

# Adquirir una forma de onda analógica con MCC DA

Mejía Cruz, Josefina.  
it@logicbus.com  
Logicbus SA de CV

*Resumen*— En el presente artículo se otorgara la información acerca de cómo adquirir una forma de onda analógica. Además de mostrar cómo capturar una forma de onda analógica utilizando un dispositivo de adquisición de datos.

## *Índice de Términos*—

**Señal analógica:** una señal analógica es una señal que varía de forma continua a lo largo del tiempo. La mayoría de las señales que representan una magnitud física (temperatura, luminosidad, humedad, etc.) son señales analógicas. Las señales analógicas pueden tomar todos los valores posibles de un intervalo; y las digitales solo pueden tomar dos valores posibles.

**Adquisición de Datos:** La adquisición de datos (DAQ) es el proceso de medir con una PC un fenómeno eléctrico o físico como voltaje, corriente, temperatura, presión o sonido. Un sistema DAQ consiste de sensores, hardware de medidas DAQ y una PC con software programable.

## I. INTRODUCCIÓN

Al hablar de los sistemas de adquisición de datos, estamos refiriéndonos a los productos o procesos que se usan para recoger información para documentar y analizar un fenómeno. Por ejemplo desde la tarea más sencilla, en el cual un técnico realiza el registro de la temperatura de un horno en un papel, está ejecutando la adquisición de datos.

Hasta actividades más complejas y con herramientas tecnológicas que hacen posible llevar a cabo estas actividades de manera más versátil y fiable a través del equipo va de graficadores sencillos a sistemas de cómputo sofisticados.

## II. DESTINATARIOS

Los desarrolladores de sistemas de adquisición de datos que necesitan digitalizar una forma de onda analógica a lo largo del tiempo y buscan comprender la implementación de la captura de formas de onda en diversos entornos de software.



**Figura 1. Desarrolladores de adquisición de datos**

Para los fenómenos analógicos que cambian muy lentamente, puede ser suficiente con muestrear la señal con poca frecuencia, y a menudo es aceptable que el tiempo delta entre las muestras varíe, ya que la necesidad principal es obtener el estado actual de la señal. En tales casos, un simple "sondeo de software" de cada canal de AI puede ser suficiente. Sin embargo, si la señal cambia rápidamente con respecto al tiempo transcurrido, o si un componente de frecuencia de la señal es de interés para el usuario, se requiere un muestreo A / D mucho más rápido. De hecho, no solo se requiere un muestreo más rápido, sino que es importante tener un delta de

tiempo uniforme y bien controlado entre las muestras.

Los dispositivos DAQ AI de MCC incorporan circuitos de temporización a bordo para controlar de manera uniforme las lecturas de uno o más canales de AI. Los dispositivos MCC DAQ también incorporan memoria integrada para hacer cola de datos para enviar periódicamente paquetes de tamaño eficiente a través del bus de interfaz (generalmente USB). El controlador de software de MCC trabaja junto con el hardware DAQ, que recibe los paquetes de datos, los agrupa en un búfer más grande en la memoria y hace que los datos y el estado estén disponibles para el programa de usuario de nivel superior.

Un desarrollador de aplicaciones querrá ampliar el proyecto desde la perspectiva de la duración del muestreo de AI y las necesidades de acceso a los datos. En otras palabras, ¿cuánto tiempo necesito muestrear y necesito acceso a los datos durante la recopilación o no hasta el final? El caso más sencillo es recopilar datos para una duración relativamente corta y no procesar ningún dato hasta que se hayan recopilado. Al extender la idea, es posible que deba comenzar a procesar los datos durante la recopilación, por ejemplo, para proporcionar una visualización en vivo al operador. En este caso, el desarrollador desea acceder al búfer de datos mientras aún se agregan nuevos datos, lo que MCC facilita con su modo de Fondo. El escenario más complejo es un tiempo de recopilación muy largo, por lo que es necesario un búfer circular debido a la memoria finita de la computadora. Con nuevos datos constantemente entrando, la aplicación debe ser capaz de realizar las matemáticas, la visualización y, tal vez, los resultados deseados en el disco, lo suficientemente eficiente como para mantenerse al día. Si bien no es el enfoque de este artículo, el controlador de software de MCC admite esta necesidad avanzada con su modo Continuo.

Para un escaneo no continuo, independiente de la plataforma de software, se debe especificar la siguiente información:

- Números de canal AI
- Número de muestras a recoger.
- Velocidad de reloj
- Rango de entrada

- El búfer de memoria para almacenar los datos en
- Además de los datos A / D, se devuelve un código de error.

Los siguientes ejemplos muestran cómo realizar un escaneo no continuo utilizando la Biblioteca Universal™, UL para Android™, NI LabVIEW™, DASyLab® y DAQami.

### III. LIBRERÍAS

El siguiente ejemplo muestra cómo realizar un escaneo no continuo utilizando la Biblioteca Universal (UL) y Visual Basic:

LowChan = 0

HighChan = 0 'Recolecte solo del canal 0

Recuento = 50000 'Especifique el número total de muestras a recolectar

Velocidad = 50000 'Especifique la frecuencia de reloj por canal

Rango = MccDaq.Range.Bip10Volts' Especifique la entrada de +/- 10V range

ULStat = DaqBoard.AInScan (LowChan, HighChan, Count, Rate, Range, MemHandle, Opciones)

Los datos de muestra se devuelven a un búfer en la memoria de la PC en el identificador provisto, mientras que el código de error se devuelve como el valor de ULStat.

A continuación, copiamos los datos del búfer de memoria a una matriz de programa local, llamada ADData:

ULStat = MccDaq.MccService.WinBufToArray (MemHandle, ADData, FirstPoint, Count)

Hasta ahora, los datos se han transferido y almacenado en el formato de conteos por razones de eficiencia. Podemos convertir a voltaje con la siguiente función UL:

ULStat = DaqBoard.ToEngUnits (Range, ADData (0), EngUnits)

### IV. UL PARA ANDROID

El siguiente ejemplo muestra cómo realizar un escaneo no continuo usando UL para Android y Java:

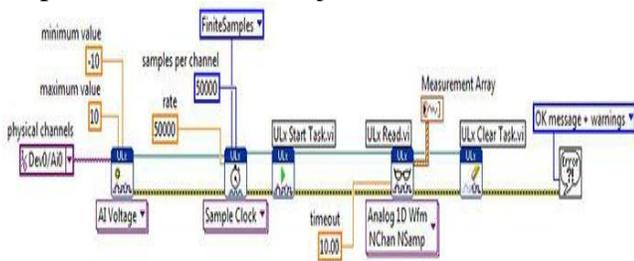
lowChan = 0;

highChan = 0; // Recopilar solo del modo canal 0

```
= ChannelMode.SINGLE_ENDED; // Especificar
rango de modo de configuración =
Range.BIP10VOLTS; // Especifique el rango de
entrada de ± 10V
samplesPerChan = 50000; // Especifique el número
de muestras por canal
= 50000; // Especifique la frecuencia de reloj por
canal
mUnit = AiUnit.VOLTS; // Especifique las
unidades de voltaje
actualScanRate = mAIDevice.aInScan (lowChan,
highChan, mode, range, samplesPerChan, rate,
options, mUnit, mScanData);
Recopile los valores llamando a la función aInScan
(). Los datos se devuelven en la matriz denominada
mScanData.
```

### V. LABVIEW

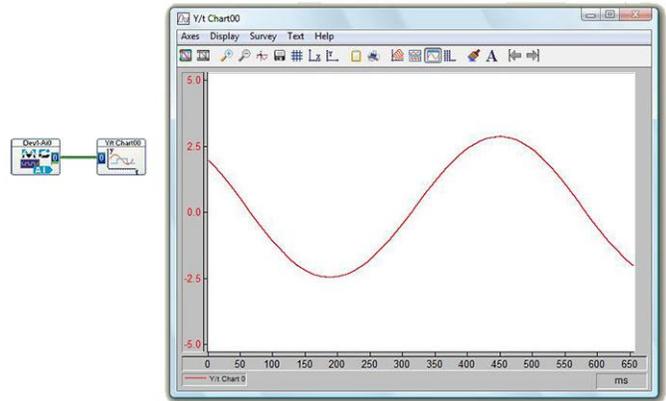
El siguiente ejemplo muestra cómo realizar un escaneo no continuo usando LabVIEW. El siguiente diagrama de cableado muestra los VIs y configuraciones pertinentes. Aquí se solicita al primer canal (Ai0) del dispositivo de hardware # 0 que devuelva 50000 muestras, a una velocidad de 50000 Hz, dentro del rango esperado de -10 V a +10 V. Los datos de voltaje resultantes están disponibles como el Conjunto de medidas.



### VI. DASYPYLAB

Los no programadores pueden considerar el uso de uno de varios paquetes que facilitan la recopilación, visualización y almacenamiento de datos de forma de onda. Un paquete popular es DASYPYLAB.

La siguiente hoja de trabajo de ejemplo muestra un módulo de entrada analógica con su salida conectada a un módulo de visualización de carta Y / t. La forma de onda del voltaje capturado se muestra en la pantalla del gráfico.



### REFERENCIAS

[1] MCC, Wireless Power: TechTip: Acquiring an Analog Waveform with MCC DAQ, Noviembre 2018, Disponible en línea en: <https://www.mccdaq.com/TechTips/TechTip-7.aspx>