# **Medición y registro continuos**

El proyecto de ejemplo **Continuous Measurement and Logging** (CML) adquiere mediciones continuamente y las registra en disco. Ejecuta cinco bucles en paralelo:

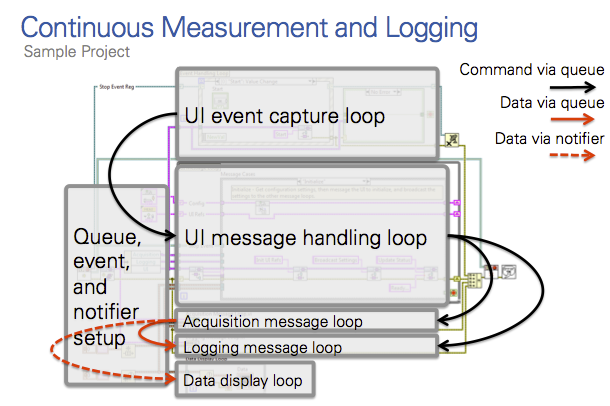
* **Manejo de eventos** (Main.vi) — El bucle de manejo de eventos (EHL) que produce mensajes basados en eventos del panel frontal, como cuando el usuario hace clic en **Inicio** o **Configuración**.
* **Mensajería de la interfaz de usuario** (Main.vi) — Un bucle de manejo de mensajes (MHL) que recibe mensajes del EHL y responde enviando mensajes a los otros MHL.
* **Adquisición de datos** (Acquisition.lvlib:Acquisition Message Loop.vi) — Un MHL que adquiere datos continuamente. Por defecto, esta plantilla simula datos adquiridos.
* **Registro de datos** (Logging.lvlib:Logging Message Loop.vi) — Un MHL que registra continuamente los datos adquiridos.
* **Visualización de datos** (Main.vi) — Un bucle While que actualiza el gráfico de forma de onda con los datos adquiridos.

Este proyecto de ejemplo también incluye un cuadro de diálogo **Settings** (Settings.lvlib) que puede utilizar para configurar la aplicación.

## **Requerimientos de Sistema**

LabVIEW Base, Completo, o Sistema de Desarrollo Profesional. Este proyecto de ejemplo está diseñado para usarse con NI-DAQmx, un controlador de instrumento u otro software controlador.

## **Diagrama del Proyecto**



## **Casos de uso**

El proyecto de ejemplo CML está diseñado para una aplicación de medición continua que requiere una interfaz de usuario con capacidad de respuesta; es decir, los usuarios deben poder hacer clic en los botones incluso mientras la aplicación está ejecutando otro comando.

## **Ejecución de este proyecto de ejemplo**

1. En la ventana del **Project Explorer**, abra y ejecute Main.vi.
2. Haga clic en **Start**. El programa comienza a adquirir datos de forma de onda simulada.
3. Haga clic en los otros botones del panel frontal para explorar el proyecto de ejemplo.

## **Modificación de este proyecto de ejemplo**

### *Añadir código de adquisición de datos*

Debe modificar el proyecto de ejemplo para adquirir datos del hardware. Complete los siguientes pasos para realizar estas modificaciones:

1. Añada refnums de hardware a Acquisition.lvlib:Hardware Configuration.ctl. Por ejemplo, puede utilizar los siguientes objetos aquí:

* DAQ tasks
* DAQ channels
* VISA sessions

1. Añada código de inicialización de hardware a Acquisition.lvlib:Initialize Hardware References.vi. Por ejemplo, puede utilizar los siguientes objetos aquí:

* DAQmx Task Name constants
* DAQmx Create Virtual Channel VI
* (Instrument Driver) Initialize VI

1. Añadir código de configuración de hardware a Acquisition.lvlib:Configure Hardware.vi Por ejemplo, puede utilizar los siguientes VIs aquí:

* DAQmx Timing VI
* DAQmx Trigger VI
* (Instrument Driver) Configure Measurement VI
* (Instrument Driver) Configure Autozero VI

1. Añada el código de adquisición de datos a Acquisition.lvlib:Acquire.vi. Por ejemplo, puede utilizar los siguientes VIs aquí:

* DAQmx Read VI
* (Instrument Driver) Read VI

1. Añada código que detenga la adquisición de datos a Acquisition.lvlib:Stop Acquisition.vi. Por ejemplo, puede utilizar los siguientes VIs aquí:

* DAQmx Clear Task VI
* (Instrument Driver) Close VI

### *Personalizando el Código de Registro de Datos*

Si el comportamiento de registro por defecto no satisface las necesidades de su aplicación, puede modificar este proyecto de ejemplo de las siguientes maneras:

* Para especificar dónde se registran los datos, ejecute Main.vi, haga clic en **Settings** y utilice el control **Log File Path**. Por defecto, esta plantilla registra los datos en LabVIEW Data\Logged Data.tdms, donde LabVIEW Data es la carpeta LabVIEW Data.
* Para cambiar el mecanismo de registro de datos, modifique Logging.lvlib:Logging Message Loop.vi. Por ejemplo, podría modificar este VI para transmitir los datos adquiridos a través de una red o a disco.
* Para cambiar el código que escribe los datos en el disco, modifique Logging.lvlib:Log Data.vi. Por ejemplo, podría utilizar los VIs Export Waveforms to Spreadsheet File o Write to Spreadsheet File. Por defecto, esta plantilla utiliza las funciones TDMS para registrar los datos en un archivo .tdms.
* Para cambiar o añadir rutas y refnums de archivos que son necesarios para el registro de datos, modifique Logging.lvlib:Logging Configuration.ctl.