cuatro resistencias divididas en dos ramas que se alimentan con tensión de excitación  $V_{in}$ . La tensión que se obtiene del puente es la resta entre dos divisores de tensión. Si hacemos que con falta de carga las dos ramas estén equilibradas ( $R_1/R_2 = R_3/R_4$ ), obtenemos una tensión de salida nula. Se consigue así un voltaje de salida en función de la deformación.

$$V_0 = V^+ - V^- = V_{in} \frac{R_2}{R_1+R_2} - V_{in} \frac{R_4}{R_3+R_4} = V_{in} \left( \frac{R_2}{R_1+R_2} - \frac{R_4}{R_3+R_4} \right) \quad (3)$$

Existen distintos tipos de configuraciones de galgas extensiométricas, según la cantidad de elementos activos que se introducen en el puente de Wheatstone. Con un único elemento, se llama configuración de cuarto de puente y es más fácil instalar, pero conlleva obtener menos sensibilidad y no ser capaz de combatir las derivas térmicas.

Conforme se incrementan los elementos activos, se pueden compensar las variaciones ante la temperatura. Además, aumenta la sensibilidad total del puente, por ejemplo con la configuración de medio puente que cuenta con dos galgas y dos resistencias fijas.

Finalmente, con la configuración de puente completo se puede distinguir entre las diferentes fuerzas que actúan en el objeto. Es decir, puedes medir únicamente fuerza axial, pandeo o torsión. Para ello existen diferentes formas de colocación de los cuatro elementos. Sin embargo, encarece la instalación y es necesario el acceso a ambos lados del elemento.

Para los *ensayos de fatiga* es suficiente con emplear galgas de cuarto puente sin compensación térmica. Esta instalación es la más sencilla como se ha mencionado y requiere soldar el mínimo número de cables de conexión. Como se emplean un gran número de galgas en los ensayos la facilidad de instalación es esencial.

Para instalar las galgas es necesario limpiar la superficie donde se van a adherir. La única forma de que la resistencia de la galga sufra la misma deformación que el objeto que se quiere medir es que esté perfectamente pegada. Es necesario comprobar después si la galga mide su

valor nominal ya que si hay defectos en la instalación los valores medidos pueden ser incorrectos.

## 2.2 Sistema de Adquisición de Datos

Con el avance de la tecnología, los sistemas se vuelven más ‘inteligentes’. Aumenta la cantidad de datos necesarios para comprender los sistemas físicos y esta información debe ser sincronizada. Cuanto más compleja es la aplicación existe un mayor número de conexiones y sube el coste. Si empleamos un Sistema de Adquisición de Datos (SAD) se puede automatizar la recogida de datos y se simplifican las conexiones.

Un SAD mide y/o controla propiedades físicas empleando un conjunto de Hardware y Software. Sus diferentes componentes se pueden apreciar en la Figura 8. Cuando ocurre un fenómeno físico, como un cambio de presión o de temperatura, causa un cambio de estado eléctrico en el *sensor*. Esta señal generalmente analógica se  *acondiciona*  y convierte en señal digital empleando un  *dispositivo DAQ* . Vía Bus de comunicación la señal llega al  *PC*  donde se encuentran el software del ‘*driver*’ y de la aplicación.



Figura 8 – Componentes de un sistema de adquisición

Un sistema de adquisición es capaz de medir cientos de señales de distintos tipos. Se emplean tanto en laboratorios para llevar a cabo investigaciones como en la industria para administrar procesos y permite tramitar las señales de entrada y salida desde un mismo ordenador.

A lo largo de esta memoria se va a emplear el término *muestrear*. Se trata del proceso de obtener los datos a través de un Conversor Analógico Digital (ADC). Una señal analógica es continua en el tiempo y se discretiza a una velocidad denominada frecuencia de muestreo. Cuanto mayor es esta frecuencia, la señal analógica proveniente del Sistema físico quedará mejor caracterizada.

Además de la conversión a señal digital, el acondicionamiento puede incluir una amplificación de los componentes deseados de la señal y un filtrado del ruido no deseado. En el caso de las galgas extensiométricas por ejemplo el acondicionamiento de éstas incluye también el circuito de puente de Wheatstone.



El **Hardware de adquisición** hace de enlace entre el entorno muestreado y el ordenador. Es el encargado de organizar la cronología del acondicionamiento de la señal y administra a su vez las diferentes señales.

En este caso el hardware que se va a emplear es de 'National Instruments'. Está especializado en sistemas de medida automatizados y tiene una amplia gama de productos que se adaptan a necesidades concretas.

El **Software de adquisición** debe integrar todo el Hardware. Entre las diferentes funciones que abarca está adquirir, almacenar, procesar y analizar los datos. El programa desarrollado debe configurar y servir como enlace con la *adquisición*.

### 3 PROYECTO DE APLICACIÓN

Una vez conocidos los diferentes componentes de un sistema de adquisición de datos, se va a realizar un análisis sobre las necesidades concretas de adquisición en la medición de esfuerzos de las bancadas durante ensayos de fatiga en palas de aerogeneradores.

#### 3.1 Definición del Sistema Físico

Antes de diseñar la aplicación es importante realizar un análisis sobre las características físicas que se quieren observar, que transductor se va a emplear y cómo van a ser sus variaciones de estado. En definitiva, identificar y caracterizar las señales a adquirir. Para ello es necesario describir las bancadas y los sensores empleados en su monitorización estructural.

El plan de ampliación de la bancada consiste en reforzar externamente con hormigón armado la estructura ya existente. Entre la estructura antigua y la nueva, se emplean unas barras de conexión para transmitir las cargas.

Se ha realizado un plan de galgado por cada bancada donde se priorizan las barras críticas, tanto de conexión como de armadura del refuerzo exterior. Las galgas empleadas son uniaxiales y será necesario realizar un acondicionamiento de cuarto de puente.

Al someterse a un ensayo de fatiga, tanto 'Edge' como 'Flap', las galgas extensiométricas sufrirán una tensión cíclica cuyo periodo coincide con la frecuencia del actuador. Dado el elevado número de sensores empleados, se transmite la señal desde éstos al equipo de adquisición mediante unas mangueras que agrupan los terminales de múltiples galgas. En estas mangueras van pares trenzados para disminuir el ruido derivado del acoplamiento de señales con una malla de tierra alrededor [12].

En resumen, a la salida de los sensores se esperan señales analógicas senoidales con un periodo cíclico y no demasiado ruido. Estas señales provenientes de las galgas extensiométricas serán complementadas con señales de entrada en corriente y tensión.

#### 3.2 Adecuación del Hardware al sistema

En este apartado se van a tratar los diferentes puntos a tener en cuenta cuando se escoge un hardware de adquisición [13]. En este caso no se ha escogido un equipo específico adaptado a las necesidades de este proyecto, sino que se va a reutilizar el sistema empleado en un proyecto previo. En concreto se trata de un conjunto de chasis cDAQ Ethernet con módulos de entrada de la serie C. Sin embargo, será necesario contemplar la adecuación de éste al equipo objetivo.

Es importante tener claro qué **tipo de señales** se van a tratar. La complejidad del equipamiento puede variar desde sistemas que están enfocados únicamente a leer señales digitales, a sistemas más completos que permiten tanto leer como generar señales digitales y analógicas.





En este caso se trata de un sistema de monitorización pasivo, el cual leerá solamente señales analógicas.

Dado que es muy probable que el sistema de medida aumente el número de señales adquiridas con el tiempo, la escalabilidad del dispositivo escogido es también un factor para tener en cuenta. Una opción posible para conseguirlo es emplear un **sistema modular**, que adaptas a las necesidades concretas del sistema a medir. Éste cuenta con un chasis que sincroniza múltiples módulos de entrada y salida.

Sensores como los que se van a emplear en este sistema (galgas, medidores de corriente...), requieren habitualmente un acondicionamiento de la señal. Este **acondicionamiento** puede ser previo, o puede estar incluido en el módulo. Por ejemplo, el módulo NI 9236 está adaptado para conectar directamente galgas extensiométricas y provee el acondicionamiento de cuarto de puente. Por lo que, de esta manera, se evitan circuitos electrónicos adicionales para la ampliación, el filtrado o la inserción de esa galga en el puente.

Todas las señales que se van a medir cuentan con módulos adecuadas para ellas:

- NI9236 para las galgas extensiométricas
- NI9205 para la medida de voltaje
- NI9203 para la medida de señales de corriente de poca amplitud

Estos módulos pueden ser empleados en dos gamas de chasis: CompactRIO y Compact DAQ. La gran diferencia entre ambos es la necesidad de la temporización en el tratamiento de la señal.

La **frecuencia de adquisición** debe cumplir siempre el criterio de Nyquist, por el cual debemos muestrear toda señal al menos al doble de su componente de mayor frecuencia. Sin embargo, para una mayor representación de la señal, se recomienda un muestreo de diez veces la frecuencia de la señal.

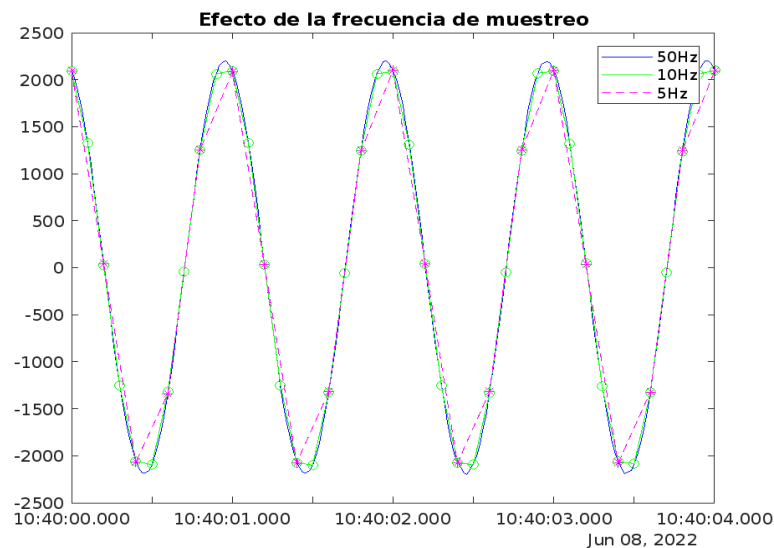


Figura 9 – Efecto de la frecuencia de muestreo



En la Figura 9 se puede ver el resultado de muestrear una señal sinodal de periodo de 1 s a diferentes frecuencias. Con 5 Hz la señal queda bastante deconstruida, y no es hasta 50 Hz que se define correctamente.

Se va a trabajar con señales tanto estáticas como senoidales, las cuales tendrán una frecuencia entre 1 Hz y 2 Hz. Con lo cual, si se muestrea a 50 Hz, se cumplen siempre las condiciones necesarias.

El dispositivo de adquisición siempre tiene que estar localizado lo más cerca posible al ordenador desde el cual se va a controlar el sistema y de esta manera se mantendrá la integridad de la señal. Sin embargo, cuando se trata de la monitorización de estructuras, el sistema se encuentra distribuido. Se puede aislar el ordenador del sistema de medida empleando tecnología inalámbrica. La información puede ser adquirida en diferentes puntos y enviada a una localización central. En estos casos el mejor **bus de conexión** a emplear es Ethernet. Empleando dispositivos con este bus, se consigue además un sistema portable, de fácil instalación y con posibilidad de controlarlo a través del ordenador.

Aunque en la aplicación en cuestión la estructura a monitorizar no tiene un tamaño tan grande como el de un edificio, y quizá esta necesidad de medida distribuida no sea tan esencial como en la monitorización de otras estructuras, es el tipo de bus que estaba disponible en la empresa. Por lo tanto, es el que se empleará. Así que es necesario comprobar si el resto de las características de Ethernet colisionan con la aplicación o entran dentro de los requisitos.

El **ancho de banda** máximo permitido es de 125 MB/s. La mayoría de los canales realizarán la medición de gas extensiométricas, empleando un módulo de características, 10 kS/s/ch con 8 canales y 24 bits de resolución. El sistema completo contiene 250 canales, si asumimos que todos están dedicados a extensiométrica, se necesitaría un ancho de banda de 7.5 MB/s. Esto supondría únicamente un **17% del disponible**, con un amplio margen para escalabilidad.

Una de las desventajas que tiene este tipo de bus es el retraso. El tiempo que ocurre desde que se adquiere una serie de datos hasta que esta información llega al ordenador central es bastante elevado comparado con otro tipo de conexiones.

En la aplicación en cuestión no es ningún problema, ya que no estamos trabajando en tiempo real y la frecuencia de adquisición deseada es de 50 Hz. El sistema no tiene unas alarmas ni un control específico, sino que únicamente provee una información adicional a los ingenieros de ensayos sobre los esfuerzos a los que está sometida la estructura.

En definitiva, la adquisición no necesita ser *determinista*, los datos no necesitan ser analizados a una frecuencia concreta para procesarla con un periodo fijo y tomar acciones periódicas, sino que únicamente nos importa que los datos tomados mantengan su valor temporal a los que asociarlos y el procesamiento se realice de manera que no se sature ningún búfer que produzca errores en el programa.

En todo sistema de adquisición es necesario relacionar las diferentes medidas que se están tomando. Para ello los diferentes equipos empleados deben contar con **un sistema de**



**sincronización.** Es posible establecer individualmente el reloj de cada dispositivo a una base de tiempos común empleando 'Time-Sensitive Networking' (TSN). Se trata de una sincronización proporcionada por el standard IEEE 802.1As, donde se especifica el protocolo y procedimiento para garantizar que se cumplen los requerimientos de una aplicación sensible al tiempo. La sincronización entre dispositivos emplea comunicación basada en paquetes y se puede emplear en distancias largas sin causar grandes retrasos en la propagación de la señal. Ente los terminales tiene que existir una vía con 'switches' de acuerdo con el protocolo (IEEE 802.1AS). Cuando ésta exista, se detectará automáticamente. Los chasis Compact DAQ 9185 y 9189 llevan integrados la estructura necesaria, por lo que pueden conectarse directamente.

Es posible emplear TSN en tipología de *anillo*, *estrella* o *línea*. Pero tanto la tipología de estrella como la de anillo requieren un 'switch' externo. En cambio, con la de línea es suficiente con conectar en cascada los diferentes dispositivos, siempre y cuando cumplan el protocolo mencionado. Sin embargo, en el caso de que un nodo pierda conexión o alimentación, los nodos aguas abajo sufrirán la misma pérdida.

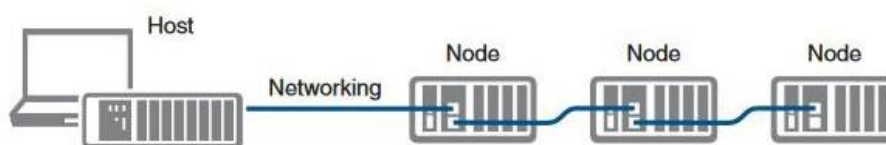


Figura 10 - Conexión TSN Ethernet en cascada [14]

Para escoger la resolución del dispositivo DAQ, hay que atender al número de bits que emplea el ADC. Excepto en aplicaciones con exigencias de rango dinámico muy alto, se cumplirán todos los requisitos al emplear resolución de 16 bits.

El **error de la medida** permitido también varía según la aplicación. Una manera de medirlo es a través del parámetro precisión absoluta. Éste depende del error de ganancia, error 'offset' y de la no-linealidad. Es decir, depende tanto del instrumento de medida como de la señal. Sistemas más estrictos pueden exigir una precisión absoluta de hasta 1 mV.

Las galgas extensiométricas son sensores muy precisos que pueden proporcionar mediciones hasta de partes por millón. De hecho, la unidad de medida es micro deformaciones. Para tratar de eliminar el efecto de la Resistencia del conductor eléctrico, a cada extremo de la galga están conectados dos cables. Además, no se trata en este caso de una aplicación crítica a nivel de resolución. La atención se va a poner en la evolución de los máximos y mínimos de cada ciclo. Si éstos se mantienen estables, no ocurre ningún problema con la expansión de la bancada. En cambio, si las variaciones a lo largo del ensayo, o de los ensayos de incrementan cada vez más o sufren un escalón, será necesario replantearse ese refuerzo y tomar medidas al respecto.

Finalmente, hay que tener en cuenta el **entorno** en el que se va a trabajar. Existen dispositivos específicos para trabajar en condiciones adversas, como por ejemplo la exposición



a lluvia y humedad. En este caso concreto que estamos exponiendo no será necesario ya que se encuentra en el interior. Sí que la nave de ensayo de palas es una zona expuesta a vibraciones y susceptible a impactos, debido al manejo de cargas suspendidas y, por ello, será necesario proteger el equipo de adquisición con armarios protectores.

Se emplearán en concreto armarios metálicos de la gama de MAS de 'NVENT HOFFMAN'. Son de acero y tienen una protección IP66 (IEC 60529) que evita que entren el polvo y el agua reduciendo así la posibilidad de fallo y resistencia ante impactos directos, IK 10 (IEC 622 62).

### 3.3 Elección del software

Una vez el hardware esté definido, es necesario tomar decisiones sobre el software y completar así el sistema de adquisición.

La primera decisión por tomar es el **driver** que se va a emplear. Es necesario que el controlador sea compatible tanto con el hardware como con el programa. En este caso, al tratarse de un sistema de 'National Instruments', el driver que se ha escogido es NI DAQmx.

Gracias a sus funciones polimórficas, NI-DAQmx permite ahorrar tiempo en el desarrollo de aplicaciones de adquisición de datos. Estas funciones polimórficas son multifunciones que se pueden emplear en diferentes tipos de dispositivos. Con un mismo bloque de programación se puede leer tanto una entrada digital como una entrada analógica o implementar un contador. Además, mejora el rendimiento general gracias a su arquitectura multihilo. De esta manera varias operaciones pueden realizarse simultáneamente.

DAQmx puede ser empleada en lenguajes de programación de texto como C o .NET, o en LabVIEW (Laboratory Virtual Instrument Engineering Workbench). Este último es un entorno de programación gráfica desarrollado por National Instruments. Los programas y subrutinas se denominan VIs y contienen tres elementos: Un panel frontal donde interactúa el usuario, un diagrama de bloques en el cual mediante hilos se realizan las conexiones y un panel de conexiones para indicar las entradas y salidas de las subrutinas.

El lenguaje empleado es un idioma de programación de flujo de datos, donde un Vi se ejecuta cuando los datos de entrada están disponibles. El flujo de ejecución se define por tanto en el diagrama de bloques.

## 4 DISEÑO FUNCIONAL

En este apartado se van a describir los pasos seguidos en el desarrollo de la aplicación, terminando con las características generales de ella. Si se quiere una descripción más detallada sobre el desarrollo del programa en el anexo se puede encontrar el Manual de Programación. Si, en cambio, se quiere un paso a paso de las funcionalidades que ofrece, ver el Manual de Usuario.

Antes de describir el software y su desarrollo, es importante introducir el concepto de *tarea*. Se trata de un conjunto de configuraciones agrupadas por canales que incluye entre otras características el nombre, el canal físico y el tipo de medida. Además, agrupa una temporización común y se le puede asignar activación u otras propiedades. [3]–[5]

Esta es la herramienta que emplea el driver de NI-DAQmx para controlar el dispositivo de *adquisición*. Por lo que a cada chasis empleado se le asignará al menos una tarea que coordine la *adquisición* de todos los canales contenidos en sus módulos. Debido a la limitación del bus de comunicación empleado, ‘Ethernet’, no se podrá asignar más de dos tareas a cada uno de los chasis.

### 4.1 Consideraciones generales

Cuando se desarrolla una aplicación es necesario tener claro los requisitos que ésta debe cumplir. Por un lado, están las entradas y salidas que tendrá el software y por otro las tareas a realizar.

En este caso, las entradas al programa son de las características del ensayo. En concreto, una serie de valores generales que definirán la frecuencia tanto de la adquisición como del actuador y otras entradas correspondientes a los diferentes canales empleados. Para introducir todos estos valores en el programa se va a emplear un archivo de configuración con formato ‘.ini’. Este formato se divide en secciones y campos. Los primeros son más generales y se corresponderá cada una con un canal. Los campos definen las diferentes características del canal. Por ejemplo, qué tipo de canal se trata o qué posición tomará en las entradas del sistema de adquisición.

Tabla 1- Inputs al programa

Sección general		Sección de cada canal	
Frecuencia de adquisición	Cantidad de datos a almacenar en un archivo binario	Tipo de canal	Posición del canal
Frecuencia del actuador		Ganancia/GF (factor de galga)	‘Offset’

Las salidas del programa están definidas por el formato actual que emplean en los ensayos de palas en el LEA. Dada la duración extensa de los ensayos, los datos obtenidos se guardan en archivos binarios que evitan que se llene la memoria. Además, para poder introducir esos archivos binarios en un postproceso concreto, los atributos que se escriben tendrán un orden y



tamaño de dato específico. El software empleado en los ensayos es de HBM y éste proporciona una función para poder leer los archivos generados ([Catman File Importer - File Exchange - MATLAB Central \(mathworks.com\)](#)), por lo que se va a hacer coincidir el formato de salida de la aplicación con un archivo compatible con la función de Matlab.

### Tareas a realizar

Una vez decididas las entradas y salidas del programa, se definen las tareas que éste debe realizar. El fundamento de esta aplicación es la adquisición de datos, por lo que va a ser necesario realizar un muestreo simultáneo de todos los canales y una administración correcta de los datos obtenidos.

Para poder realizar el postproceso adecuado a un Ensayo de Fatiga de palas de Aerogeneradores, en cada ciclo del actuador deben calcularse máximos y mínimos en cada uno de los canales. Esos máximos y mínimos se guardarán en el formato binario mencionado previamente, con lo cual se dedicarán subrutinas únicamente a traducir los datos y escribir un archivo.

Por otro lado, se quiere poder realizar y alterar los archivos de configuración de entrada al programa y, para ello, habrá un conjunto de funciones dedicadas a leer y escribir archivos '.ini'. El SAD se va a implementar en una estructura fija donde la monitorización está estructurada de la misma manera siempre, por lo que las señales que se van a adquirir apenas van a variar. Si cada vez que se quiera realizar un nuevo ensayo ligeramente diferente, como por ejemplo retirando o añadiendo un único canal, es necesario volver a crear desde cero la configuración, supone evidentemente una pérdida de tiempo de los ingenieros a cargo del ensayo. Así que es importante poder modificar los archivos de configuración a través de la aplicación.

Finalmente, se va a implementar el cerado de galgas. Cuando se acondiciona un sensor en un puente de Wheatstone, la salida sin carga no es 0 V, sino que proporciona un valor aleatorio de señal analógica que se puede denominar como 'offset'. Este desviamiento del cero se debe a ligeras variaciones entre las resistencias empleadas en ambas ramas o debido a efectos residuales de la instalación de la galga. Este valor no aporta información sobre el esfuerzo al que está sometido el sistema durante el ensayo, por lo que no es un valor que se quiera mantener. Antes de realizar un ensayo, se realiza un cancelado de ese 'offset'. De esta manera se consigue que la onda sinodal que se obtiene del ensayo esté centrada en cero.

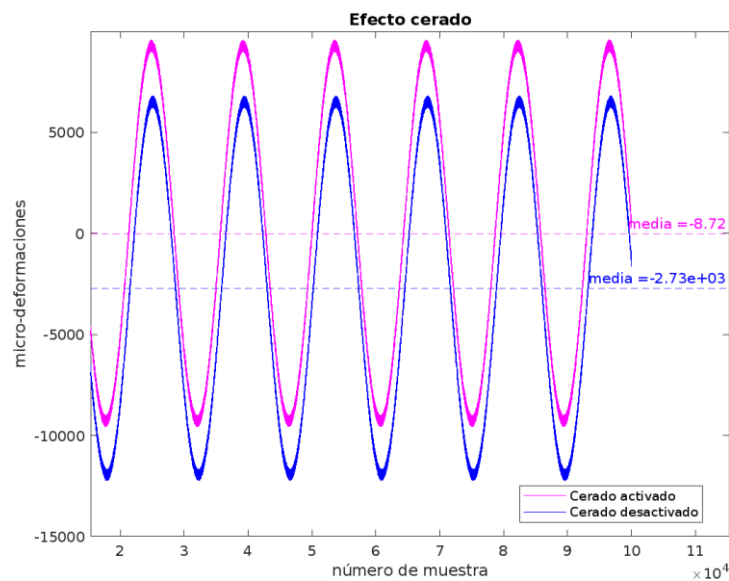


Figura 11 - Efecto del cerado de galga

Es posible cancelar el 'offset' tanto por hardware como por software. La cancelación por Hardware se realiza reemplazando una de las resistencias del puente por un potenciómetro para hacer que el voltaje inicial sea cero; o implementando un circuito que proporcione el voltaje necesario para anular el valor. En este caso la medida que se va a implementar es por compensación de software. Se toma el valor inicial y éste se emplea cuando se escalan los valores muestreados. Este método es fácil de implementar y el único posible cuando se está desarrollando la aplicación. Sin embargo, tiene un inconveniente: dado que la diferencia entre las ramas del puente nunca se llega a cancelar, el rango del convertidor ADC queda limitado.

## 4.2 Diseño del programa

Al tener todos los requisitos establecidos, es recomendable realizar un diagrama general del flujo del programa donde, además de las funcionalidades contenidas, se describe cómo navegará el usuario entre los diferentes procesos.

Como podemos apreciar en la Figura 12, habrá dos partes diferenciadas en el programa. Una estará dedicada exclusivamente a generar archivos de configuración. La otra contiene todo el proceso de adquisición desde la creación de *tasks* hasta el almacenamiento final de los datos.

Las flechas que circulan entre los bloques principales coinciden con los controles que se le habilitarán al usuario para circular entre las diferentes subrutinas del programa.

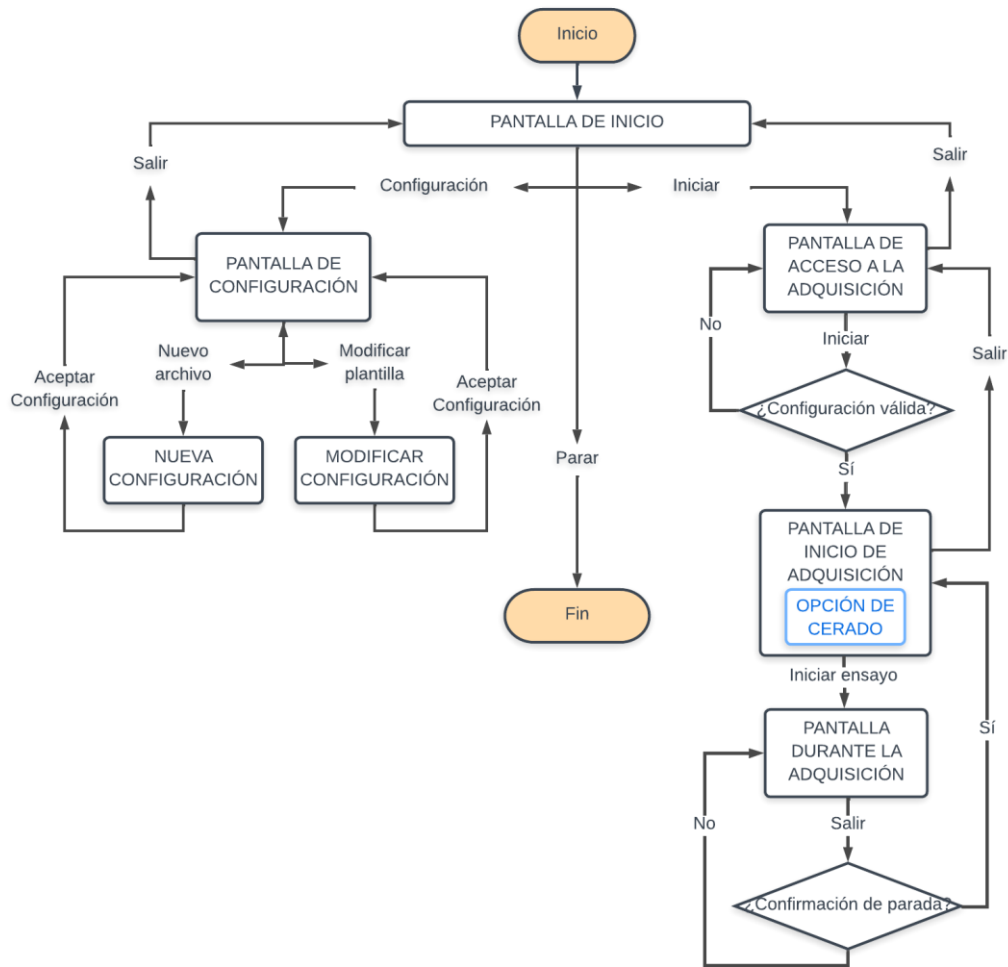


Figura 12 - Esquema general del programa

Una vez la estructura está clara, se lleva a cabo el desarrollo de cada bloque principal. Los bloques dedicados a la manipulación de archivos o datos pueden comprobarse empleando las diferentes herramientas de ‘debugging’ que ofrece LabVIEW. Como la ventana de ‘Error List’ o la ejecución paso a paso. Para la ejecución de los bloques relacionados con la adquisición en cambio, es necesario un dispositivo donde implementarlo.

NI Measurement & Automation Explorer (MAX) es una herramienta que proporciona acceso al Hardware de NI. Se instala junto con el driver NI DAQmx y ofrece conexión tanto con equipos reales como con simulados. La gran ventaja de este simulador es que te permite detectar las limitaciones de tu desarrollo sin poner tu equipo en riesgo.



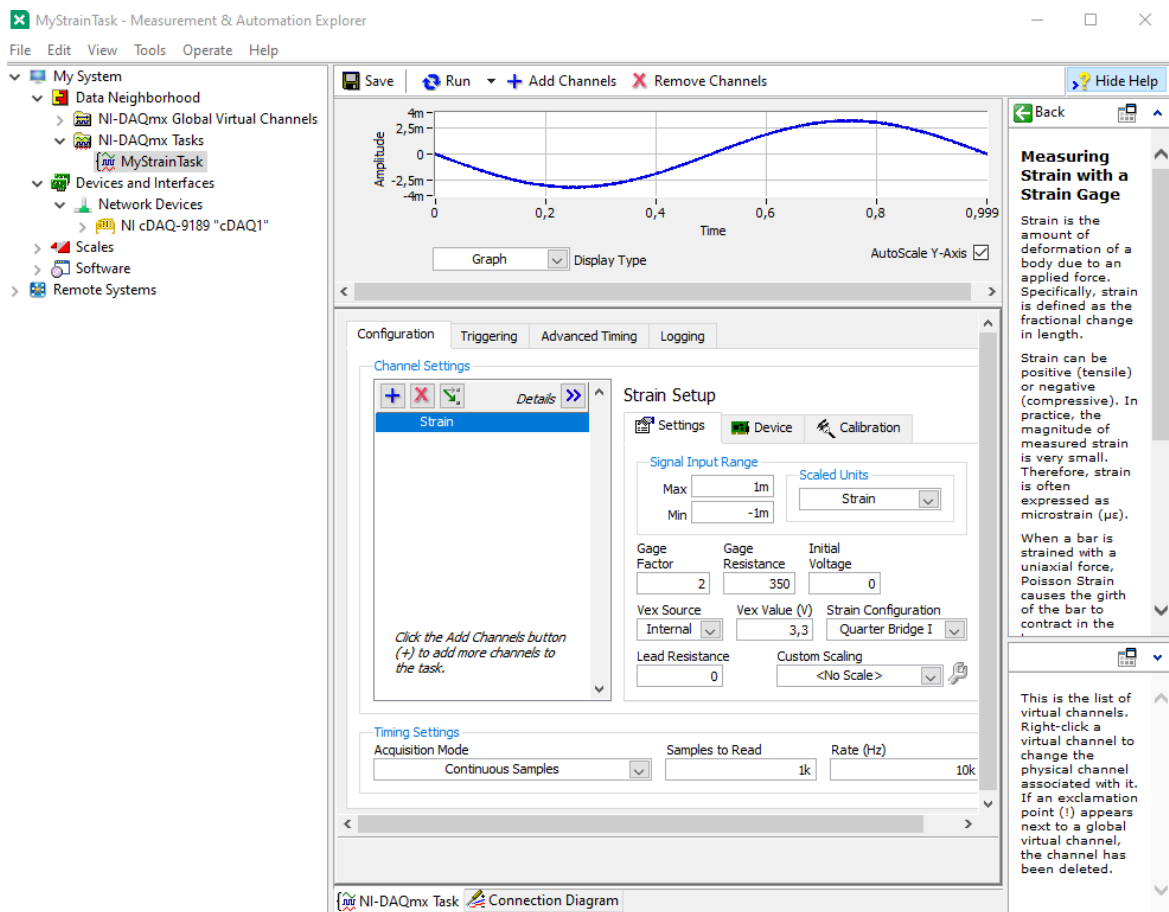


Figura 13 - Dispositivo simulado en NI MAX

Como se puede ver en la Figura 13, configurada correctamente la lectura analógica de la señal será una onda sinodal de periodo de 1 s y amplitud fija. Dadas las similitudes con la forma de onda esperada proveniente del ensayo, esta herramienta es la manera idónea de comprobar el funcionamiento del programa, y por lo tanto es la que se ha empleado.

Como referencia de configuración de adquisición de datos se han empleado los ejemplos proporcionados en 'NI Example Finder' y a partir de ahí se ha desarrollado la función correspondiente a la creación de *tareas*.

Es importante destacar que en todo momento se han dividido las subrutinas del programa según tipología. Esto se debe a que cada módulo hay diferentes parámetros a configurar o los parámetros toman diferentes valores. En la Figura 14 se puede ver cómo se configuran las *tareas* correspondientes a módulos de entrada en corriente (NI 9203).

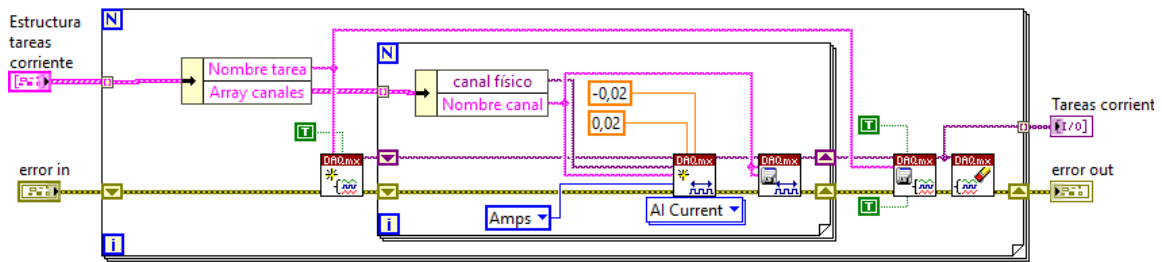


Figura 14 - Diagrama de bloques de creación de tareas de corriente

Se ha empleado una estructura de datos fundamental que se transfiere a lo largo de todas las funciones relacionadas con la adquisición. Ésta abarca las *tareas* creadas y los diferentes canales (Figura 15).

Figura 15 - Estructura de canales y tareas

Antes de poner en marcha la *adquisición*, es necesario programar la temporización asociada a las *tareas*. La función 'Timing' de NI DAQmx permite configurar como Adquisición continua, de esta manera no se pierden muestras. Este modo de configuración asigna automáticamente un 'buffer' lo suficientemente grande para la aplicación que se quiere implementar. [15]



Tras la programación de la *adquisición* y el *procesamiento* se ha añadido al panel de usuario una gráfica que sirve como comprobación del funcionamiento del programa ya que dibuja tanto los datos adquiridos como los máximos y mínimos calculados (Figura 16).

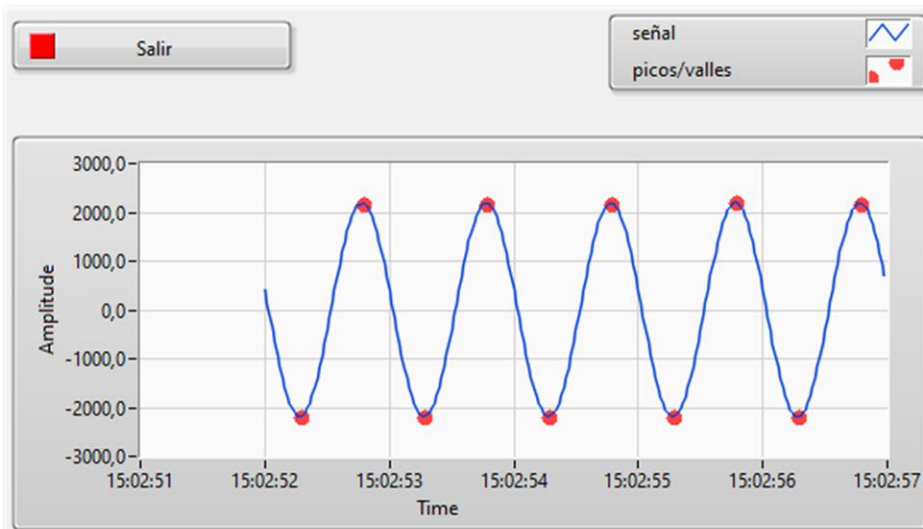


Figura 16 - Gráfica de datos adquiridos y procesados

Al finalizar la aplicación se construye un ejecutable que permitirá emplear el programa desarrollado en cualquier ordenador, y no únicamente en uno con licencia de LabVIEW.



## 5 CONCLUSIONES

**Se ha desarrollado un Software de Adquisición de Datos para poder monitorizar la estructura de las bancadas de ensayo de palas de aerogeneradores.**

El Software desarrollado ofrece funciones específicas adaptadas a los estándares internacionales: Los valores adquiridos reciben un procesamiento de las señales para la obtención de picos y valles y son almacenados en un archivo viario para no rebosar el disco duro. Adicionalmente, es posible guardar los datos obtenidos a la frecuencia de adquisición para que en caso de que se esté produciendo algún defecto poder realizar un análisis más detallado.

La configuración de los ensayos puede realizarse y modificarse de forma intuitiva empleando el Software también.

Específicamente está diseñado para las necesidades que tiene el Laboratorio de Ensayos de Aerogeneradores de CENER. Sin embargo, dada la **versatilidad del software**, es posible emplear el software en toda estructura que se quiera monitorizar a fatiga.

Una de las **dificultades añadidas** de este proyecto ha sido que desde el comienzo el Hardware de adquisición estaba impuesto. Este Hardware tiene unas limitaciones que han condicionado en todo momento el desarrollo del Software. Por ejemplo, un chasis debe controlarse como máximo con dos *tareas*, la frecuencia de adquisición mínima de los módulos es entorno a 800Hz y los módulos no contienen posibilidad de filtrar por hardware la señal entrante.

Otra dificultad añadida ha sido el entorno de programación. Los conocimientos previos al proyecto sobre el lenguaje LabVIEW eran muy básicos y estaban muy lejos de poder realizar una aplicación completa. Además, en el centro donde se ha desarrollado el proyecto, ningún ingeniero tenía los conocimientos suficientes para poder prestar apoyo. Por lo que la fase inicial del proyecto ha estado enfocada a la lectura de documentación y práctica con ayuda de libros prestados de la biblioteca de la UPNA. Gracias a la gran cantidad de documentación y el foro que dispone 'National Instruments', se han podido resolver las dudas que han ido surgiendo.

Por otro lado, en todo momento se ha contado con el **apoyo de técnicos e ingenieros de ensayo especializados** en ensayos de pala de aerogeneradores que han proporcionado unos conocimientos esenciales sobre los mismos.

Finalmente, el proyecto ha seguido su progreso tras los pasos documentados en esta memoria. Al Software se le han realizado una serie de pruebas de validación y se ha instalado el sistema completo en la Nave de Ensayo de Palas a tamaño real que dispone el LEA. Esta segunda fase del proyecto esta descrita en el documento 'Validación e Instalación del Software de un Sistema de Adquisición de Datos para Ensayos de Fatiga de Palas de Aerogeneradores'.



## 6 BIBLIOGRAFÍA

- [1] “• Energía eólica: participación en la generación total de energía en España | Statista.” [En línea]. Disponible en: <https://es.statista.com/estadisticas/1004226/participacion-de-la-generacion-eolica-en-la-generacion-total-en-espana/>. [Accedido: 31-may-2022]
- [2] L. S. Avendaño, “Modelación de aerogenerador con compensación difusa”, tesis doctoral, Cinvestav, 2015.
- [3] J. R. Lajara y J. Pelegrí Sebastián, *LabVIEW : entorno gráfico de programación*. Marcombo, 2017.
- [4] J. del Río Fernández *et. al.*, *LabVIEW : programación para sistemas de instrumentación*, Madrid: Ibergarceta publicaciones SL., 2011.
- [5] National Instruments, “NI-DAQmx Available en Online Documentation - NI”. [En línea]. Disponible en: <https://www.ni.com/es-es/support/documentation/supplemental/06/getting-started-with-ni-daqmx--ni-daqmx-documentation.html>. [Accedido: Feb. 27, 2022]
- [6] B. F. Sørensen, “Improved design of large wind turbine blade of fibre composites based on studies of scale effects (phase 1) - summary report”, 2004.
- [7] H. F. Zhou *et. al.*, “A review of full-scale structural testing of wind turbine blades,” *Renewable and Sustainable Energy Reviews*, vol. 33, pp. 177–187, may 2014, doi: 10.1016/J.RSER.2014.01.087.
- [8] *Aerogeneradores – Parte 23: Ensayos estructurales de palas a escala real*, IEC TS 61400-23, 2014.
- [9] “Laboratorio de Ensayo de Palas - CENER - Centro Nacional de Energías Renovables.” [En línea]. Disponible en: <https://www.cener.com/areas-de-investigacion/energia-eolica/areas-de-actuacion-en-energia-eolica/laboratorio-de-ensayo-de-aerogeneradores-lea/laboratorio-de-ensayo-de-palas/>. [Accedido: May 21, 2022]
- [10] J. C. Middleton, “Strain gauge performance during fatigue testing,” *Strain*, vol. 22, no. 3, pp. 135–140, ago. 1986, doi: 10.1111/J.1475-1305.1986.TB00608.X.
- [11] “Fundamentos sobre galgas extensométricas - HBM”. [En línea]. Disponible en: <https://www.hbm.com/es/7074/fundamentos-sobre-galgas-extensometricas/>. [Accedido: 30-abr- 2022].
- [12] M.-M. Vishay Precision Group, “Noise Control in Strain Gage Measurements”, 2013. [En línea]. Disponible en: [www.micro-measurements.com](http://www.micro-measurements.com). [Accedido: 22-abr- 2022].
- [13] “Complete Guide to Building a Measurement System - Academia.edu.” [En línea]. Disponible en:



[https://www.academia.edu/29694925/Complete\\_Guide\\_to\\_Building\\_a\\_Measurement\\_System](https://www.academia.edu/29694925/Complete_Guide_to_Building_a_Measurement_System). [Accedido: Apr. 01, 2022]

- [14] “Designing Distributed TSN Ethernet-Based Measurement Systems - NI.” [En línea]. Disponible en: <https://www.ni.com/es-es/innovations/white-papers/17/designing-distributed-tsn-ethernet-based-measurement-systems.html>. [Accedido: May 31, 2022]
- [15] J. Hnidka and D. Rozehnal, “Strain Gauge Measurement System with Wi-Fi Data Transfer in LabVIEW,” in *2017 International Conference on Military Technologies (ICMT)*, 2017, pp. 520–525, doi: 10.1109/MILTECHS.2017.7988813.



## 7 ANEXO

1	MANUAL DE USUARIO.....	31
1.1	Introducción .....	31
1.2	Descripción del programa .....	31
1.2.1	Funciones principales.....	31
1.2.2	Compatibilidad de hardware.....	31
1.3	Requisitos y limitaciones.....	32
1.3.1	Instalaciones previas .....	32
1.3.2	Conexión hardware .....	32
1.4	Puesta en marcha y utilización.....	33
1.4.1	Puesta en marcha.....	33
1.4.2	Utilización.....	35
1.5	Funcionamiento .....	37
1.5.1	Configuración .....	37
1.5.2	Adquisición .....	40
1.5.3	Lectura de archivos de datos.....	44
1.6	Soluciones a errores más frecuentes .....	45
1.6.1	Errores en la puesta en marcha .....	45
1.6.2	Error de creación de <i>tareas</i> .....	45
1.6.3	Error de guardado .....	47
2	MANUAL DE PROGRAMACIÓN.....	48
2.1	Introducción .....	48
2.2	Herramientas principales del programa.....	49
2.2.1	Bucles anidados.....	50
2.2.2	Bucles ejecutados paralelamente .....	50
2.2.3	Estructura de datos y tareas .....	51
2.3	Diagramas de flujo del programa.....	52
2.3.1	Función principal .....	52
2.3.2	Funciones de configuración .....	53
2.3.3	Funciones de adquisición .....	58
2.4	Modificaciones en el programa.....	62
2.4.1	Aumentar número de canales admitidos.....	62



2.4.2	Modificar frecuencia de muestreo .....	63
2.4.3	Modificar el formato del archivo de guardado .....	63
2.4.4	Añadir un nuevo tipo de módulo .....	65





## 1 MANUAL DE USUARIO

### 1.1 Introducción

Este manual describe el funcionamiento del programa en LabVIEW 'Aplicación.exe'. Su contenido incluye los diferentes pasos a seguir para conectar, configurar y poner en marcha la adquisición y almacenamiento de datos a partir del ejecutable.

La estructura de este documento es la siguiente:

- Descripción del programa
- Requisitos y limitaciones
- Puesta en marcha y utilización
- Funcionamiento
- Soluciones a los errores más comunes

En caso de tener dudas más concretas sobre el desarrollo del programa, ver el Manual de Programación donde, mediante diagramas de flujo, se detallan las diferentes funciones que forman el software.

### 1.2 Descripción del programa

Este programa está diseñado para la adquisición de datos para ensayos de fatiga de palas de Aerogenerador. Incluye el procesado en tiempo real de datos de acuerdo con los requerimientos de la norma internacional y permite, además, configurar hasta 250 canales, los cuales pueden ser tanto de galgas extensiométricas como de voltaje o corriente.

#### 1.2.1 Funciones principales

- Configuración de los distintos canales
  - Crear archivo de configuración
  - Modificar un archivo de configuración ya existente
- Cerado de las galgas
- Procesado de los datos adquiridos
- Guardado binario

#### 1.2.2 Compatibilidad de hardware

- Chasis permitidos:
  - NI cDAQ 9189
  - NI cDAQ 9185
- Módulos permitidos:
  - Canales de galgas extensiométricas
    - Módulo NI 9236
  - Canales de corriente
    - Módulo NI 9203
  - Canales de voltaje
    - Módulo NI 9205



### 1.3 Requisitos y limitaciones

Para poder ejecutar la aplicación y que funcione de manera correcta hay algunas restricciones que hay que tener en cuenta:

#### 1.3.1 Instalaciones previas

Este programa está realizado mediante el software LabVIEW y para ejecutar la aplicación es necesario tener instalado 'LabVIEW Run-Time Engine'. Además, para que el ejecutable funcione de manera correcta, es necesario que el 'Run Time Engine' sea compatible con la versión de LabVIEW donde se ha desarrollado la aplicación. Para ello se necesita al menos instalar la versión 2021. Se puede descargar en el siguiente enlace: [Descargar LabVIEW Runtime - NI](#)

Además, con el objetivo de realizar una adquisición optimizada, se va a emplear el controlador de instrumentos NI DAQmx, que deberá estar correctamente instalado para su funcionamiento. ([Descargar NI DAQmx - NI](#))

Las conexiones con el hardware se realizarán a través de NI Measurement & Automation Explorer (MAX). No se trata de un software independiente, si no que cuando se descarga el driver NI DAQmx, se instalará automáticamente.

Los archivos de datos generados por la aplicación van a tener formato binario para que ocupen el mínimo espacio en el disco duro. La lectura correcta de estos archivos se realizará empleando la función 'read.m' realizada en MATLAB®.

#### Resumen de los programas necesarios

- LabVIEW Run-Time Engine
- NI MAX
- Driver NI-DAQmx
- MATLAB® (cuando se quieran leer los archivos generados)

#### 1.3.2 Conexión hardware

La compatibilidad hardware de este programa es únicamente válida para chasis Ethernet, tanto cDAQ-9189 con 8 ranuras para módulos de entrada y/o salida como 9185, con 4 ranuras. Los diferentes módulos que se pueden insertar en las ranuras sin ninguna modificación son los siguientes:

- NI 9236: Este módulo permite la lectura de 8 canales de galgas extensiométricas a cuarto puente 350Ω.
- NI 9203: Este módulo permite conectar 8 canales de entrada en corriente con un rango de medición de ±20mA.
- NI 9205: Este módulo permite conectar 15 canales de entrada en vataje diferencial con un rango de ±10V.

En el caso de querer emplear otro tipo de módulo, debemos consultar el 'Manual de Programación'. La programación está hecha de manera que se puedan incluir nuevos tipos de



módulos con facilidad, siempre que se tenga un conocimiento básico de LabVIEW y se sigan los pasos indicados en el manual.

Una de las ventajas que tiene el equipo empleado en este sistema es la capacidad modular. A conveniencia del usuario se pueden conectar en serie tantos chasis como uno quiera. La única limitación del programa es que la configuración actual permite hasta un total de 250 canales. En caso de necesitar más consultaremos nuevamente el 'Manual de Programación' para modificar esta restricción.

## 1.4 Puesta en marcha y utilización

El procedimiento a seguir para que el programa funcione de forma correcta.

### 1.4.1 Puesta en marcha

1. Con el armario **desconectado de la alimentación**, se insertan todos los módulos que se van a emplear en las ranuras de los chasis y se conectan los terminales de los módulos a los diferentes sensores.
  - Para desconectar el armario de la alimentación hay varias opciones: desenchufar el cable de alimentación, hacer saltar el magnetotérmico o hacer saltar el diferencial.
  - Insertar los módulos en los chasis presionando las pestañas superior e inferior. Únicamente se podrán introducir dos tipos de módulos en cada chasis debido a limitaciones del Software.
  - Conectar los bornes de cada módulo con el transductor correspondiente
    - NI 9236: Se conecta en cada canal el terminal EX a un extremo de la galga y los terminales AI y RC al otro extremo.
    - NI 9203: Se conecta directamente el terminal de cada canal y las medidas serán respecto al terminal común que está internamente conectado a la referencia de tierra aislada del módulo.
    - NI 9205: La conexión con señales diferenciales se realiza conectando el terminal positivo a lado izquierdo de los bornes del módulo y el terminal negativo al lado derecho. Por ejemplo, V+ -> ACH0 y V- ->ACH8.
2. Con el objetivo de sincronizar las señales adquiridas, todos los **chasis** que se vayan a usar simultáneamente deberán estar **conectados en cascada** mediante cables Ethernet.



Figura 17 - Conexión de los cables Ethernet entre chasis



3. Una vez garantizado que todas las conexiones están realizadas, conecta la alimentación del equipo y asegúrate de que **NI MAX detecta todos los chasis y módulos**.

En la Figura 18 se observa NI MAX, el hardware detectado en este caso es el chasis 'cDAQ6' y el módulo 'cDAQ6Mod1'. Sin embargo, el módulo 'cDAQ6Mod2' no establece conexión, por lo que no podrá ser empleado en la adquisición.

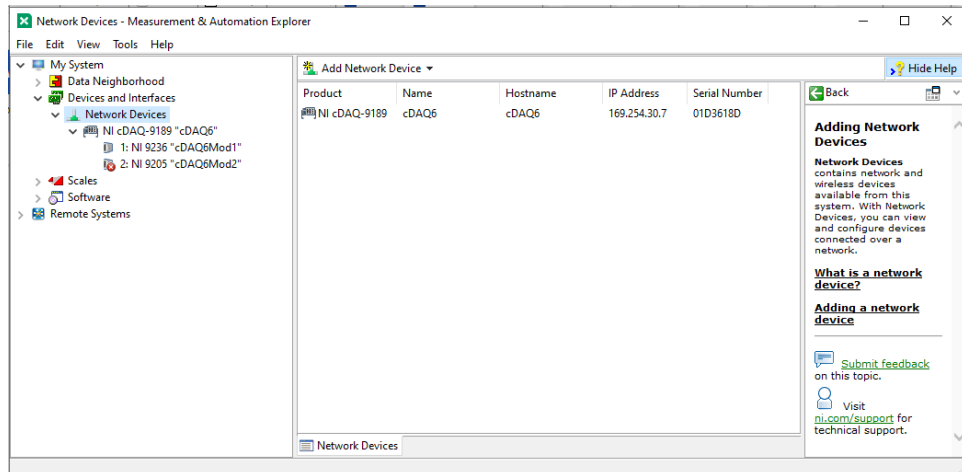


Figura 18 - Conexión con Hardware empleando NI MAX

- Para alimentar el equipo hay que conectarlo a una alimentación monofásica de 230V y asegurarse de que no están manualmente desactivados tanto el interruptor magneto térmico como el diferencial.
- Se abre la aplicación de NI MAX. Si el equipo está alimentado y conectado al ordenador mediante cable Ethernet, en el apartado de dispositivos de red se detectará automáticamente los chasis que estén en tu red. En el caso de que no se detecte automáticamente revisar la conexión.

#### Primera conexión:

Si es la primera vez que se emplea el equipo en el ordenador, el icono del dispositivo aparecerá incoloro, tal y como se ve en la Figura 19. Habrá que añadirlo haciendo clic en 'Add Device'.

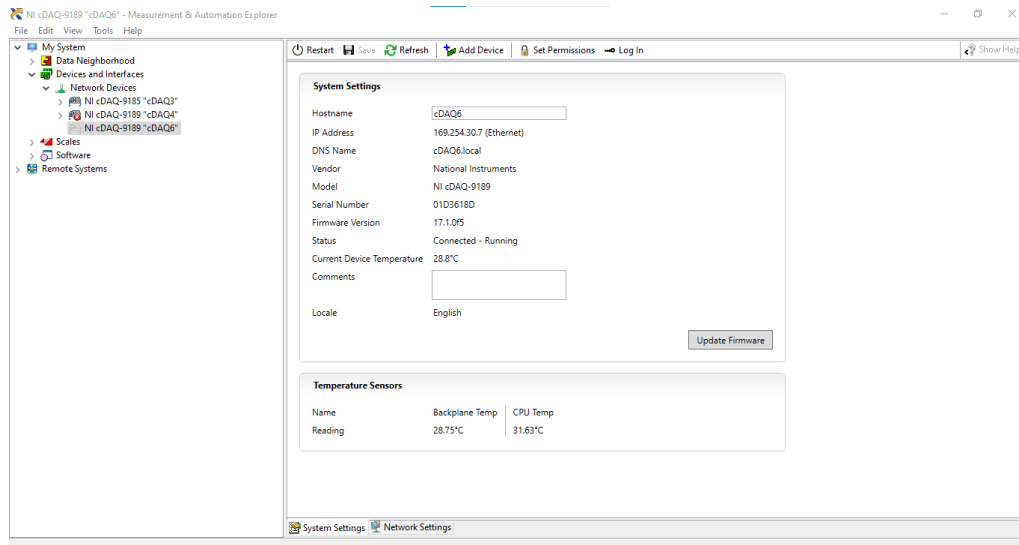


Figura 19 - Primera conexión con Hardware

Es posible en este caso que el dispositivo esté reservado por el PC que se haya empleado previamente. Aparecerá un error como el mostrado en la Figura 20, y para solucionarlo bastará con clicar en 'Reserve Network Device' y aceptar la ventana emergente.

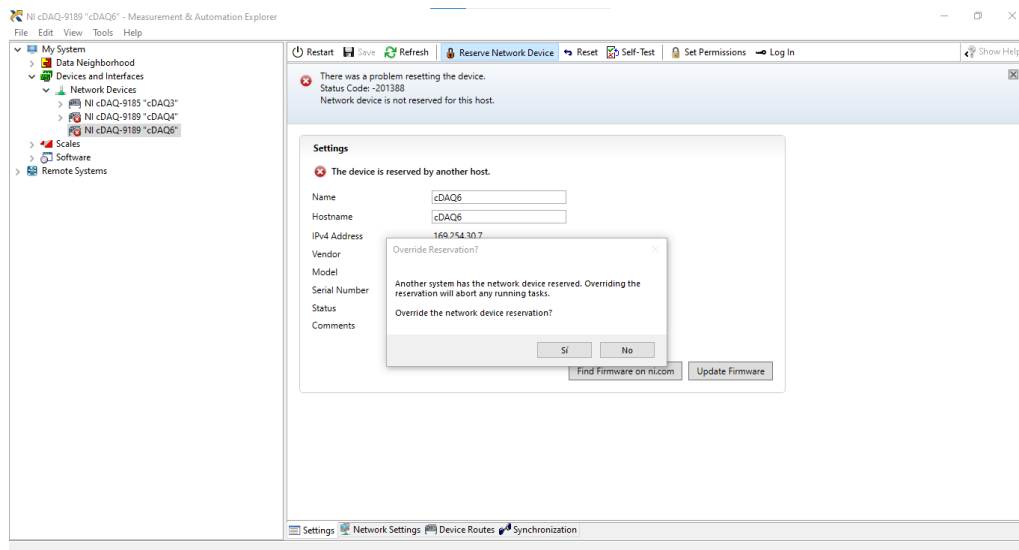


Figura 20 - Error en primera conexión con hardware

### 1.4.2 Utilización

Para ejecutar la aplicación se debe tener en una carpeta los siguientes elementos:

- Una carpeta de archivos con el nombre 'Config': En esta carpeta se guardan los archivos de configuración que se crean o modifican con la aplicación.
- Dos archivos auxiliares a la aplicación 'Application.aliases' y 'Application'.
- El ejecutable: Para entrar en el programa basta con hacer doble clic en este icono.

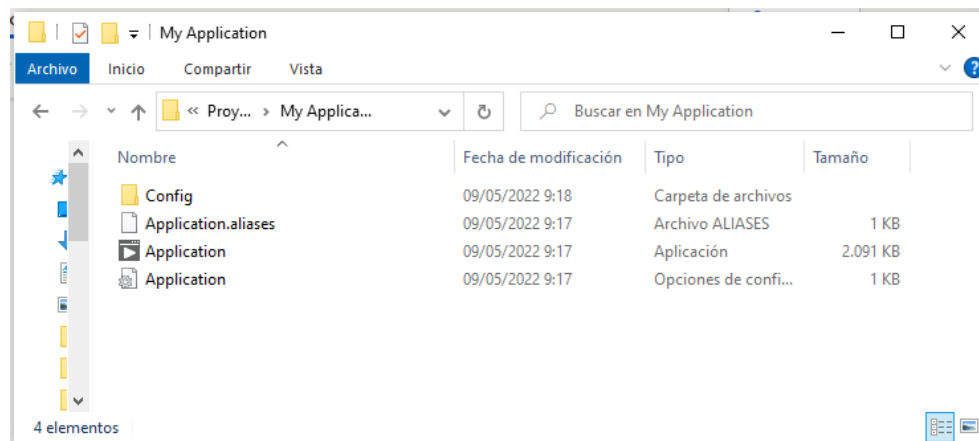


Figura 21 - Carpeta con los archivos para ejecutar el programa

Al entrar en el ejecutable se abrirá la pantalla principal del programa (Figura 22). Ésta permite acceder a la pantalla de configuración, para poder crear nuevos archivos de configuración o modificar uno ya existente. Esta funcionalidad está explicada con profundidad en el apartado 'Configuración'. En caso de que ya se tenga creado un archivo de configuración se procederá a la adquisición accediendo a la ventana de 'iniciar', explicada en el apartado 'Adquisición'.

Para poder moverse entre ventanas, basta con pulsar en los 'botones' que aparecen en las ventanas. En caso de querer volver atrás, nunca cerrar o minimizar la ventana de la aplicación, sino que todas las pantallas tendrán un 'botón' de salida que cerrará la última pantalla y volverá a la anterior. Si nos encontramos en la pantalla principal, el botón de parada cerrará la aplicación.

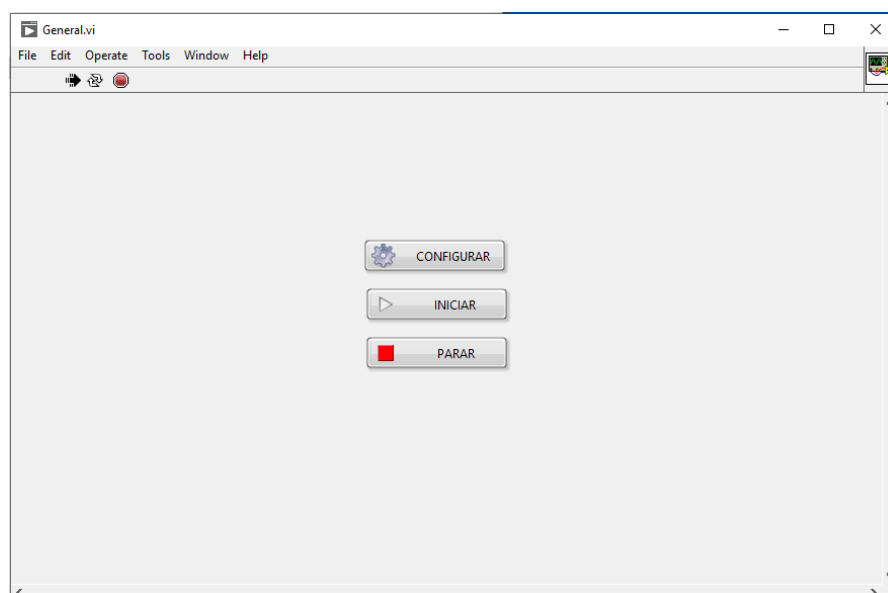


Figura 22 - Pantalla de inicio del programa

Los iconos de la parte superior izquierda indican el estado del programa. Si la flecha es de color negro, quiere decir que el programa se está ejecutando, lo que ocurre por defecto al abrir

el ejecutable. Si el programa no se está ejecutando la flecha será de color blanco. Se puede realizar una parada forzada del programa en caso de bloqueo de alguna pantalla, para ello basta con pulsar el círculo rojo. No se recomienda emplear esa opción a no ser que no quede otro remedio.

## 1.5 Funcionamiento

A partir de la pantalla principal de la Figura 22 debes escoger una de las dos opciones.

### 1.5.1 Configuración

Si se quiere emplear un archivo de configuración diferente al empleado en los anteriores ensayos, se puede modificar una configuración existente o crear una desde cero.

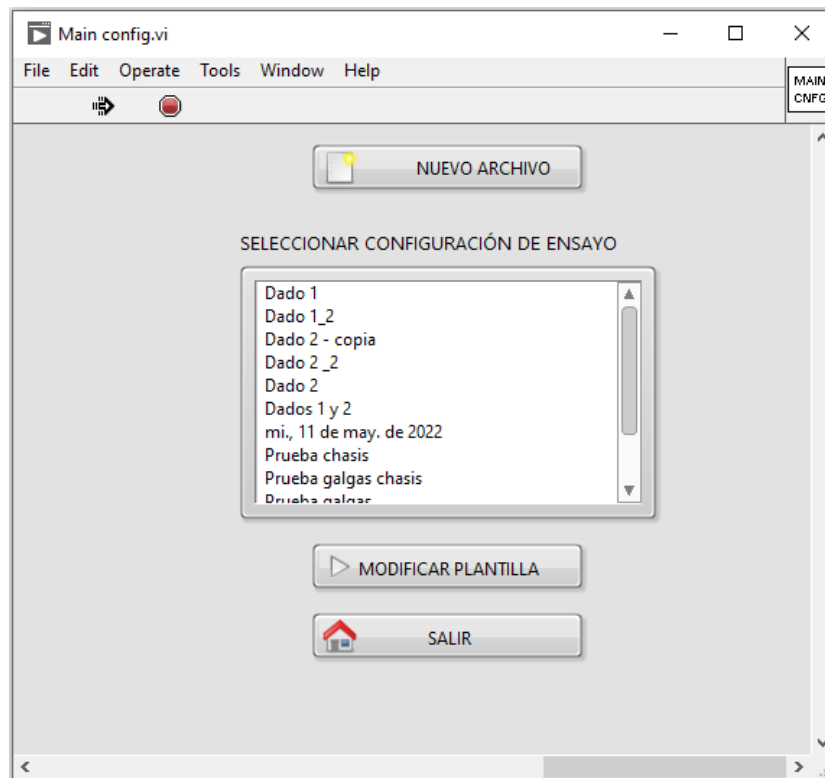


Figura 23 - Pantalla de configuración

Para crear la configuración desde cero se pulsa en '**Nuevo archivo**' y se abre la pantalla de la Figura 24.

Los diferentes campos que hay que rellenar son los siguientes:

- **NOMBRE ARCHIVO:** Se debe asignar un nombre de archivo, en caso contrario aparecerá una ventana emergente que indica que el nombre del archivo creado será la fecha actual.
- **IDENTIFICADOR ENSAYO:** Texto de apoyo para distinguir qué configuración es.
- **FRECUENCIA DE ADQUISICIÓN.** Frecuencia a la que se adquirirán los datos temporales. Los límites de esta frecuencia son [0-200 Hz].



- FRECUENCIA DE ENSAYO: Frecuencia a la que el actuador del ensayo va a funcionar. Esta frecuencia marcará el procesado de datos. La frecuencia debe ser inferior a la frecuencia de adquisición.
- NÚMERO DE CICLOS POR ARCHIVO: Cantidad de ciclos que se almacenarán en cada archivo de datos.
- CONFIGURACIÓN DE CANALES: Canal por canal se tiene que ir rellenando los diferentes campos, para rellenar más canales mover la barra de desplazamiento inferior hacia la derecha.
  - Tipo de canal: Elegir entre galga extensiométrica, canal de corriente o de voltaje.
  - NChasis/NModulo/Ncanal: Localización del canal en el armario de adquisición. La indexación de los chasis y los módulos comienza en 1 y la de los canales en 0. En caso de dudas consultar en NI MAX los chasis y módulos que están conectados. En los canales de voltaje, al tratarse de señales diferenciales, los canales 8-15 y 24-32 están reservados para el terminal negativo.
  - Nombre: El nombre del canal. No puede tener caracteres especiales, únicamente letras, números y barra baja ('\_').
  - FG/Ganancia: En el caso de las galgas extensiométricas el factor de galga, en el resto de los canales la *ganancia* que se les quiera aplicar.
  - Offset: La compensación necesaria.
  - Unidades: Cada tipo de canal realizará la medición en unas unidades: Las canales de extensimetría en ' $\mu$ Strain', los canales de voltaje en Voltios y los canales de corriente en miliAmperes.





**Nueva configuración**

Nombre archivo

IdentificadorEnsayo

FrecuenciaAdquisición

FrecuenciaEnsayo

N°Ciclos/Archivo

0

Tipo de canal	Tipo de canal	Tipo de canal
Extensimetría	Corriente	Voltaje
NChasis <input type="text" value="0"/>	NChasis <input type="text" value="0"/>	NChasis <input type="text" value="0"/>
NModulo <input type="text" value="0"/>	NModulo <input type="text" value="0"/>	NModulo <input type="text" value="0"/>
NCanal <input type="text" value="0"/>	NCanal <input type="text" value="0"/>	NCanal <input type="text" value="0"/>
Nombre <input type="text"/>	Nombre <input type="text"/>	Nombre <input type="text"/>
FG/Ganancia <input type="text" value="0"/>	FG/Ganancia <input type="text" value="0"/>	FG/Ganancia <input type="text" value="0"/>
Offset <input type="text" value="0"/>	Offset <input type="text" value="0"/>	Offset <input type="text" value="0"/>
Unidades uStrain	Unidades mA	Unidades V
Enabled <input checked="" type="checkbox"/>	Enabled <input checked="" type="checkbox"/>	Enabled <input checked="" type="checkbox"/>

ACEPTAR CONFIGURACIÓN

Figura 24 - Pantalla para crear una configuración

Tras rellenar los campos correctamente se acepta la configuración.

En el caso de querer realizar cambios a alguna configuración, seleccionar en la lista un archivo y pulsar '**Modificar plantilla**'. En este caso se importarán todos los campos del archivo y se podrán realizar cambios. (Figura 25)



Figura 25 - Pantalla para modificar la configuración

Cuando se hayan realizado los cambios necesarios, clicar 'Aceptar configuración' y se volverá a la pantalla general de configuración.

Cuando se crea o se modifica un archivo de configuración no es necesario que se siga el orden del hardware. Es decir, no hay que añadir consecutivamente los mismos canales del mismo módulo, o del mismo chasis. Sin embargo, es **necesario que los tipos de canales estén ordenados**. Los primeros en añadir tienen que ser las galgas extensiométricas; a continuación, los de corriente y finalmente los de voltaje.

### 1.5.2 Adquisición

Una vez dispones de un archivo con la configuración correcta, se accede a la adquisición pulsando 'Iniciar' en la pantalla general. Aquí se selecciona el archivo de configuración y se procede con el botón 'Iniciar'. (Figura 26)

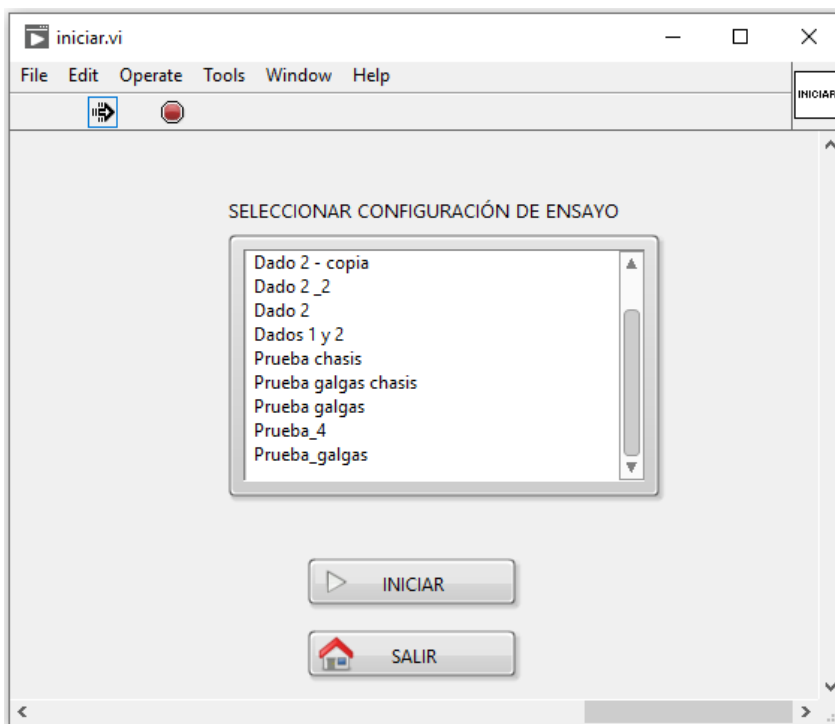


Figura 26 - Pantalla de acceso a la adquisición

Llegados a este punto, el programa carga la configuración y crea las *tareas* necesarias para realizar la *adquisición*. Mientras esto ocurre, aparecerá una pantalla de transición. (Figura 27)



Figura 27 - Pantalla de espera durante la creación de tareas

Una vez se ha importado la configuración correctamente, aparecerá una ventana emergente para que se introduzca la ruta a la carpeta de almacenamiento deseada. Por defecto se abre la ruta 'C:\Users\Cener\Documents\Datos\_ensayos'. Para seleccionar la carpeta de destino hay que situarse en ella y clicar en 'Current Folder'.

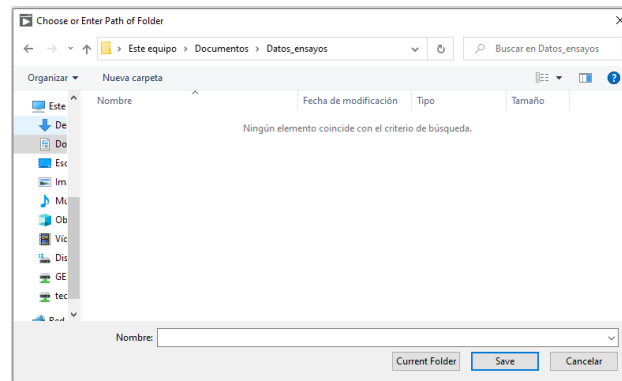


Figura 28 - Ventana emergente para configurar la ruta de guardado

Si en lugar de la ventana emergente de la Figura 28 aparece un mensaje de error, posiblemente será porque las *tareas* no se han creado. Ver el apartado 'Error de creación de *tareas*' para encontrar su origen y posibles soluciones.

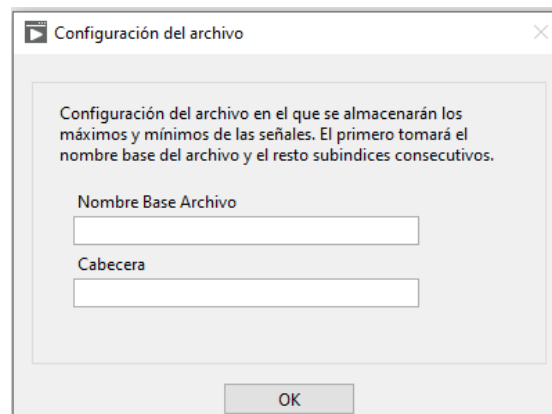


Figura 29 - Ventana emergente para configurar el archivo de guardado

Una vez seleccionada la carpeta aparecerá la ventana emergente de la Figura 29. Aquí se indica el *nombre base* que tendrán los archivos binarios generados con los datos. En el apartado 'Cabecera' se puede añadir un texto explicativo o si se desea puede quedar vacío. Si al aceptar la configuración aparece un mensaje de error ver el apartado 'Error de guardado'.

Una vez configurado el ensayo correctamente aparecerá la pantalla de la Figura 30. Esta pantalla permite cerrar las galgas, para que las oscilaciones estén centradas en cero. Esta opción se activa mediante el interruptor, que si está activado encenderá una luz amarilla y si está desactivado tomará el aspecto de la Figura 30.

Cada vez que se para el ensayo se volverá a esta pantalla, y se podrá volver a arrancar de nuevo con la misma configuración tantas veces como se desee, siempre y cuando no se pulse el botón de 'salir' para volver a pantallas anteriores. Los archivos de datos generados seguirán con el mismo *nombre base* y subíndices consecutivos.

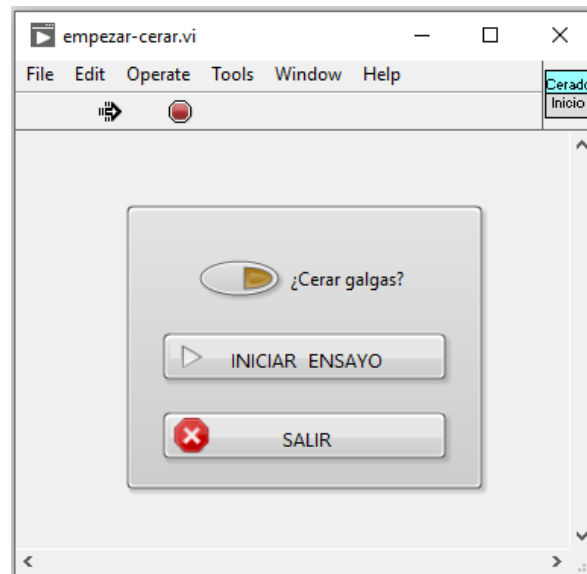


Figura 30 - Pantalla de inicio de adquisición

Al pulsar el botón 'Iniciar ensayo' comienza la *adquisición* y aparece la pantalla de la Figura 31. En esta pantalla se visualiza la señal temporal junto con los picos y valles de varios ciclos. De esta manera nos aseguramos de que estamos adquiriendo. Únicamente se mostrará un canal para tener mayor claridad de imagen. Si se quiere visualizar otro canal conectado, se puede emplear el selector junto al gráfico.

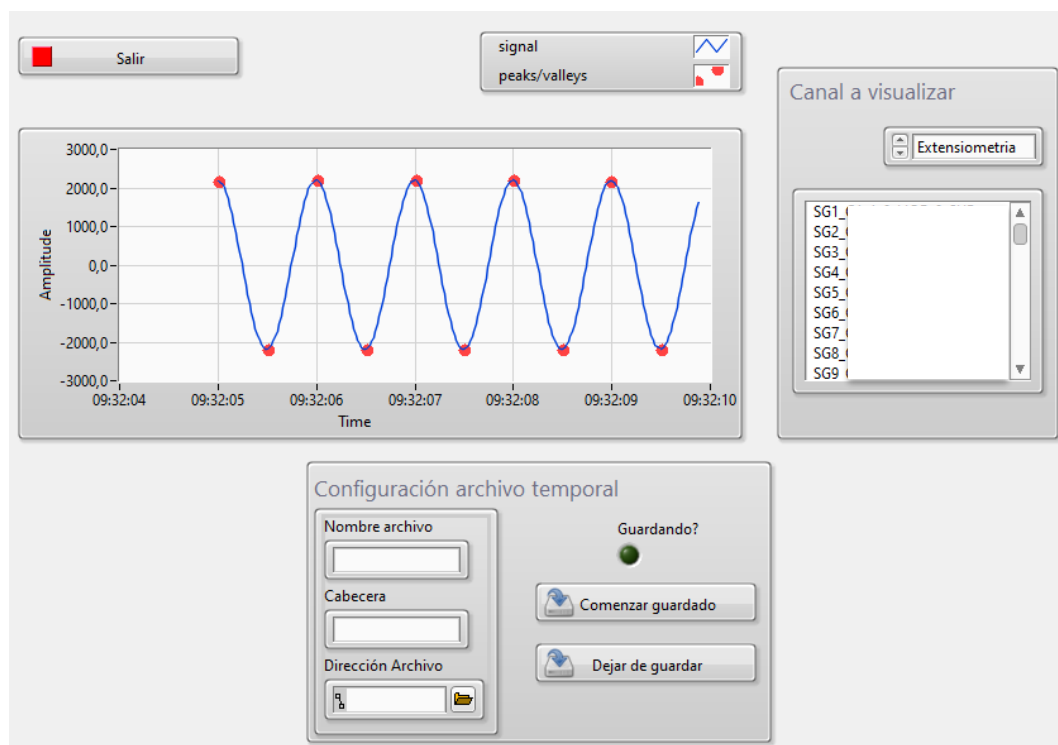


Figura 31 - Pantalla durante la adquisición

En caso de en un momento dado querer almacenar los datos adquiridos, se emplea el cuadro inferior. Donde se indican los parámetros de nombre de archivo, cabecera (si se quiere



añadir algún comentario) y ruta de la carpeta donde se quiere guardar el archivo. Cuando se ha configurado el archivo se pulsa el botón ‘comenzar guardado’ y se encenderá el led que indica que se están almacenando los datos. Para dejar de guardar se pulas el otro botón y el led se apagará.

Para salir de la *adquisición* se emplea ‘Salir’. Volverás a la pantalla anterior. Si se ha terminado el ensayo definitivamente, se recomienda seguir cerrando pantallas pulsando los botones de salida. En el caso de que se quiera continuar adquiriendo se vuelve a pulsar ‘iniciar ensayo’.

En el caso de que no se haya dejado de guardar el archivo de valores a frecuencia de adquisición se cerrará y no se habrá perdido ningún dato. En cuanto el archivo binario con los datos de picos y valles, se guardarán todos los valores que se hayan ido acumulando pese a no llegar al número de ciclos por archivo, de esta manera no se perderán datos.

### 1.5.3 Lectura de archivos de datos

Los archivos de datos generados estarán en formato binario para ocupar el mínimo espacio posible en el disco duro. Para poder leerlos, se ha realizado una función en Matlab ‘read.m’. No es necesario tener instalado en el PC que se va a emplear para la *adquisición* Matlab, sino que, una vez que se termine el ensayo se transfieren los archivos a un ordenador con licencia y se realiza ahí la lectura de los archivos.

La función genera dos estructuras de datos: La estructura 1 contiene 8 campos genéricos sobre el ensayo. En cambio, la estructura 2 contiene un vector de resultados cada una con 12 campos. Los primeros dos elementos del vector corresponden al tiempo de adquisición y el resto a los picos y valles de los distintos canales.

**Ejemplo** de los datos obtenidos en la ventana de resultados de Matlab:

```
[Estructura_1, Estructura_2]=read_binary_function('Test_0.bin')
```

*Estructura\_1 =*

*struct with fields*

*filename: 'Test\_0.bin'*

*fileid: 5012* - Identificación del archivo, si se ha generado con la aplicación siempre será 5012

*dataoffset: 708* - Posición del archivo en la que se empiezan a escribir los datos de los canales

*comment: 'Texto que quieras que aparezca'* - Texto escrito en cabecera

*ReservedSTR: [32x256 char]-> 'C:\Users\aperez\Documents\Datos\_ensayos\Test\_0.bin'* -

Ruta

*noofchan: 6* - Cantidad de canales almacenados, los de tiempos, los picos y los valles

*mcl: 0*

*redufact: 0*

*Estructura\_2(1)*

=

*struct with fields:*

*Channelnumber: 0* - Número del canal en el archivo

*Channellength: 10* - Cantidad de datos en el canal



*ChannelName: 'Abs\_time'* - Nombre del canal  
*Unit: ''* - Unidades del canal, en este caso está en blanco porque se trata de un canal de tiempos absolutos

*comment: 'Tiempo de ciclo de acuerdo con reloj interno'* - Descripción del canal

*format: 0*

*datawidth: 8*

*datumzeit: 3.7351e+09*

*header: ''* - Cabecera del canal

*linmode: 0*

*userscale: 1*

*data: [4.6351e+04 4.6351e+04 4.6351e+04 4.6351e+04 4.6351e+04 4.6351e+04 4.6351e+04 4.6351e+04 4.6351e+04 4.6351e+04]*

La propia función de 'Matlab' contiene comentarios para indicar los pasos que realiza. En caso de dudas consultarlos.

## 1.6 Soluciones a errores más frecuentes

### 1.6.1 Errores en la puesta en marcha

Es posible que al detectar el Hardware en NI MAX el chasis se conecte automáticamente, sin embargo los módulos no y aparecerán con una cruz roja. En este caso basta con pulsar encima del módulo y la cruz roja desaparecerá. Si no lo hace revisaremos el módulo, porque puede que no esté colocado correctamente.

### 1.6.2 Error de creación de tareas

Después de importar la configuración, las *tareas* necesarias para la *adquisición* no se han podido crear y, por lo tanto, no se puede proceder. Este error está detectado por el software y permite ser corregido volviendo a pantallas anteriores de la aplicación.

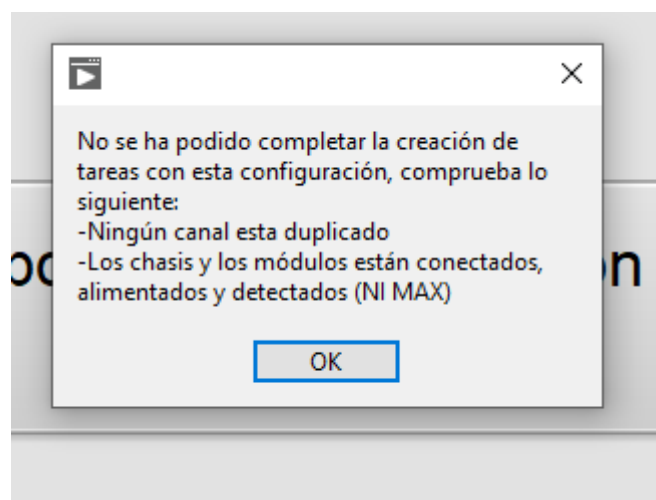


Figura 32 - Error de creación de tareas

Posibles **causas** para este error:

- EL ARCHIVO DE CONFIGURACIÓN CONTIENE ERRORES



Al crear o modificar el archivo de configuración se ha asignado el mismo canal físico (mismo NChasis, NModulo y NCanal) o se les ha asignado a dos canales el mismo nombre.

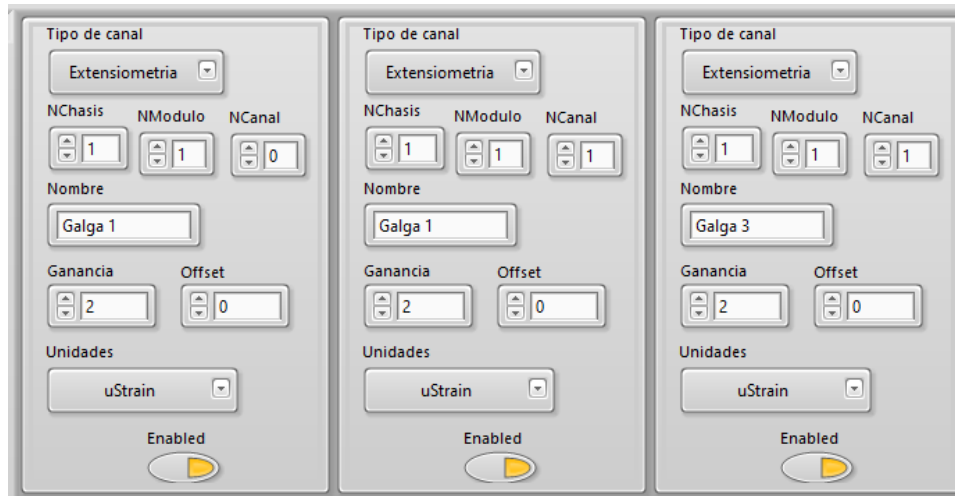


Figura 33 - Configuración de canales errónea

Como se puede observar en la Figura 33, el primer y segundo canal tienen asignado el mismo nombre. El segundo y el tercer canal, en cambio, tienen nombres diferentes aunque están ambos asignados a cDAQ1Mod1/ai1. Cualquiera de los dos casos causará error en la creación de *tareas*.

Además, el nombre del canal no puede contener ningún signo de puntuación o caracteres especiales ya que la mayoría de ellos no los reconoce el programa en su ejecución. Se recomienda emplear únicamente letras, números o barra baja para asegurarnos su buen funcionamiento y evitar errores.

- MÓDULO NO DETECTADO EN NI MAX:

Otra posibilidad es que el archivo de configuración contenga un canal en un módulo no detectado por NI MAX. Si se está empleando una configuración antigua, puede que ese módulo se haya reubicado por necesidades del sistema, por lo que habrá que intentar localizar la nueva posición del canal que se quiere adquirir. Revisar si no el apartado de 'puesta en marcha' de este manual.

- MÓDULOS CONECTADOS INCORRECTAMENTE:

Únicamente será posible emplear dos tipos de módulos por cada chasis. Es decir, en cada chasis se podrán medir únicamente dos tipos de señales. Teniendo en cuenta que el objetivo general son las señales provenientes de galgas extensiométricas esto no supondrá un gran inconveniente. Una posible manera de evitar esta restricción es incluir distribuidos los módulos de señales auxiliares en los chasis dedicados a las galgas. Esto es posible debido a que los módulos son intercambiables entre los diferentes chasis siempre y cuando esté bien especificado en el archivo de configuración.





### 1.6.3 Error de guardado

Una vez se ha iniciado la adquisición, puede que el guardado de los archivos binarios no sea posible, con lo que aparecerá el siguiente error.

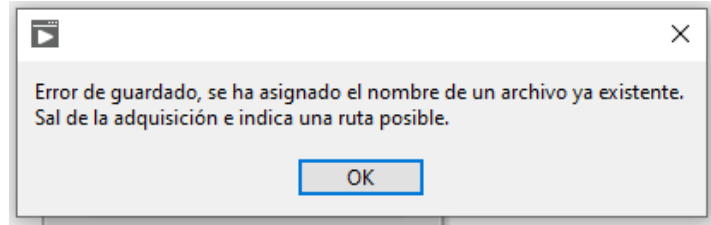


Figura 34 - Error de guardado

#### NOMBRE BASE DEL ARCHIVO DUPLICADO

En la carpeta seleccionada como ruta de guardado de los archivos binarios, no puede haber archivos con el mismo *nombre base*. Si no, no se crearán los archivos ya que al salir de la configuración los subíndices se actualizan y esta aplicación no permite sobrescribir archivos para no perder datos. Por lo tanto, si se introduce un nombre no válido como 'Nombre de archivo base' (Figura 29), el programa desechará toda la configuración y volverá a la 'pantalla de inicio de adquisición'.

Para evitar este error, basta con seguir los mismos pasos e indicar un nuevo nombre de archivo base o elegir una nueva carpeta para almacenar los archivos.

## 2 MANUAL DE PROGRAMACIÓN

### 2.1 Introducción

Este manual contiene una explicación del Software de adquisición de datos para ensayos de fatiga de palas de aerogeneradores. Antes de leerlo se recomienda tener un conocimiento básico sobre LabVIEW. Si nunca antes se ha trabajado con LabVIEW, ver el manual [Getting Started with LabVIEW - National Instruments \(ni.com\)](http://www.ni.com). Éste permite adquirir las bases del lenguaje de programación LabVIEW y del driver NI DAQmx que también se ha utilizado al desarrollar el sistema.

Para poder acceder al entorno de programación es necesario tener instalados tanto el software LabVIEW como el driver de instrumentación NI-DAQmx. El programa está contenido en un proyecto desde donde se puede navegar a través de los distintos 'Virtual Instruments' (VIs).

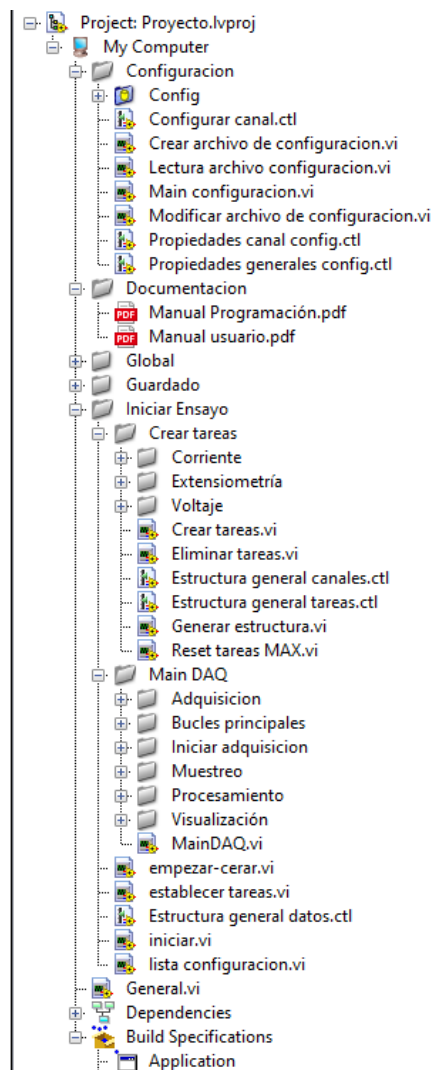


Figura 35 - Organización del proyecto



El programa consta de diferentes funcionalidades entre las cuales se navega de la siguiente manera:

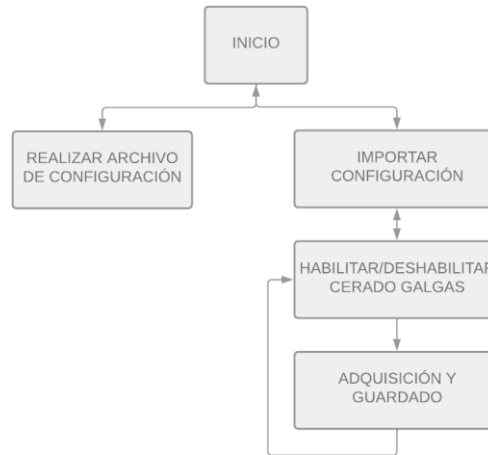


Figura 36 - Navegación entre las funcionalidades del programa

Como se puede apreciar en la Figura 36, al entrar en General.vi, se inicia el programa y se puede acceder tanto al **menú de configuración** como al **de adquisición**.

Si se accede al menú de configuración se procederá a 'Main configuración.vi'. Desde aquí se tramitan todas las funciones relacionadas con crear o modificar un archivo de configuración. Todas estas funciones y las estructuras de datos empleadas en ellas estarán situadas en la carpeta 'Configuración'. Dentro de ésta, se encuentra la carpeta 'Config' que se adapta al contenido de los archivos de configuración que se encuentran en ella.

Cuando ya se disponga de un archivo de configuración ajustado a las necesidades del ensayo se vuelve a la función inicial y se accede a 'iniciar.vi'. Esta función se encuentra en la carpeta 'Iniciar Ensayo' y en ella únicamente se escoge la configuración deseada y se procede a la creación de tareas ('establecer tareas.vi'). Todas las funciones para la creación de *tareas* se encuentran en 'Crear tareas' y divididas en diferentes carpetas según el tipo de canal que se va a emplear.

Una vez creadas las *tareas* se permite el cerado de las galgas y se accede a la *adquisición* ('empezar-cerar.vi'). La *adquisición y guardado* de datos se administra mediante la función 'Main DAQ.vi'. Durante la *adquisición* se mostrará en el panel frontal los datos que se están adquiriendo y los picos y valles que se almacenarán en el archivo binario.

En caso de tener dudas sobre las funcionalidades del software, consultar el manual de usuario.

## 2.2 Herramientas principales del programa

En este apartado se describen las estructuras de programación y de datos más comunes empleadas.



### 2.2.1 Bucles anidados

Cuando el programa entra en una función principal, el panel frontal dispone de botones para interactuar con el usuario. Si uno de ellos se pulsa, el programa realizará los comandos asociados a esa opción y entrará en una nueva función que tendrá habitualmente asociada un cambio de pantalla.

El diagrama de bloques asociado a cada panel frontal estará compuesto por un bucle 'while' dentro del cual se encuentra un 'event case'. Se asociará cada uno de los botones con un evento y se incluirá un último caso llamado 'timeout' para cuando pasado el tiempo el usuario no haya pulsado ninguna opción.

En la Figura 37 se puede ver un ejemplo de esta estructura de programación. Este diagrama de bloques pertenece a la función principal del programa, 'General.vi'. En concreto, el caso que se muestra corresponde al botón denominado 'PARAR', con el cual se sale del programa.

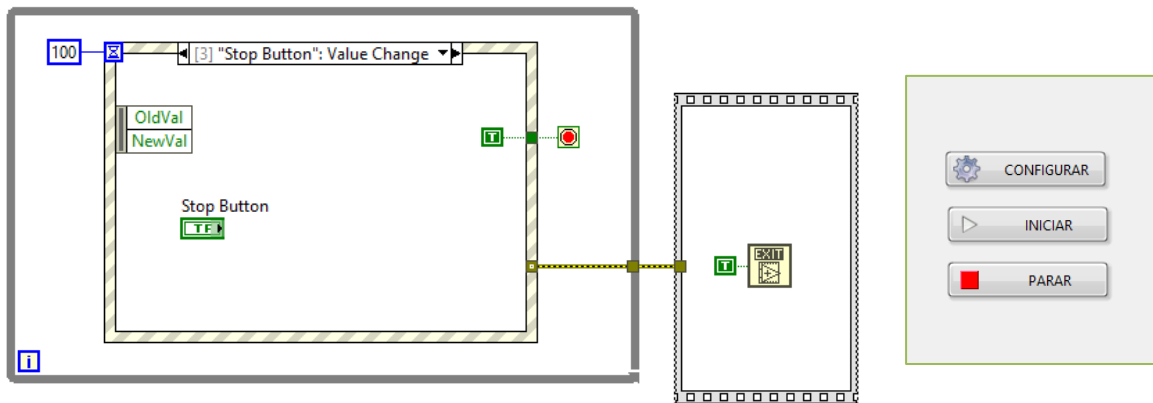


Figura 37 - Bucle 'event case' anidado a un bucle 'while'

### 2.2.2 Bucles ejecutados paralelamente

Esta arquitectura de programación se emplea en la función 'Main DAQ.vi'.

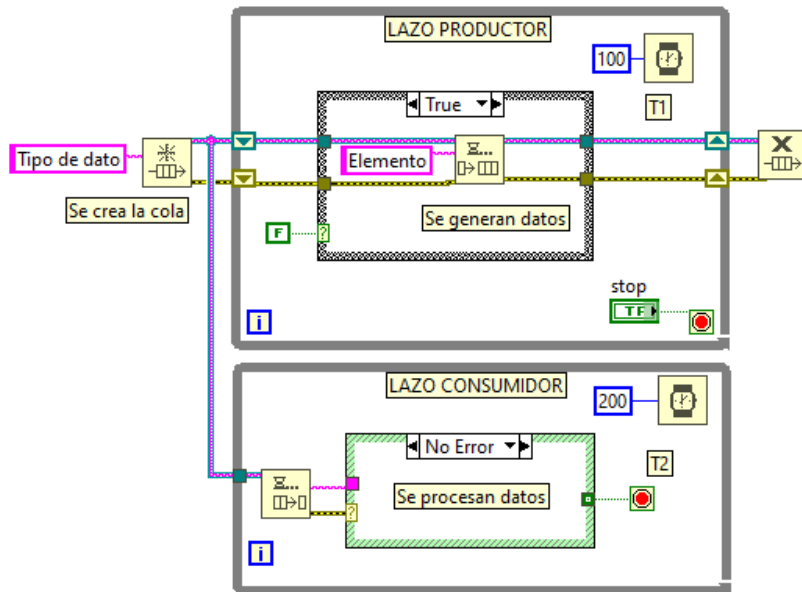


Figura 38 - Bucles productor y consumidor

Durante la *adquisición* se llevan a cabo diferentes procesos que se ejecutan paralelamente:

- ✓ Se muestrea los datos a la velocidad impuesta por el hardware
- ✓ Se adquiere los datos a la frecuencia de adquisición decidida por el usuario en el archivo de configuración
- ✓ Se calcula el pico y el valle de cada ciclo a la frecuencia impuesta por el ensayo
- ✓ Cuando se acumula un número de ciclos concretos se almacena en un archivo binario todos los picos y valles de esos ciclos

Dado que cada bucle tiene su temporización no es posible el envío directo de señales entre ellos, por ello se emplea una arquitectura de productor consumidor a través de una cola.

### 2.2.3 Estructura de datos y tareas

Dado que la adquisición de cada tipo de módulo va a ser manipulada por *tareas* diferentes, los canales indicados en el archivo de configuración van a ser ordenados e introducidos en una estructura de datos como la que se muestra en la Figura 39.



Figura 39 - Estructura general de datos

Por un lado, contiene los vectores de las diferentes *tareas* que se han creado para referirse a los diferentes módulos. En cada chasis se agrupan todos los canales del mismo tipo y se guarda una tarea global en 'NI Measurement and Automation Explorer' (NI MAX).

Por otro lado, se agrupan las propiedades de cada canal: la ganancia y el 'offset' a añadir en el procesado, el nombre y la unidad del canal y el terminal físico al que pertenece.

Finalmente, se encuentran los valores de la frecuencia de adquisición y frecuencia de ciclo del ensayo que servirán como temporización a los bucles de adquisición.

## 2.3 Diagramas de flujo del programa

Se va a describir la lógica de programación seguida a lo largo de las diferentes funciones del software.

### 2.3.1 Función principal

#### General.vi



A través de un 'event case' anidado a un bucle while, se decide si acceder a la 'función de configuración', a la 'función de acceso a la adquisición de datos' o si se sale del programa.

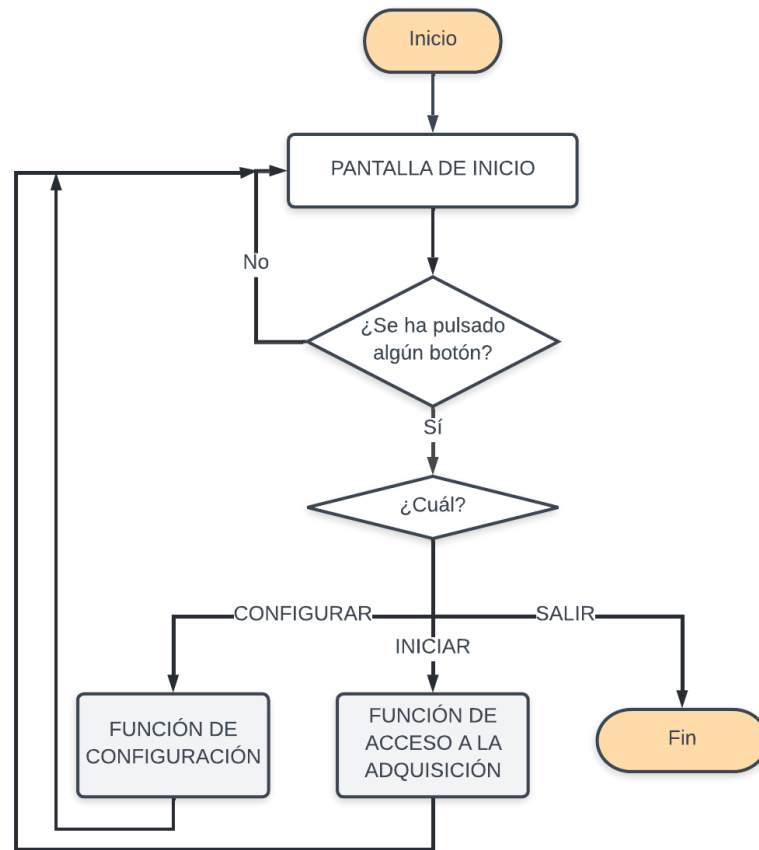


Figura 40 - Diagrama de flujo de la función principal del programa

### 2.3.2 Funciones de configuración

#### Main configuración.vi

Es posible seleccionar un archivo de configuración situado en la carpeta 'Config' para acceder a la 'función de modificar la configuración' o acceder a la 'función de crear de configuración' para empezar desde cero un archivo .ini. Se podrá también volver a la función 'General.vi'.

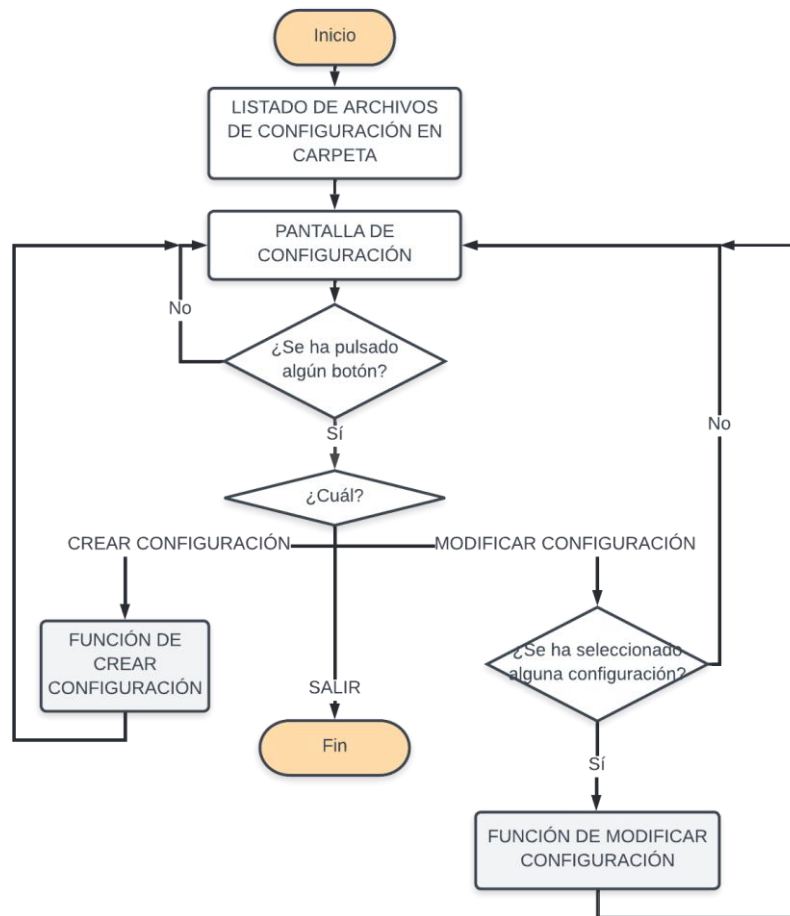


Figura 41 - Diagrama de flujo de la función de configuración

### Crear archivo de configuracion.vi

En el panel frontal de la configuración se encuentran los parámetros a rellenar para disponer de una configuración completa y válida.





**Nueva configuración**

Nombre archivo

IdentificadorEnsayo

FrecuenciaAdquisición

FrecuenciaCiclo

CantidadCiclosArchivo

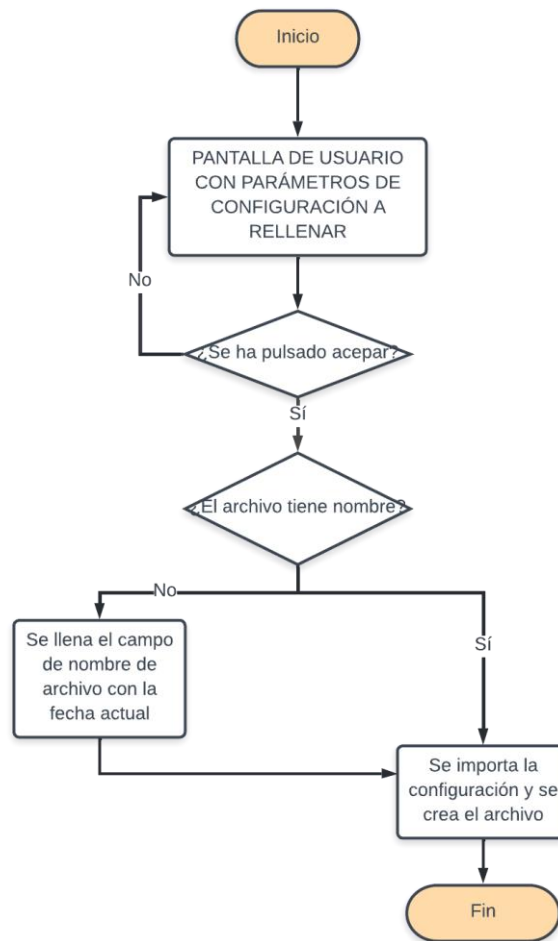
0

Tipo de canal	Tipo de canal	Tipo de canal
Extensimetría	Corriente	Voltaje
NChasis: <input type="text" value="0"/> NModulo: <input type="text" value="0"/> NCanal: <input type="text" value="0"/>	NChasis: <input type="text" value="0"/> NModulo: <input type="text" value="0"/> NCanal: <input type="text" value="0"/>	NChasis: <input type="text" value="0"/> NModulo: <input type="text" value="0"/> NCanal: <input type="text" value="0"/>
Nombre <input type="text"/>	Nombre <input type="text"/>	Nombre <input type="text"/>
Ganancia: <input type="text" value="0"/> Offset: <input type="text" value="0"/>	Ganancia: <input type="text" value="0"/> Offset: <input type="text" value="0"/>	Ganancia: <input type="text" value="0"/> Offset: <input type="text" value="0"/>
Unidades: uStrain	Unidades: mA	Unidades: V
Enabled <input checked="" type="checkbox"/>	Enabled <input checked="" type="checkbox"/>	Enabled <input checked="" type="checkbox"/>

Figura 42 - Panel frontal de la función de crear de configuración

El tipo de variable asociado a cada campo es el siguiente:

- NOMBRE ARCHIVO: 'String', cadena de caracteres
- IDENTIFICADOR ENSAYO: 'String', cadena de caracteres
- FRECUENCIA DE ADQUISICIÓN: 'Double precision floating-point'
- FRECUENCIA DE ENSAYO: 'Double precision floating-point'
- NÚMERO DE CICLOS POR ARCHIVO: 'Unsigned long' (U32)
  - o Configuración de canales: Enumerador: Galga extensiométrica, canal de corriente o de voltaje. Para guardarlo en el archivo .ini se convierte en 'string' (Gal,Cor,Vol)
  - o NChasis/NModulo/Ncanal: 'Unsigned word' (U16)
  - o Ganancia: 'Double precision floating-point'
  - o Offset: 'Double precision floating-point'
  - o Unidades: Enumerador: uStrain, mA y V. Para guardarlo en el archivo .ini se convierte en 'string'.
  - o Enable: 'Boolean'



*Figura 43 - Diagrama de flujo de la función crear configuración*

Tras rellenar los campos de la Figura 42, cuando el usuario acepta la configuración se crea un archivo .ini que aborda, en una sección general, los campos de frecuencias y cantidad de ciclo, y por cada canal crea una sección. Este archivo se podrá abrir y modificar también con un lector de texto (ver Figura 44). En caso de dejar vacío el campo de 'Nombre archivo', automáticamente se le asignará un nombre con la fecha actual.

```
mi, 11 de may. de 2022: Bloc de notas
Archivo Edición Formato Ver Ayuda
[General]
IdentificadorEnsayo = "Ejemplo de configuración"
FechaCreacion = "11/05/2022"
FrecuenciaAdq = 200,000000
FrecuenciaCiclo = 2,000000
CantidadCiclosArchivo = 200

[Section0]
TipoCanal = "Gal"
NChasis = 1
NModulo = 1
NCanal = 0
Nombre = "Galga 1"
Ganancia = 2,000000
Offset = 0,000000
Unidades = "uStrain"
Enabled = 1
```

Figura 44 - Archivo de configuración en lector de texto

### Modificar archivo de configuracion.vi



Figura 45 - Diagrama de flujo de la función modificar configuración



En un panel frontal prácticamente idéntico al de la Figura 42 se rellenan los campos con los datos que se han importado del archivo de configuración seleccionado previamente. Para poder visualizar y modificar la misma variable, se hace uso de variables locales. Mientras el usuario no pulsa el botón ‘aceptar’ el programa estará en un bucle ‘while’ y al hacerlo se asignarán los valores del panel frontal a las variables. Finalmente se reescribirá el archivo .ini con los campos nuevos.

### 2.3.3 Funciones de adquisición

#### Iniciar.vi (Función de acceso a la adquisición)

En el panel frontal aparecen en una lista todos los archivos de configuración almacenados en la carpeta ‘Config’. La selección de un archivo de la lista está asociado a un caso del ‘event case’, que guarda la ruta en una variable local. Cuando se acceda a la opción de ‘iniciar ensayo’ se comprobará que la ruta almacenada no está vacía.

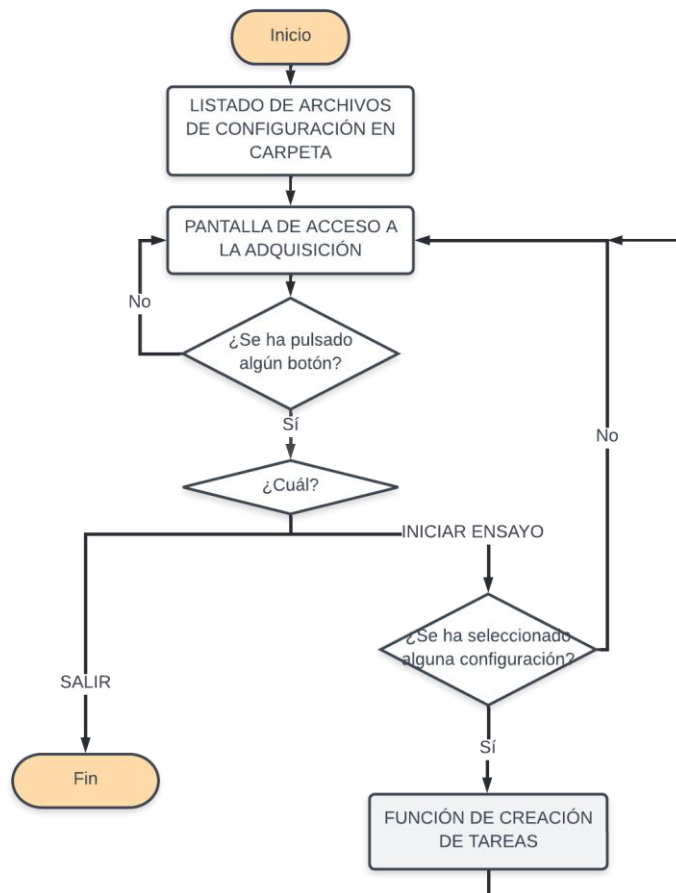


Figura 46 - Diagrama de flujo de la función de acceso a la adquisición

#### Establecer tareas.vi

Con el archivo de configuración seleccionado en la función ‘iniciar.vi’ se realiza la lectura del archivo de configuración y se reestructuran todos los datos obtenidos para hacer coincidir con la ‘Estructura de datos y tareas’.

Para ello, lo primero que se hace es dividir los canales según su tipología. Esto es necesario porque cada tipo de canal está asociado a un módulo concreto y cada módulo tiene sus limitaciones, por lo que se configuran de manera diferente.

Antes de crear las *tareas* para el ensayo, se eliminará cualquier otra *tarea* que quede almacenada en NI MAX. De esta manera se evitan conflictos con antiguas configuraciones y se evitan errores.

Cada chasis tendrá asociado una *tarea* por cada tipo de canal que contenga, con lo cual se agruparán los canales según el chasis al que pertenezcan. Esto se hace asignando a cada canal de configuración su canal físico. Por ejemplo, si en el archivo de configuración un canal tiene asignados NChasis=1, NModulo=3 y NCanal=5; corresponderá con cDAQ1Mod3/ai5.

Una vez se han realizado todas las agrupaciones necesarias, se crean las diferentes *tareas* y se guardan en NI MAX. Si la configuración no es correcta, por ejemplo, porque contiene un canal duplicado, la creación de tareas producirá un error y el programa volverá a la función 'iniciar.vi' para que el usuario pueda escoger otra configuración.

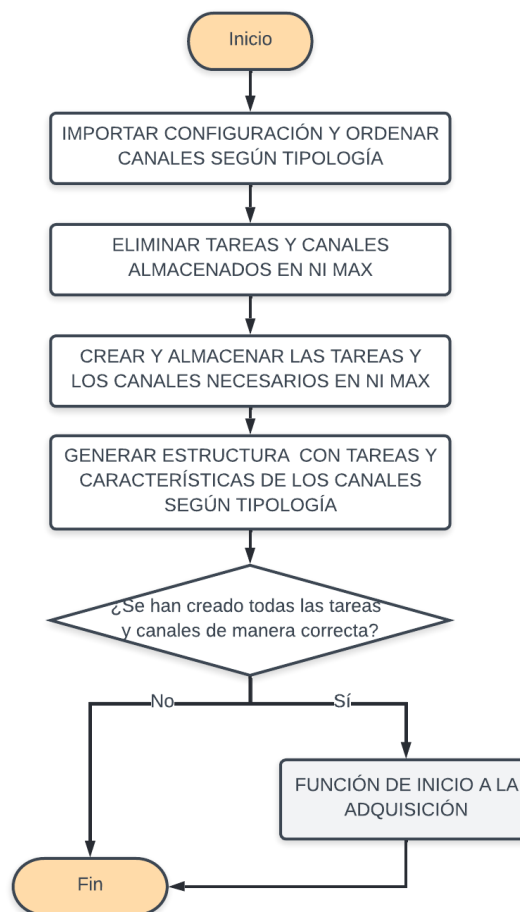


Figura 47 - Diagrama de flujo de la función de creación de tareas



### empezar-cerar.vi

Al entrar en la función, en la herramienta de NI MAX se habrán creado unas *tareas* y unos canales indicados en la configuración. A raíz de estas *tareas* se realizará todo el ensayo. Por lo que lo primero que se hace es configurar el guardado de los archivos de ensayos.

Para acceder a la *adquisición* el panel de usuario tiene asociados casos de 'event case' y permite además indicar mediante un booleano si se quiere cerrar las galgas cada vez que se pone en marcha el ensayo. En el momento de comenzar la *adquisición*, se comprueba que la ruta de archivo de *guardado* es una opción válida y que no se va a sobrescribir ningún archivo existente.

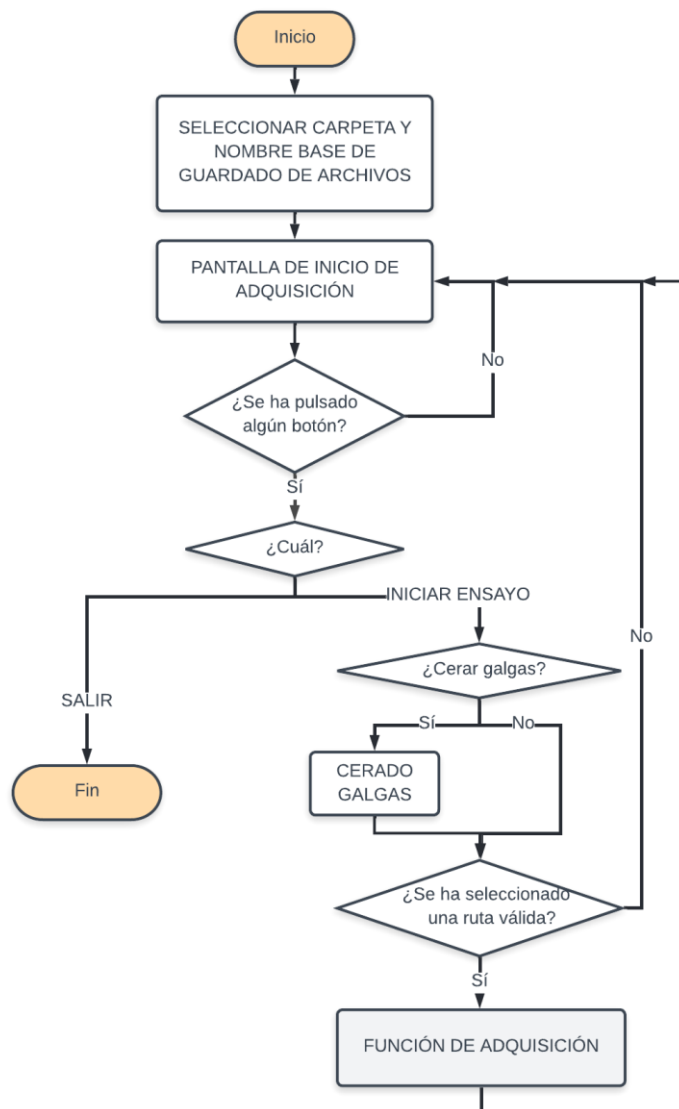


Figura 48 - Diagrama de flujo de la función de inicio a la adquisición

### Main DAQ.vi



Esta función contiene cinco bucles con diferente temporización funcionando en paralelo. El primer bucle está dedicado al panel de usuario, por lo que contiene un 'event case'. Cuando el usuario indique que quiere parar el ensayo mediante el botón 'SALIR', se activará una variable booleana global que hará que todos los bucles terminen.

El resto de los bucles están dedicados a la *adquisición, procesado y guardado* de datos. Estos bucles necesitan transferir datos y, para ello, se emplea la arquitectura productor-consumidor mencionada previamente. Las tres colas empleadas entre ellos se pueden distinguir por colores en la Figura 49.

**El bucle de muestreo** está temporizado por las distintas tareas. Es decir, al crear las *tareas* se ha asignado un reloj común a todas ellas. Este mismo reloj será el que controle la función de lectura de datos de las funciones polimórficas del driver NI DAQmx.

Una vez la función adquiera los valores de los diferentes canales se comparará un contador con el indicador 'i' para saber si es necesario enviar a cola el dato. Esto es necesario porque se quiere una adquisición de datos cercana a 50 Hz, pero los módulos empleados para la adquisición tienen una frecuencia de muestreo mínima de 800Hz. Por ello, el indicador 'i' se define como frecuencia de muestreo dividida por la frecuencia de adquisición.

**El bucle de procesamiento** a frecuencia de adquisición se ejecutará cuando le llegue un elemento a través de la cola de datos muestreados. A cada canal se le aplicarán los valores de 'offset' y ganancia que se han indicado en el archivo de configuración y se enviará el dato escalado a una 'cola de datos adquiridos' donde se acumularán todos los datos correspondientes a un ciclo.

Cuando la 'cola de datos adquiridos' alcanza el valor definido por la frecuencia de adquisición dividida por la frecuencia de ciclo (x), se realizará el *procesamiento* asociado a los ensayos de fatiga en palas de aerogeneradores. Únicamente se trata de obtener el pico y el valle asociado a cada ciclo en cada galga extensiométrica. Para ello es importante dividir los datos adquiridos en diferentes matrices según al canal al que pertenecen. Hasta ahora, las estructuras de datos estaban asociadas a *tareas* por lo que en un bloque de datos se incluían varios canales. Una vez se han obtenido las matrices de valores por canal se almacena el máximo y el mínimo en una nueva cola, 'cola de picos y valles'.

Finalmente, el último **bucle** está dedicado al **almacenamiento** de los valores obtenidos en archivos binarios. Se acumulan ciclos hasta llegar al valor N predeterminado por la configuración de ensayo. Al llegar a este valor se crea un archivo binario donde se escriben todos los atributos del ensayo y de los distintos canales además de los datos. Al terminar la escritura se cerrará el archivo y se modificará la ruta para el próximo archivo con un valor consecutivo. Estos subíndices se acumulan siempre y cuando no se salga de la función de 'empezar-cerrar.vi'.

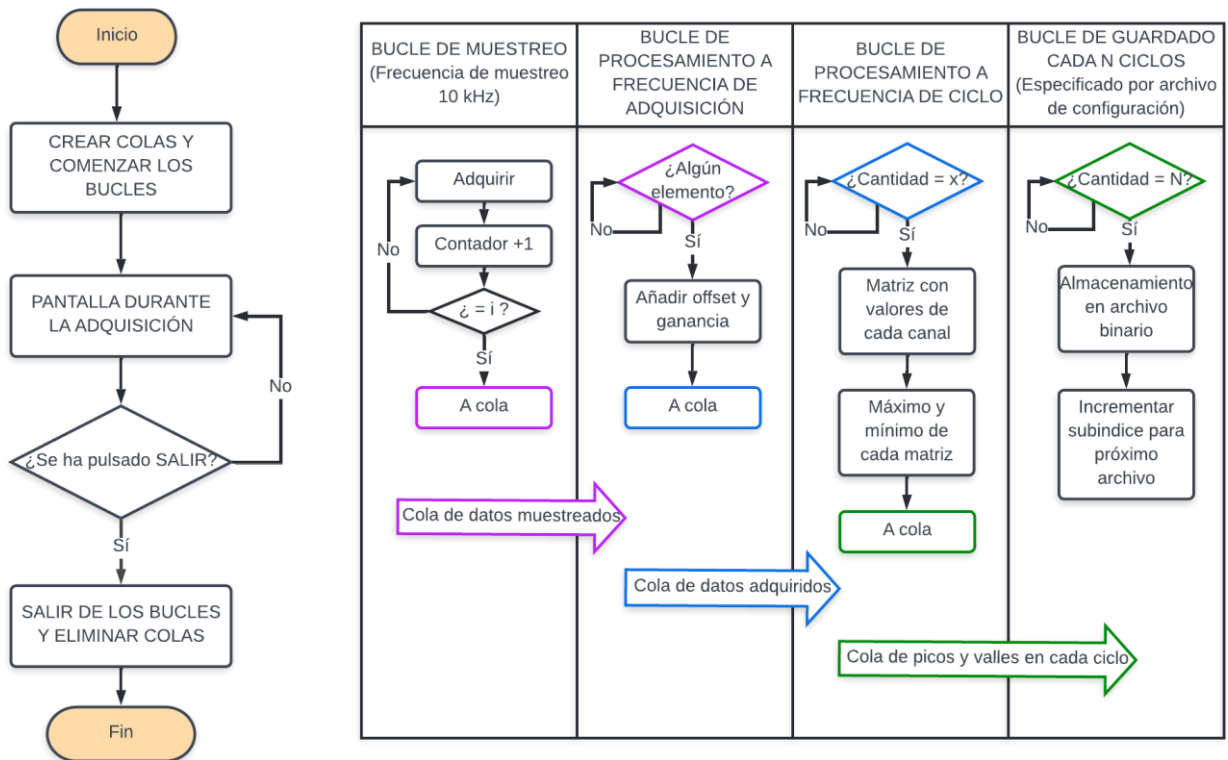


Figura 49 - Diagrama de flujo de la función de adquisición

## 2.4 Modificaciones en el programa

En caso de querer ampliar las funciones que tenga el programa, en este apartado se describen cómo se deberían hacer las futuras modificaciones.

Al realizar cualquiera de estas modificaciones es necesario volver a construir el ejecutable para poder apreciar los cambios en la aplicación.

### 2.4.1 Aumentar número de canales admitidos

Cuando se importan los canales desde un archivo de configuración se le asigna inicialmente un máximo de canales que se van a leer.

Esto se realiza en la función 'Lectura archivo configuracion.vi' situada en la carpeta '\\\_Aplicacion\Configuracion'. El diagrama de bloques de esta función se encuentra en la Figura 50. Para modificar este límite bastaría con cambiar el número de la constante que se le asigna a 'Cantidad Canales'.



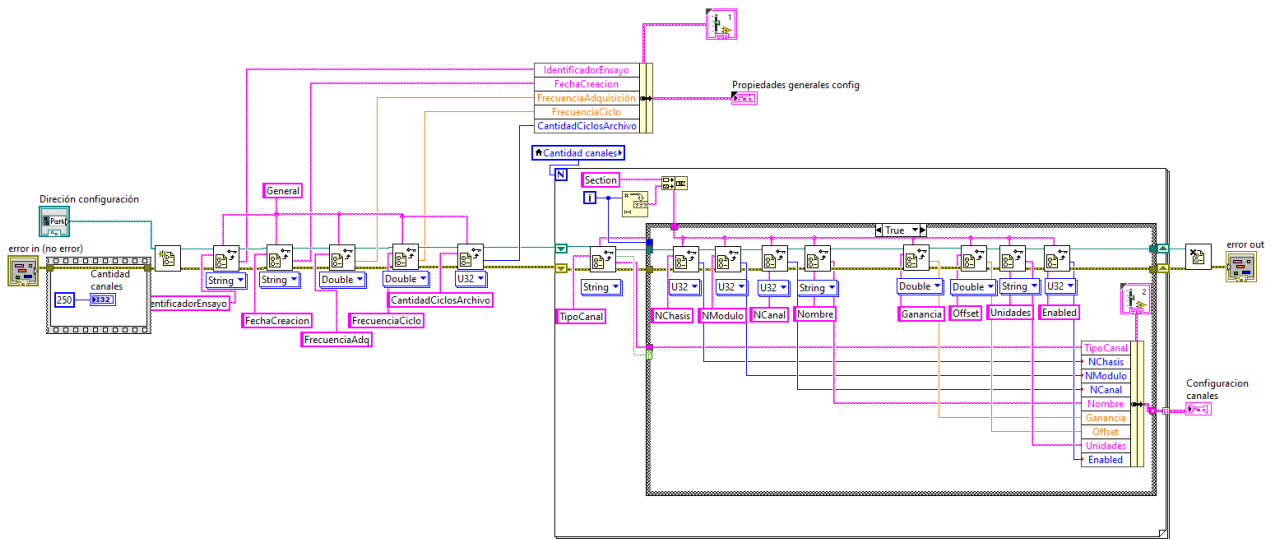


Figura 50 - Diagrama de bloques de la función lectura archivo de configuración

### 2.4.2 Modificar frecuencia de muestreo

La frecuencia de muestreo de los canales queda impuesta por el hardware, más concretamente por el módulo de adquisición de galgas extensiométricas NI 9236. Este únicamente permite las frecuencias indicadas en el enumerador de la función 'Generar estructura.vi'. Actualmente la frecuencia de muestreo asignada es la más elevada que permite el módulo (10kHz). En caso de querer modificarla sólo hay que desplegar el enumerador y escoger otro valor.

Es posible además asignarle un valor al buffer de muestreo. Este valor se asignará tanto al buffer de adquisición continua del reloj como al número de muestras a obtener en cada lectura. Los datos obtenidos en una lectura se promediarán de manera que se consigue un filtro para disminuir así el efecto de valores erróneos.

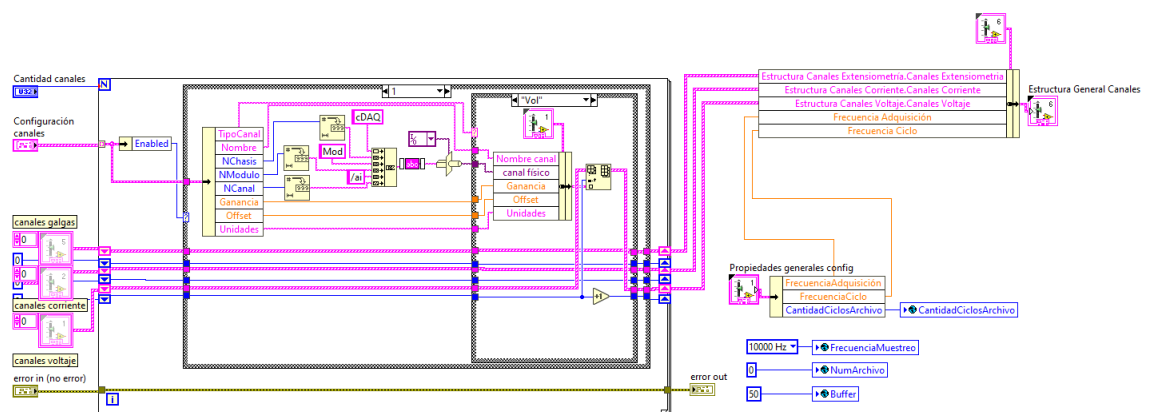


Figura 51 - Diagrama de bloques de la función Generar estructura

### 2.4.3 Modificar el formato del archivo de guardado

El formato actual del archivo de guardado binario está hecho de manera que coincida con el formato actual que se emplea en el 'Laboratorio de Ensayos de Aerogeneradores-CENER'.

Dado que se va a leer los archivos con una función de Matlab, el orden de datos empleado será 'little-endian'.

Se describen los atributos contenidos actualmente en el archivo binario para saber así dónde realizar los cambios.

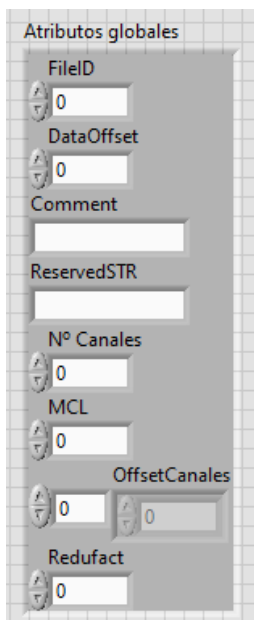


Figura 52- Estructura de atributos globales

- FileID, (I16): Todos los elementos creados con esta aplicación tendrán como identificador de archivo 5012.
- DataOffset, (I32): Posición del archivo donde se empiezan a escribir los datos de los canales
- Comment, (String): Le precede la longitud de la cadena de caracteres en un entero (I16).
- ReservedSTR, (String): Se reservan 32\*256 caracteres para escribir la ruta de guardado del archivo.
- NCanales, (I16): Cantidad de canales, N. Los dos primeros pertenecen al tiempo absoluto y relativo desde el comienzo del ensayo. El resto será el pico y el valle de cada canal incluido en el archivo de configuración.
- MCL, (I32): Longitud máxima de cada canal, si es 0 es que no hay límite.
- OffsetCanales, (N\*I32): La posición del archivo donde se empieza a escribir las características de cada canal. La función de Matlab no almacena en la estructura de datos estos valores, pero sí que realiza la lectura por lo que es necesario escribirlos.
- Redufact, (I32): Factor de reducción, normalmente 0.

#### ATRIBUTOS CANAL:

- Número, (I16): Número asignado al canal, empezando por 0 primero los canales de tiempo y luego los de picos y valles.
- Longitud, (I32): Cantidad de datos en el canal.
- Nombre, Unidad y Comentario (String): Les precede la longitud de la cadena de caracteres en un entero (I16).
- Formato, (I16): En caso de que sea un número tomará valor 0.
- DataWidth, (I16): Normalmente 8.
- Datumzeit, (DBL 64, float).
- Cabecera, (String): Les precede la longitud de la cadena de caracteres en un entero (I16).

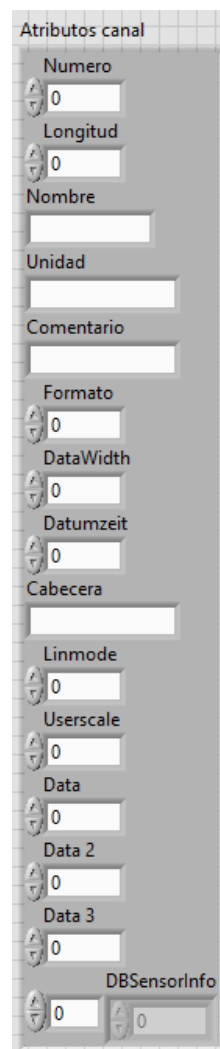


Figura 53 - Estructura atributos canales



- Linmode, (I8): 1
- Userscale, (I8): 0 si no se emplea escala personalizada.
- Valores vacíos para poder leer con la función ya existente (I8, I16 e I16)
- DBSensorInfo, (1D array of I8): Le precede el tamaño del vector.

Muchos de estos valores no aportan significado a los datos, por lo que se les asigna valor '0' para poder leer con la función ya existente en la empresa y, de esta forma, poder emplear las aplicaciones de post procesado propias.

En el caso de cambiar cualquier parámetro será necesario modificar también la función de Matlab asociada a la lectura del archivo binario 'read.m'.

#### 2.4.4 Añadir un nuevo tipo de módulo

En caso de necesitar adquirir datos no compatibles con los módulos para los que está diseñada esta aplicación. Se necesitará modificar el programa añadiendo en cada bloque del proceso una nueva función con los datos adaptados a las necesidades del nuevo módulo. Las funciones o estructuras donde se deberían añadir bloques adicionales son:

- Configurar canal.cti: Añadir en el enumerador un nuevo caso
- Crear tareas.vi: Duplicar todos los bloques asociados al tipo de canal. Por un lado, están las funciones de generar estructura de *tarea* y crear la *tarea*. Por otro lado, están las estructuras asociadas a esas funciones.
- Estructura general datos.cti: Hay que añadir una fila más de *tareas* y una columna más de estructura de canales.
- Main DAQ.vi: En los bucles de muestreo y procesamiento a la velocidad de adquisición, se funciona por tareas según módulos con lo cual se deberá añadir bloques también.

Se recomienda, antes de incluir el nuevo módulo en la aplicación general, comprobar el funcionamiento de éste empleando los ejemplos que ofrece NI Example Finder para el hardware en cuestión (Ni Example Finder\Hardware Input and Output\DAQmx). A través de estos ejemplos se puede observar cómo configurar adecuadamente el módulo y qué limitaciones tiene.