

Adquisición de Datos usando Matlab

Bruno Vargas Tamani

Facultad de Ingeniería Electrónica y Eléctrica, Universidad Nacional Mayor de San Marcos, Lima, Perú

RESUMEN: La interconexión a nivel de computadoras personales que contienen tarjetas de adquisición de datos a procesos, nos permiten realizar experimentos y desarrollar proyectos en diferentes áreas (procesamiento de señales, control automático, etc.). Mostramos los pasos necesarios para realizar la adquisición de datos analógicos a través de la tarjeta PCI 6024E utilizando como interfase de programación el MATLAB. También se muestra cómo producir una salida de datos analógicos con el mismo sistema. El MATLAB se ha convertido en un poderoso software de ingeniería a través del cual podemos implementar diferentes técnicas de procesamiento de datos, aumenta esa potencia por la ventaja de poder acceder a procesos mediante la toma de datos de estos, procesarlos y modificar el comportamiento del proceso mediante la salida de datos. Se indica cómo se puede acceder a señales analógicas de entrada y salida de un proceso, con las facilidades que ofrece el MATLAB.

I. INTRODUCCIÓN

Contando con un sistema de adquisición de datos, basado en una tarjeta DAQ (data acquisition) de National Instruments PCI 6024E y el software MATLAB versión 6.5 de Math Works, se han realizado pruebas de lectura de datos analógicos, así como generado salidas de señales analógicas. Se describirá los pasos necesarios para realizar esas operaciones [1],[2]. MATLAB se ha constituido en un programa de alto nivel bastante utilizado en ingeniería. Particularmente en el área de control automático, nos permite utilizar las diferentes herramientas que ofrece para el análisis y diseño de sistemas de control, siendo relevante el hecho de poder experimentar con procesos de control, mediante la comunicación con ellos a partir de tarjetas de adquisición de datos para computadoras personales a través del mismo software. Entonces en el mismo ambiente de programación podemos

experimentar con el proceso prácticamente en tiempo real.

II. INSTALACIÓN DEL SOFTWARE REAL-TIME WINDOWS TARGET

Para poder realizar adquisición de datos con MATLAB, se requiere instalar el software Real-Time Windows Target que forma parte del mismo MATLAB. Este software permitirá que las aplicaciones accedan al Kernel del sistema donde los ejecutables deben ser cargados y ejecutados. El driver del modo Kernel intercepta las interrupciones de tiempo del reloj del computador, manteniéndolas para operaciones del sistema Windows y asegura la ejecución de las aplicaciones en tiempo real.

La instalación del Real-Time Windows Target, se realiza en el área de trabajo del MATLAB, escribiendo:

```
>> rtwintgt -install
```

Para verificar las características resultado de la instalación del Kernel, puede escribir :

```
>> rtwho
```

El Real-Time Windows Target requiere de un compilador C que no está incluido en el software de MATLAB. De no tener instalado algún compilador C en su computadora, es necesario instalar uno. Para elegir el compilador C, de entre los instalados debe escribir en el área de trabajo :

```
>> mex -setup
```

Ud. debe indicar que le muestre la lista numerada de compiladores instalados y luego debe escoger el que desee, escribiendo el número correspondiente. Normalmente se recomienda elegir si está en la lista el Microsoft Visual Studio C/C++.

III. LECTURA DE DATOS ANALÓGICOS

Luego de instalado el Kernel mediante el Real-Time Windows Target y elegido el compilador para crear los ejecutables para ser usados en tiempo real, ya podemos hacer uso de las librerías de MATLAB para adquisición de datos.

Para leer datos analógicos, se conectan esas señales analógicas a los puertos de entrada de la DAQ. Luego cree un modelo nuevo en Simulink, como se muestra en ejemplo en la figura 1. De la librería Real-Time Windows Target seleccione y arrastre el bloque Analog Input, el cual nos permitirá la lectura de datos Analógicos. Para poder visualizar los datos adquiridos agregue un elemento Scope. Opcionalmente puede agregar un elemento To Workspace, para crear una variable que almacene los datos adquiridos en el área de trabajo del Matlab. Se muestra también la creación de una variable para la base de tiempos.

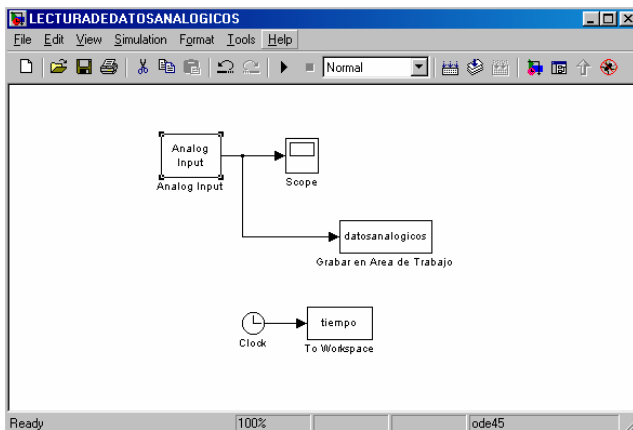


Fig. 1. Creación de modelo de lectura de datos.

Debemos configurar el bloque Analog Input, seleccionémoslo y hagamos doble clic en él y aparecerá la pantalla de la figura 2.

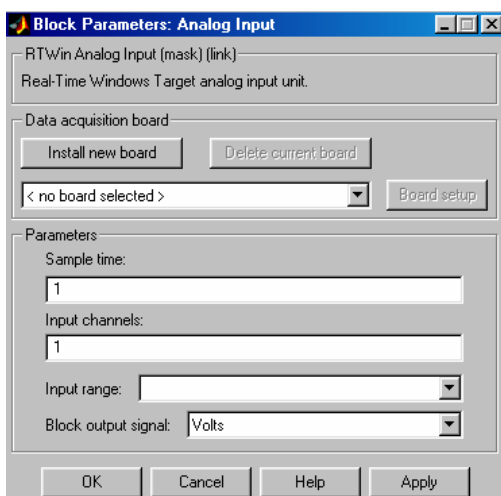


Fig. 2. Pantalla para configurar bloque Analog Input.

Primero debe seleccionar de las tarjetas de adquisición de datos instaladas en su computador, la que está usando (en este caso National Instruments PCI-6024E). Observe la figura 3.

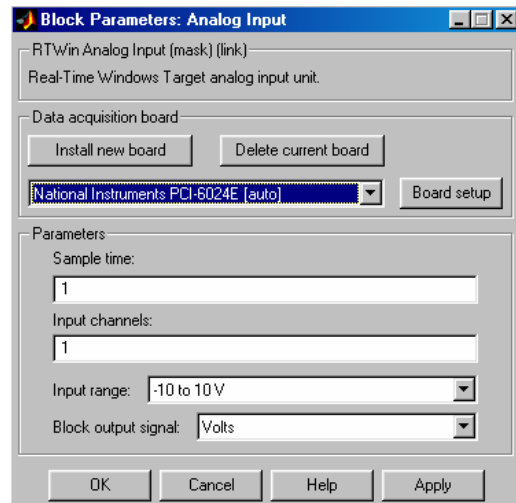


Fig. 3. Selección de DAQ National Instruments PCI-6024E.

Aparecerá en el modelo del proyecto el bloque Analog Input indicando la tarjeta seleccionada (figura 4).

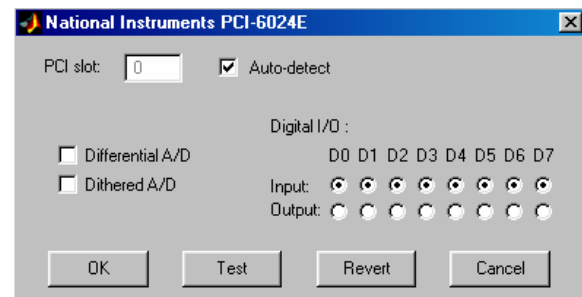


Fig. 5. Pantalla para test de tarjeta DAQ.

Pulse Test para verificar el estado de la tarjeta; en caso de estar en perfecto estado envía un mensaje con la pantalla mostrada en la figura 6.

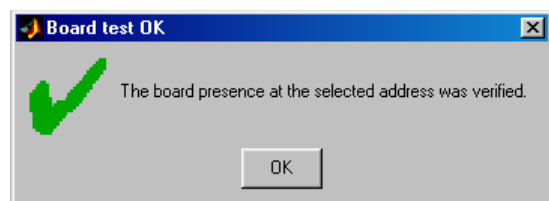


Fig. 6. Pantalla de verificación de DAQ.

En esa pantalla pulsamos OK y regresamos a pantalla de configuración (figura 3). Ahora configuramos el período de muestreo a utilizar y los

canales a utilizar en la lectura de datos analógicos, como muestra la figura 7.

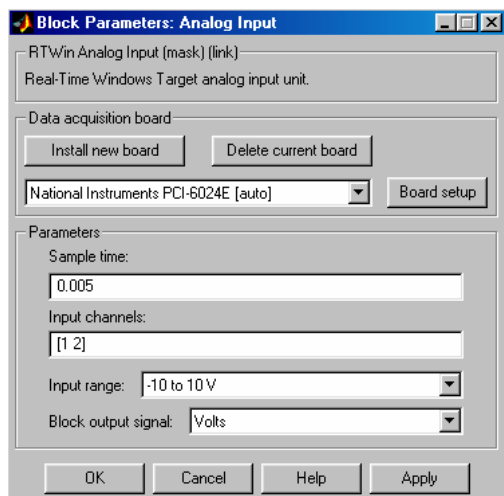


Fig. 7. Configura período de muestreo y canales de lectura.

Finalizamos pulsando OK.

Continúa la configuración de los parámetros de la presentación de la simulación. En la barra de menús Tools del modelo, seleccionar External mode control panel. Aparece la pantalla mostrada en la figura 8.

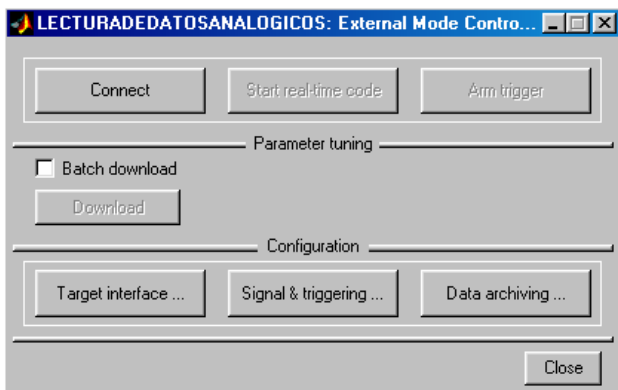


Fig. 8. Pantalla External Mode Control Panel.

Seleccionar la opción Target interface, escribir en el MEX-file : rtwinext.

Observe la figura 9.

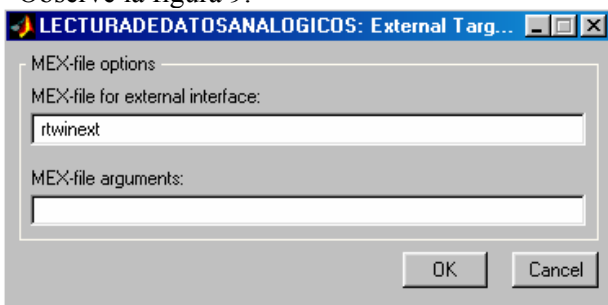


Fig. 9. Configuración del MEX-file.

Con OK, regresamos a pantalla External mode control panel, (figura 8.) seleccionar Signal&triggering. Configurar el número de datos adquiridos que presentarán los bloques Scope y almacenarán los bloques To Workspace. Observe figura 10.

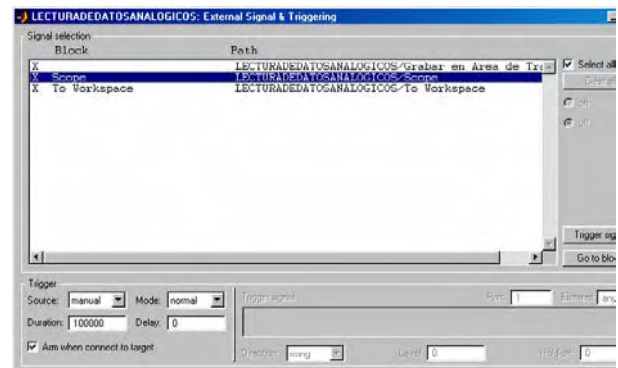


Fig. 10. Configurar número de puntos que presentarán los Scope.

Luego pulsamos Apply y Close para terminar esta configuración.

Continúa ahora la configuración de los parámetros de simulación. En la barra de menús Simulation del modelo, seleccionar Solver y aparece pantalla de la figura 11.

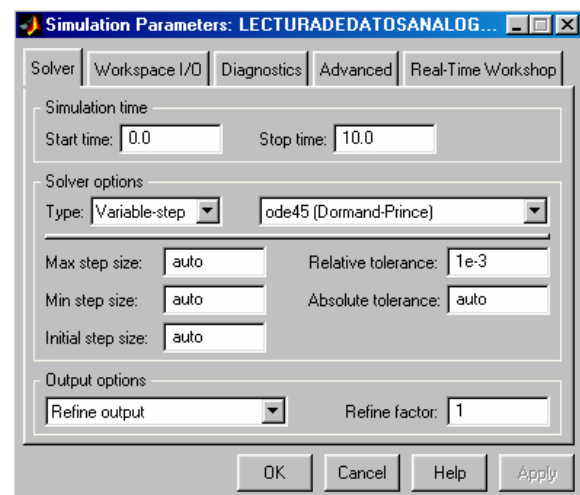


Fig. 11. Configurar solución de simulación.

Seleccionamos tipo de solución Fixed-step y Ode5. Observe figura 12.

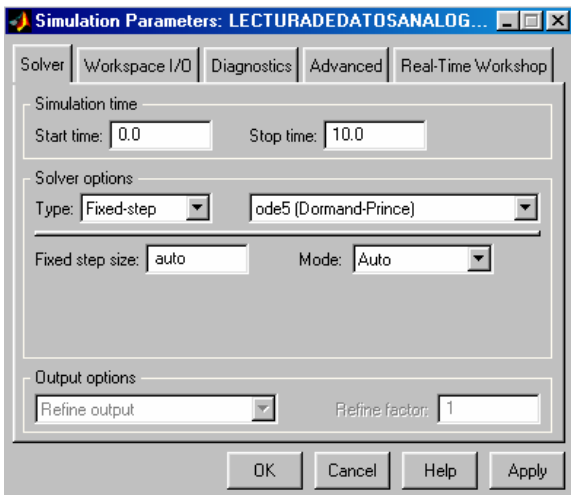


Fig. 2. Selección del modo de solución.

En la misma pantalla seleccionar Real-Time Workshop. Aquí configuraremos las opciones de creación y ejecución en tiempo real. Aparece inicialmente la pantalla de la figura 13.

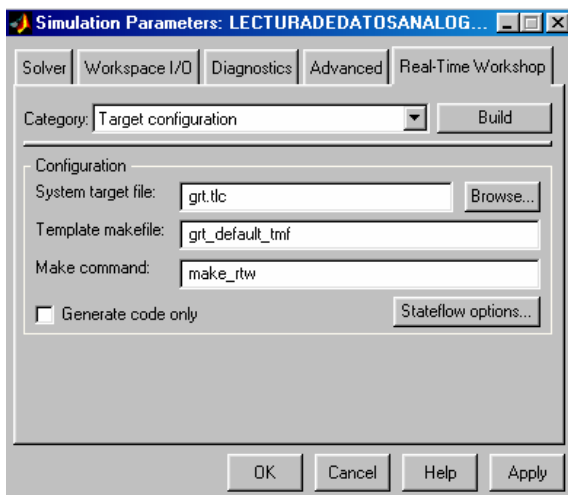


Fig. 13. Pantalla de configuración de opciones de creación y ejecución en tiempo real.

Al pulsar Browse, aparecen los diferentes archivos para el manejo de de creación y ejecución en tiempo real. Seleccionar : rtwin.tlc. Observe figura 14.

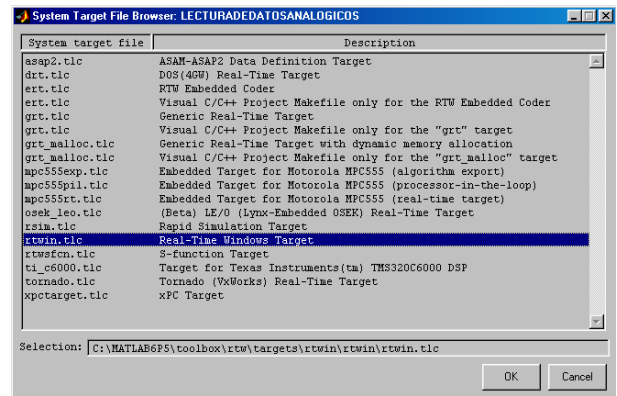


Fig. 14. Selección de manejador rtwin.tlc.

Pulse OK y aparece inicialmente la pantalla de la figura 15, que muestra la opción seleccionada.

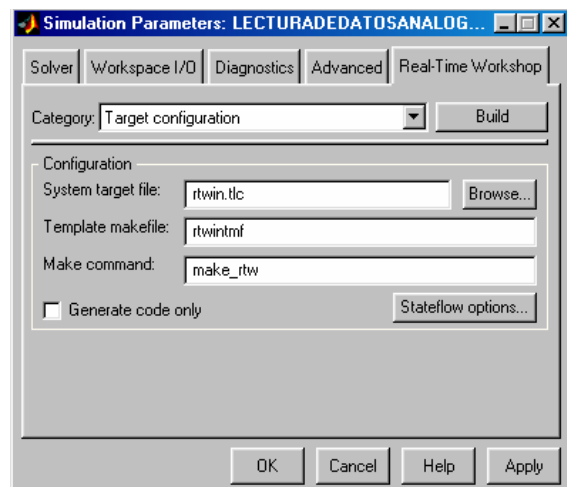


Fig. 15. Manejador rtwin.tlc para la DAQ.

Luego de pulsar OK, hemos terminado la configuración de la lectura de datos analógicos.

Para realizar la adquisición de datos de lectura analógica; en la barra de menús Simulation del modelo, seleccionar External, como muestra la figura 16.

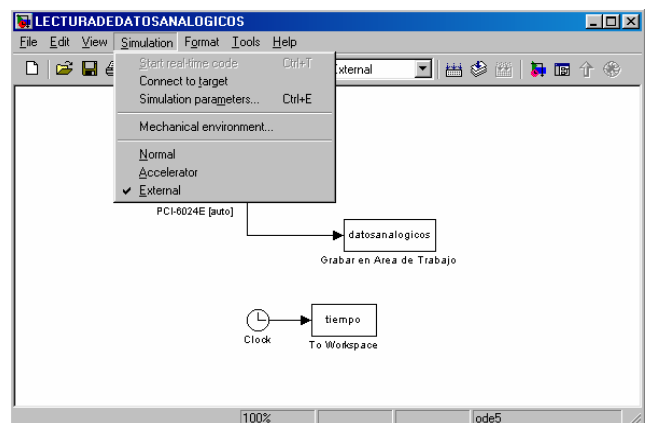


Fig. 16. Selección del modo External.

Luego debemos ordenar que se construya el modelo diseñado, para ello en la barra de menús Simulation del modelo, seleccionar Simulation Parameters y Build, como muestra la figura 17.

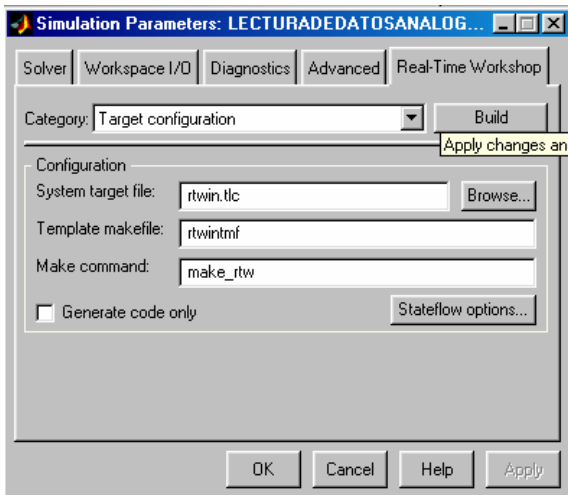


Fig. 17. Orden de construcción del modelo.

El proceso de construcción se puede observar en la pantalla del Matlab, mediante mensajes como se aprecia en la figuras 18.

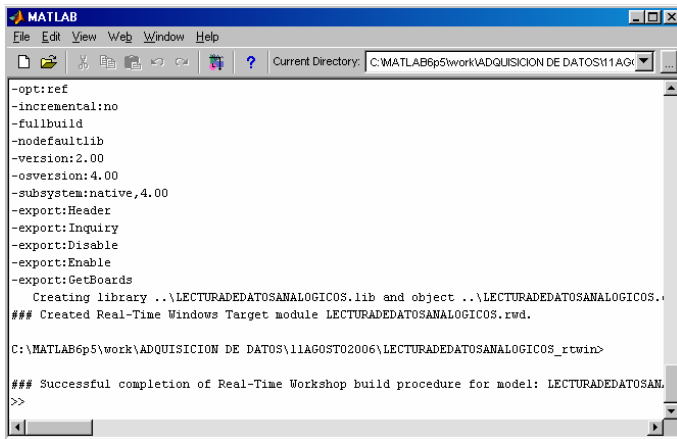


Fig. 18. Mensajes del proceso de construcción.

Durante el proceso de construcción se crean un grupo de archivos, como los mostrados en las figuras 19 y 20.

Nombre	Tamaño	Tipo
LECTURADEDATOSANALOGICOS_rtwin		Carpeta de archivos
LECTURADEDATOSANALOGICOS	9 KB	Rose Model
LECTURADEDATOSANALOGICOS.rwd	18 KB	Archivo RWD

Fig. 19. Carpetas generadas en construcción del modelo.

Nombre	Tamaño	Tipo
lexl_svr.obj	13 KB	Archivo intermedio
LECTURADEDATOSANALOGICOS	1 KB	Archivo por lotes MS-DOS
LECTURADEDATOSANALOGICOS.c	7 KB	Archivo de código fuente de C
LECTURADEDATOSANALOGICOS.h	4 KB	Archivo de encabezado de C
LECTURADEDATOSANALOGICOS.mk	15 KB	Archivo MK
LECTURADEDATOSANALOGICOS.obj	7 KB	Archivo intermedio
LECTURADEDATOSANALOGICOS_data.c	1 KB	Archivo de código fuente de C
LECTURADEDATOSANALOGICOS_data.obj	1 KB	Archivo intermedio
LECTURADEDATOSANALOGICOS_dt.h	2 KB	Archivo de encabezado de C
LECTURADEDATOSANALOGICOS_private.h	3 KB	Archivo de encabezado de C
LECTURADEDATOSANALOGICOS_targ_data_map	3 KB	MATLAB M-file
LECTURADEDATOSANALOGICOS_types.h	1 KB	Archivo de encabezado de C
modelsources	1 KB	Documento de texto
rt_nonfinite.obj	2 KB	Archivo intermedio
rt_sim.obj	4 KB	Archivo intermedio
rt_stub.obj	4 KB	Archivo intermedio
rtw_proj_tmw	1 KB	Archivo TMW
updown.obj	12 KB	Archivo intermedio

Fig. 20. Archivos generados en construcción del modelo.

Antes de ejecutar el modelo de lectura de datos analógicos, debemos ordenar la conexión con la tarjeta de adquisición de datos. Debajo de la barra de menús del modelo, seleccionar el icono de conexión de con la tarjeta DAQ, como muestra la figura 21.

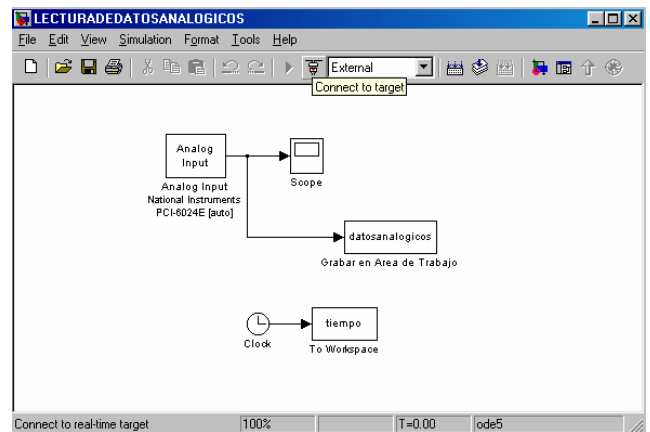


Fig. 21. Orden de conexión con la tarjeta de adquisición de datos.

La figura 22, muestra el proceso de conexión con la la tarjeta de adquisición de datos.

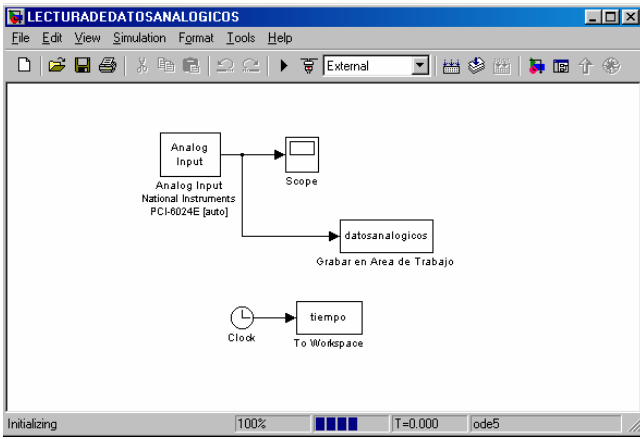


Fig. 22. Conexión con la tarjeta de adquisición de datos.

Para ejecutar el modelo debajo de la barra de menús del modelo, seleccionar el icono de ejecución del modelo (Start simulation), como muestra la figura 23.

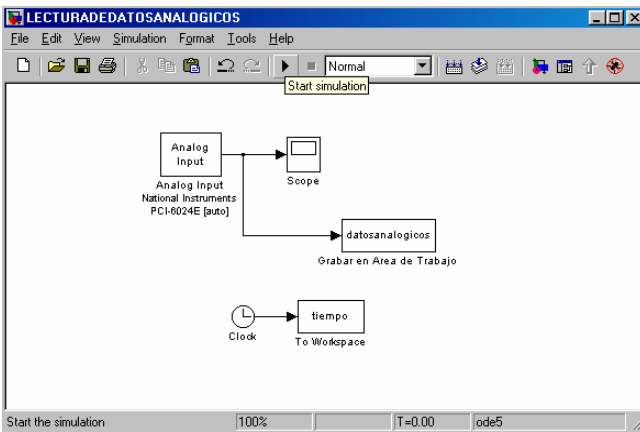


Fig. 23. Orden de ejecución del modelo.

Conforme se ejecuta el modelo, Ud. puede observar en el Scope del modelo las señales adquiridas, como muestra la figura 24.

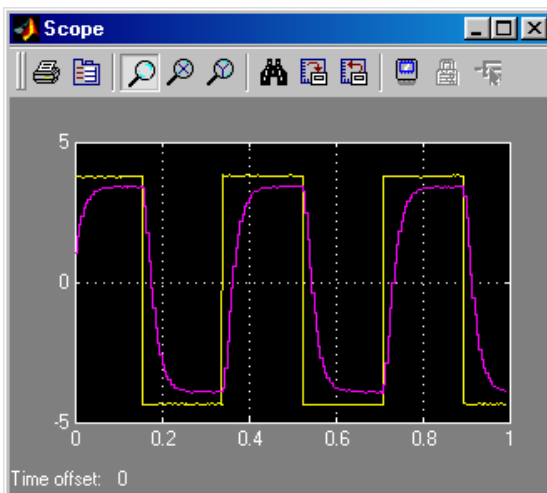


Fig. 24. Datos adquiridos que muestra el Scope.

Otra forma de obtener los graficos de las señales adquiridas, es utilizar las variables almacenadas en el área de trabajo. La figura 25, muestra el grafico de las señales adquiridas, ploteadas a partir de las variables almacenadas en el área de trabajo.

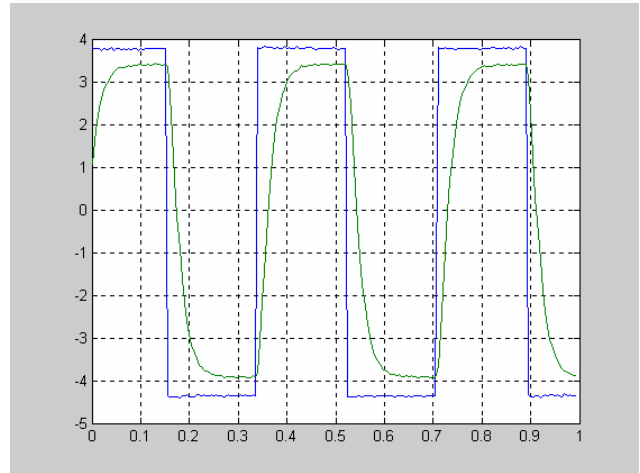


Fig. 25. Ploteo de las señales adquiridas, a partir de las variables almacenadas en el área de trabajo.

Podemos comparar las señales adquiridas por nuestro modelo de lectura de datos, comparando con las mismas señales medidas por medio de un osciloscopio, como muestra la figura 26.

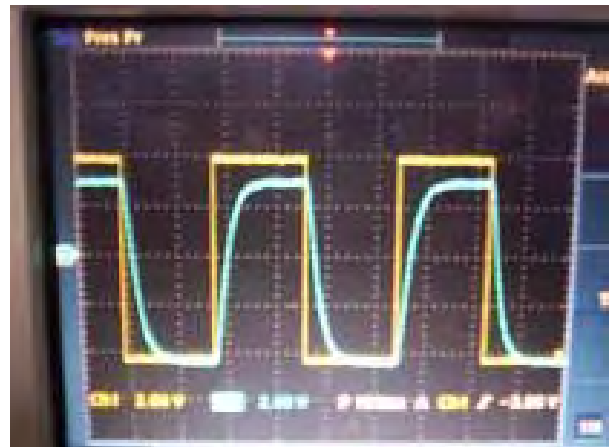


Fig. 26. Señales medidas por osciloscopio.

IV.SALIDA DE DATOS ANALÓGICOS

Para enviar datos de salidas analógicas por los puertos de salida de la tarjeta de adquisición de datos mediante Matlab, se realizan una secuencia bastante parecida al caso de lectura analógica.

Creamos un modelo nuevo en Simulink, como se muestra en ejemplo en la figura 27. De la librería Real-Time Windows Target seleccione y arrastre el bloque Analog Output, el cual nos permitirá la lectura de datos Analógicos. Elegimos la generación de dos señales senoidales de salida (la DAQ a nuestra disposición tiene dos puertos de salida analógicos). Nuevamente mediante un Scope visualizaremos los datos de salida generados. Utilizamos otra vez la opción de agregar un elemento To Workspace, para crear una variable que almacene los datos y la base de tiempos en el área de trabajo del Matlab.

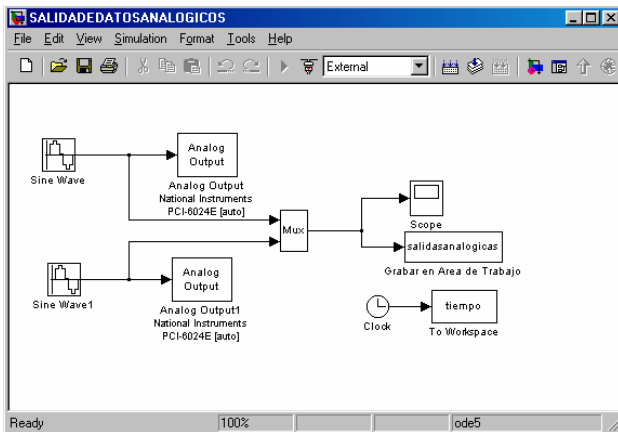


Fig. 27. Creación de modelo de salida de datos.

Debemos configurar cada bloque Analog Output, seleccionémoslo y hagamos doble clic en él. Configuramos el período de muestreo a utilizar y el canal a utilizar en la salida de datos analógicos, como muestra la figura 28.

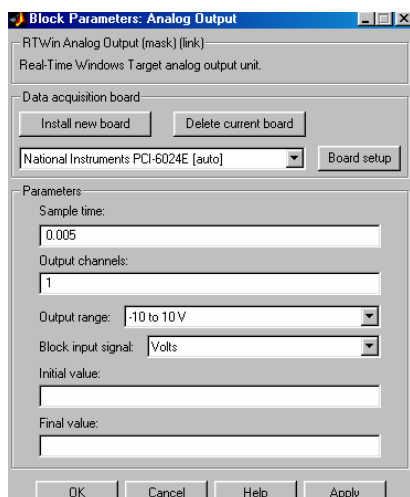


Fig. 28. Configura período de muestreo y canal de salida de datos.

Luego se realiza la misma secuencia que para la lectura de datos, hasta la conexión con la tarjeta de adquisición de datos y la ejecución del modelo de salida de datos.

Las señales generadas generadas se pueden observar en el Scope, como muestra la figura 29.

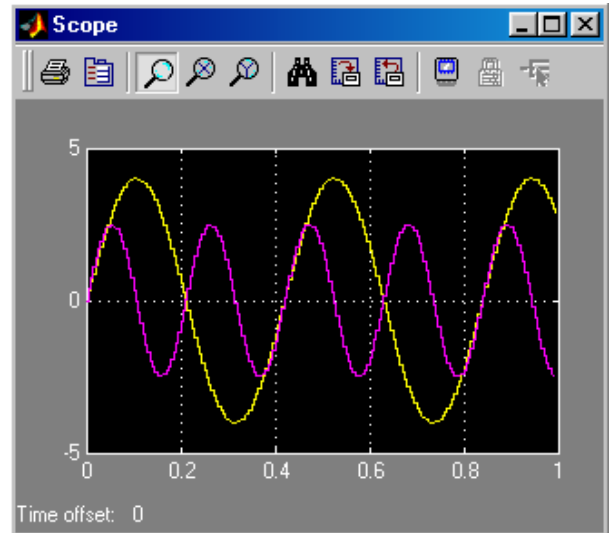


Fig. 29. Datos generados de salida mostrados en el Scope.

Las señales analógicas de salida generadas por el modelo se han medido con un osciloscopio y se muestran en la figura 30.



Fig. 30. Señales de salida medidas por osciloscopio.

V CONCLUSIONES

La adquisición de datos a través de tarjetas DAQ con Matlab, permite conectarnos en tiempo real con procesos analógicos en forma simple. Sólo se requiere que la DAQ sea reconocida por Matlab. Matlab reconoce los sistemas de adquisición de datos de la mayoría de fabricantes conocidos.

- Se debe tener cuidado en realizar la secuencia de configuración mostrada, esto asegura no tener ningún inconveniente en la captura y salida de datos analógicos.

Es conveniente activar los elementos Scope, luego de la orden de conexión con la DAQ y antes de ejecutar un modelo que contiene bloques de lectura y/o salida analógica.

- Considere que el menor tiempo de muestreo que puede utilizar es de 1 msg.

BIBLIOGRAFÍA

-Real-Time Windows Target 2.6.1. Run Simulink models on a PC in real time. Disponible en : <http://www.mathworks.com/products/rtwt> . Acceso : Julio 2006.

-Real-Time Windows Target Release Notes. Disponible en : http://www.mathworks.fr/access/helpdesk/help/pdf_doc/rtwin/rn.pdf . Acceso : Julio 2006.