



Guía experiencia 2: Adquisición, procesamiento y generación de señales por medio de computador

Guía preparada por el profesor Juan C. Agüero

para la asignatura *Laboratorio de Control Automático* (ELO-271)

16 de Abril 2024

Índice

1. Introducción	2
2. Trabajo Previo	2
3. Interacción sistema de tiempo continuo - sistema de tiempo discreto	3
3.1. Conversión análogo a digital	3
3.2. Conversión digital a análogo	6
3.3. Equipo disponible	8
3.4. Configuración Simulink-Quarc	9
4. Trabajo de Laboratorio	13
4.1. Materiales	13
4.2. Actividad 1: Configuración tarjeta de adquisición de datos	14
4.3. Actividad 2: Análisis circuito RC	14

1. Introducción

El desarrollo de la experiencia 2 busca la familiarización del estudiante con el proceso de comunicación entre un sistema real de tiempo continuo y un dispositivo disponible para interactuar con él (computador). Esta interacción requiere de una interfaz adecuada para que las señales de tiempo continuo (provenientes del sistema real) puedan ser leídas e interpretadas desde el computador (cuyo funcionamiento es en base a señales de tiempo discreto). También se requiere asegurar el proceso inverso, es decir, que las señales de tiempo discreto provenientes del computador puedan ser recibidas e interpretadas apropiadamente por el sistema de tiempo continuo. En esta experiencia usted aprenderá a configurar el computador para leer y enviar señales desde y hacia el sistema real utilizando el hardware y software disponibles. Para evaluar la correcta configuración de dicho proceso, se propone comunicar un simple circuito RC con un computador, y observar su comportamiento.

2. Trabajo Previo

Como se mencionó anteriormente, en esta experiencia usaremos un circuito RC como sistema real. Esta sección plantea una serie de tareas simples a desarrollar para refrescar su conocimiento sobre los circuitos RC. **El desarrollo de esta sección debe ser incluido en el informe final.**

Considere el circuito en la Figura 2.1 donde $R = 10\text{ K}\Omega$ y $C = 47\text{ }\mu\text{F}$.

1. Muestre que la función de transferencia $H(s)$, que relaciona $V_i(t)$ con $V_o(t)$ es de primer orden, es decir:

$$H(s) = \frac{K}{1 + \tau s} \quad (2.1)$$

y encuentre el valor de la constante de tiempo τ y la ganancia K .

2. Encuentre la respuesta a escalón en forma analítica.
3. Demuestre que para $t = \tau$ la respuesta a escalón esta cercana al 63,2% de su valor final y que para $t = 4\tau$ la respuesta a escalón esta cercana al 98,2% de su valor final.
4. Muestre (utilizando MATLAB para el informe final) la forma de onda de $V_o(t)$ cuando la entrada $V_i(t)$ es una señal cuadrada de frecuencia 0,1 [Hz] (período 10 [s]) y amplitud 2,5 [V].

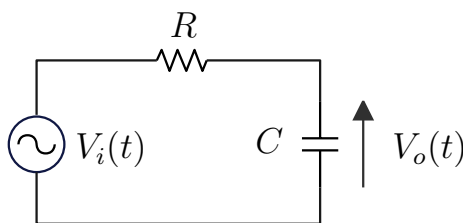


Figura 2.1: Circuito RC.

3. Interacción sistema de tiempo continuo - sistema de tiempo discreto

Cuando se desea comunicar un sistema de tiempo continuo con uno de tiempo discreto debemos garantizar la existencia de una interfaz adecuada para ello, que opere en tiempo real. Esta interfaz suele ser un dispositivo llamado conversor análogo-digital / digital-análogo, que transforma señales de tiempo continuo a tiempo discreto y viceversa (ver Figura 3.1). El nombre se debe a que las señales análogas son de tiempo continuo, mientras que las digitales son de tiempo discreto. En realidad el concepto de *señal digital* no es sinónimo de *señal de tiempo discreto*, pues la primera es más bien una representación aproximada (expresada en bits) de la segunda, sin embargo, para propósitos de este curso supondremos que ambas son exactamente la misma señal de tiempo discreto.

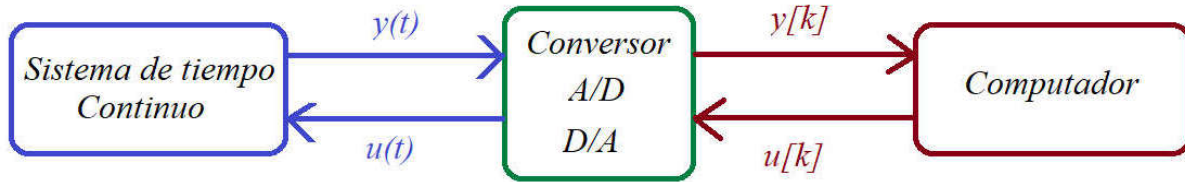


Figura 3.1: Interfaz entre sistema de tiempo continuo y sistema de tiempo discreto

A continuación detallaremos los procesos de conversión mencionados anteriormente.

3.1. Conversión análogo a digital

Considere una señal $y(t)$ de tiempo continuo. Si se desea tener una versión de tiempo discreto de ella es necesario someterla a un proceso de conversión análogo a digital, en donde se procede a tomar muestras de la señal continua cada cierto intervalo de tiempo T_s . Este proceso se llama muestreo. Luego, esta señal se transforma en una señal digital luego de los procesos de cuantización y codificación para obtener una señal digital (binaria). Este proceso se muestra en la Figura 3.2 (entiéndase *codificación* en este contexto como el paso de un valor real a su expresión equivalente en bits). Como se mencionó anteriormente, nosotros supondremos que la señal digital y la señal de tiempo discreto son iguales, por lo que el dispositivo encargado de la cuantización y codificación podría quitarse, sin embargo hemos decidido dejarlo para que usted esté consciente que son procesos que existen en la conversión análogo digital. Note la convención en la notación en la Figura 3.2: paréntesis $()$ para señales de tiempo continuo y corchetes $[]$ para denotar señales de tiempo discreto.

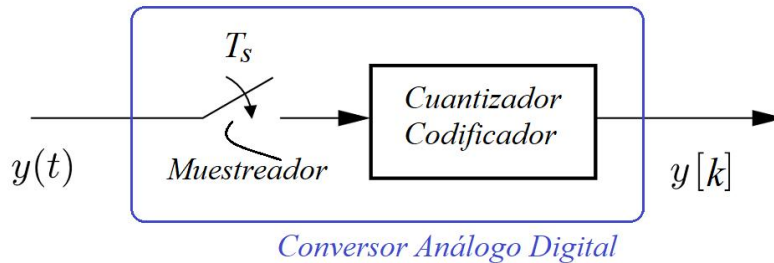


Figura 3.2: Representación matemática del proceso de muestreo de una señal continua

Para ilustrar el proceso anterior se muestra la Figura 3.3, en donde se hace una representación matemática del proceso de conversión. En la gráfica *a)*, se muestra la señal original $y(t)$ que se desea convertir a señal discreta. Para ello se define una señal de tiempo continuo, $m_{T_s}(t)$, correspondiente a un tren de impulsos unitarios (deltas de Dirac) espaciados un intervalo de tiempo constante T_s , la cual se muestra en la gráfica *b)*. Estas señales se multiplican, como se muestra esquemáticamente en la gráfica *c)*, para generar $y_{T_s}(t)$, que es una señal de tiempo continuo cuyo valor es cero en casi todo el dominio del tiempo, excepto en los instantes múltiplos de T_s , en donde se obtiene el valor de la señal original. Esta señal resultante, formada también por impulsos de tiempo continuo, es mostrada en la gráfica *d)*. Finalmente, al “reemplazar” los impulsos de tiempo continuo por impulsos de tiempo discreto (Deltas de Kronecker), se obtiene la señal de tiempo discreto $y[k]$ representada en la gráfica *e)*. Note que en dicha gráfica el tiempo pasa de forma discreta en términos del índice $k \in \mathbb{Z}$.

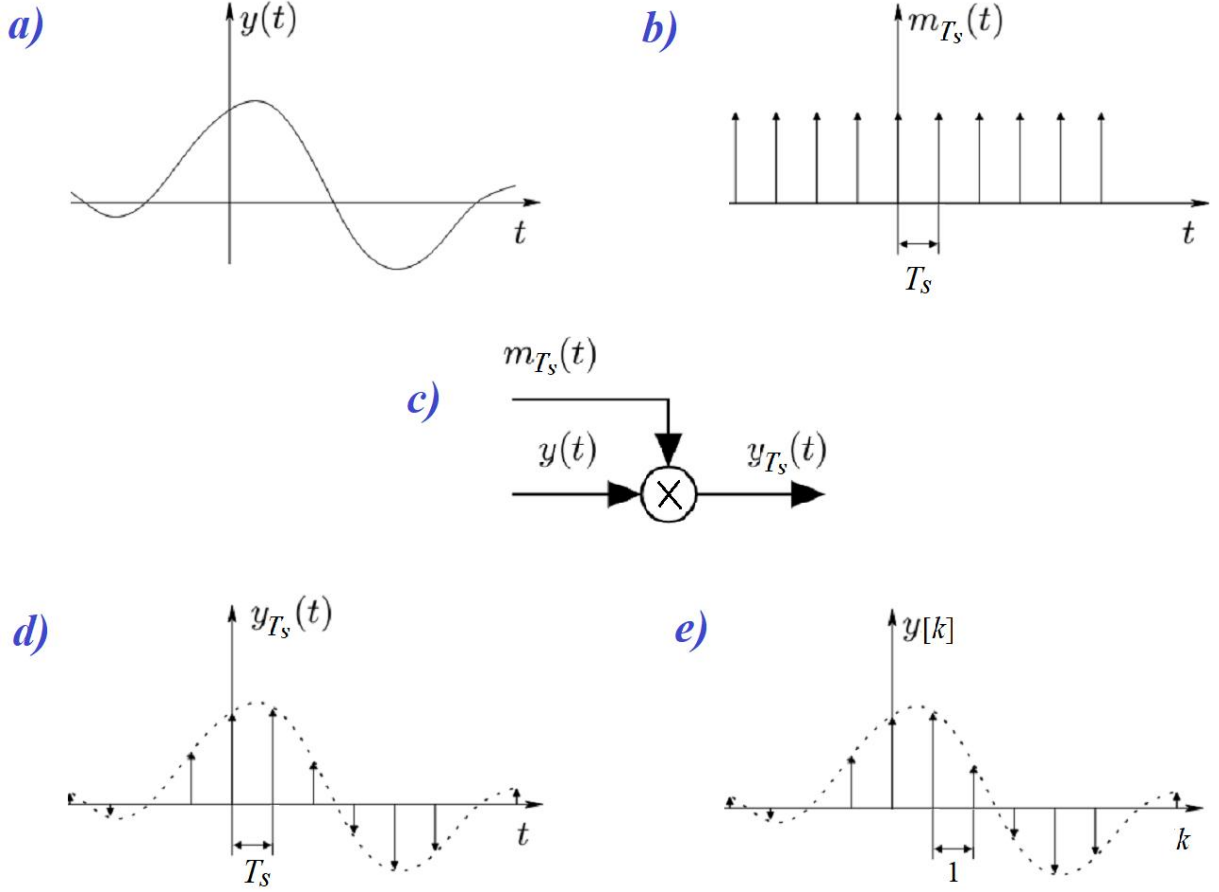


Figura 3.3: Representación matemática del proceso de muestreo de una señal continua

Es importante recalcar que el proceso descrito en la Figura 3.3 es sólo una representación teórica del proceso de muestreo, que permite hacer análisis matemáticos de dicho proceso. Nosotros no profundizaremos en ello en este curso. En el dispositivo real, en cambio, no se implementan los pasos descritos en la Figura 3.3 pues recuerde que las funciones delta de Dirac no existen en la práctica.

Cómo escoger el tiempo de muestreo T_s ?

Al convertir una señal a tiempo discreto debemos escoger de forma adecuada el tiempo de muestreo T_s , que corresponde al intervalo de tiempo entre muestras de la señal continua.

Para ilustrar lo crítico que es la elección de T_s , presentaremos 3 casos. En la Figura 3.4 vemos el caso de una señal sinusoidal amortiguada de tiempo continuo $y(t)$ y la señal discreta $y[k]$, usando como tiempo de muestreo $T_s = 0,2$. En dicha figura se aprecia que la señal de tiempo discreto, si bien no es tan parecida visualmente a la señal de tiempo continuo, aún es capaz de describir sus principales características razonablemente. En la Figura 3.5 vemos el mismo caso anterior pero ahora usando como tiempo de muestreo $T_s = 0,5$. En este caso notamos claramente que la elección de $T_s = 0,5$ nos permite “ver”, en $y[k]$, solo los peaks y valles de la señal sinusoidal amortiguada, quedándonos con una idea pobre de cómo es la señal de tiempo continuo original. En la Figura 3.6 vemos el caso con $T_s = 1$, en donde se aprecia que en $y[k]$ solo se observan los valles de la señal sinusoidal amortiguada, lo que conducirá a una conclusión errónea de sobre cómo era la señal original.

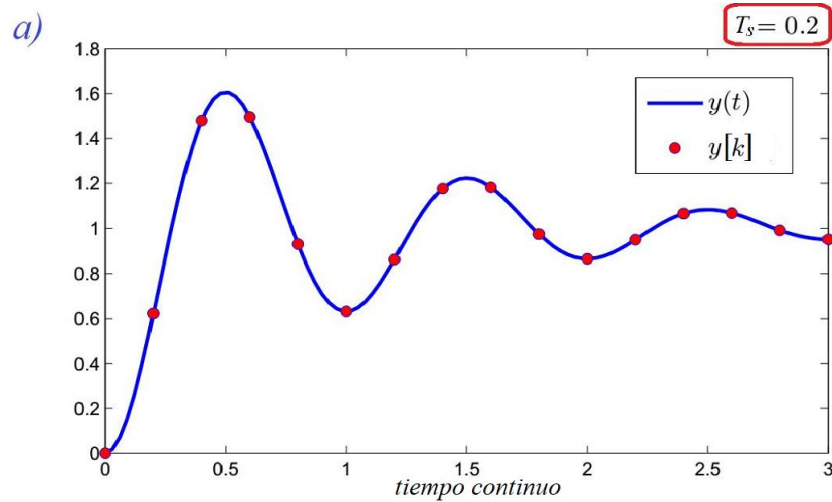


Figura 3.4: Muestreo de una señal a tiempo de muestreo $T_s = 0,2$

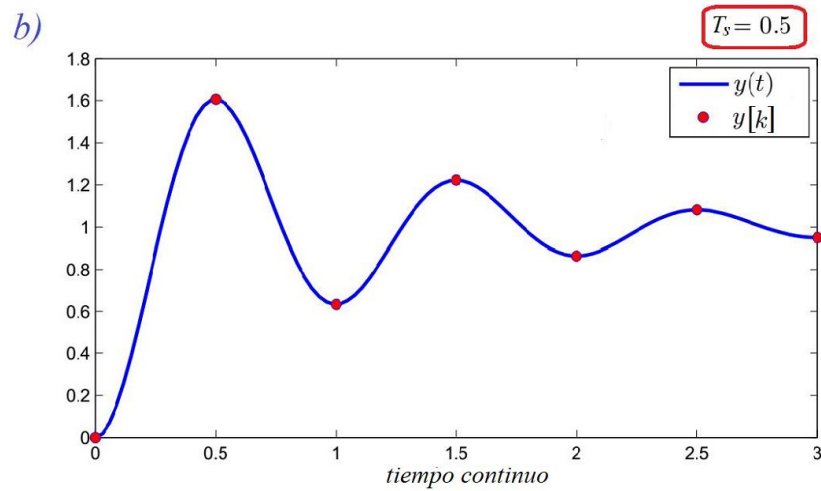


Figura 3.5: Muestreo de una señal a tiempo de muestreo $T_s = 0,5$

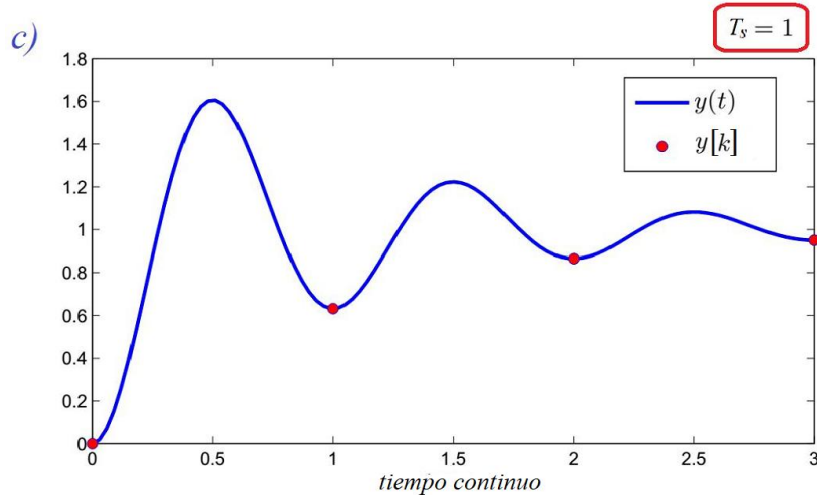


Figura 3.6: Muestreo de una señal a tiempo de muestreo $T_s = 1$

Los ejemplos anteriores indican que se debe escoger un tiempo T_s lo suficientemente pequeño. Ahora bien, para tener un criterio más concreto, podemos recordar que al aplicar la transformada de Fourier de tiempo continuo en cualquier señal acotada de tiempo continuo $y(t)$, podemos conocer su contenido en frecuencias (espectro). Es sensato entonces suponer que se debe escoger T_s de tal forma que se puedan representar las frecuencias más altas asociadas a la señal $y(t)$. Esta idea es formalizada en el siguiente Teorema.

Teorema de reconstrucción de Shannon: Una señal $y(t)$ cuyo espectro es distinto de cero solo en el rango $[-\omega_c, \omega_c]$, puede ser reconstruida de forma exacta a partir de las muestras de la señal, si estas son obtenidas a una frecuencia de muestreo $\omega_m \geq 2\omega_c$, donde $\omega_m = 2\pi/T_s$.

En el Teorema anterior, ω_m se conoce como la **tasa de Nyquist**, mientras que ω_c se conoce como la **frecuencia de Nyquist**. Este Teorema establece que, si ω_c es la frecuencia más alta asociada a una señal, entonces existe una función matemática que permite reconstruir la señal continua original de forma exacta a partir de las muestras. Este resultado establece un criterio claro para la elección de T_s , sin embargo, existen restricciones prácticas importantes por lo que no es recomendable muestrear a la frecuencia ω_m . Dentro de estas restricciones está el hecho que la función que permite reconstruir de forma exacta es una función no lineal, y no causal. Otra limitante es que las señales provenientes de sistemas reales no suelen tener un espectro como el descrito en el Teorema, y por lo tanto no se puede establecer un valor para ω_c con claridad.

Ahora bien, si la señal continua corresponde a la salida de un sistema continuo (lo que es muy común), es claro que la frecuencia más alta de la señal estará limitada por el ancho de banda de dicho sistema. Así, un criterio usual para diseñar $\omega_m = 2\pi/\Delta$ es considerar ω_m mayor a 5 o 10 veces el ancho de banda del sistema correspondiente. Qué desventaja cree usted que tendría muestrear a una tasa muy mayor (por ejemplo, 100 veces el ancho de banda?)

3.2. Conversión digital a análogo

En esta sección se describirá el proceso en el cual una señal digital se transforma en una señal de tiempo continuo. Dicho proceso es representado en la Figura 3.7, en donde se muestra que la señal digital pasa por un bloque decodificador, que transforma la señal digital a la señal de tiempo discreto (este bloque puede omitirse en nuestro análisis) y luego se pasa por un dispositivo *retentor* que transforma la señal de tiempo discreto en una señal de tiempo continuo. La forma en que opera el dispositivo retentor es bastante sencilla, pues este lee un valor de la señal discreta y congela dicho valor hasta que lee el siguiente valor de

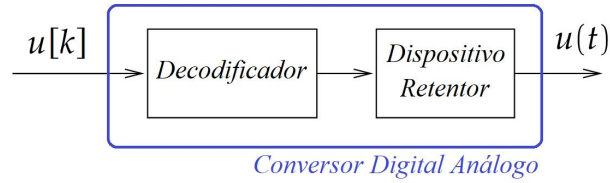


Figura 3.7: Esquema de conversión digital-análogo con retentor

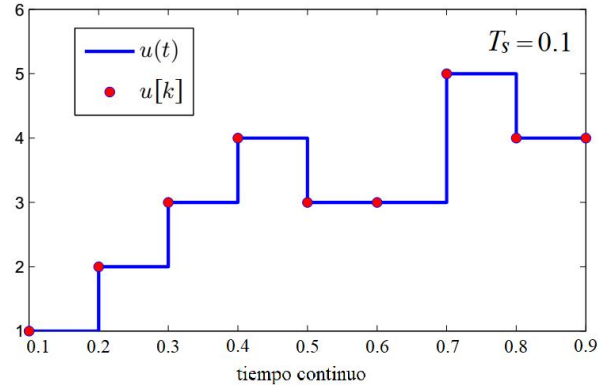


Figura 3.8: Reconstrucción con retentor de orden cero (zoh)

la señal discreta, repitiendo el proceso. Esto es descrito gráficamente en la Figura 3.8.

El dispositivo retentor previamente descrito se conoce como retentor de orden cero (de sigla *zoh*, proveniente del inglés *zero order hold*), y entrega señales continuas tipo escalera. Este dispositivo entonces se puede representar matemáticamente a través de la función de transferencia

$$R(s) = \frac{1 - e^{-sT_s}}{s} \quad (3.1)$$

Note que la función de transferencia del retentor $R(s)$ está descrita en términos de Laplace (tiempo continuo), sin embargo la señal de entrada $u[k]$ es de tiempo discreto, y por lo tanto incompatibles. Para evitar esta inconsistencia matemática, primero debemos recordar que una señal de tiempo discreto se puede representar como un tren de impulsos (deltas de Kronecker) con distintas ponderaciones. Así, podemos entonces considerar que la señal de entrada al retentor es la señal continua que se obtiene al reemplazar los deltas de Kronecker en los instante $k = 0, 1, 2, \dots$ por deltas de Dirac en los instantes $t = kT_s$.

Usted seguramente piensa (y estaría en lo correcto) que la señal de tiempo continuo proveniente del retentor de orden cero podría mejorarse si se usa otro dispositivo, por ejemplo, que use rectas con distintas pendientes para construir la señal continua. De hecho, existen otro tipos de dispositivo que tienen métodos más sofisticados para la construcción de la señal continua a partir de la señal discreta (en particular, el descrito anteriormente que entrega rectas con distintas pendientes corresponde a un *retentor de orden uno* (foh)). Sin embargo, el retentor de orden cero es uno de lo más usados por su simpleza y por que, si T_s es suficientemente pequeño, la señal continua tipo escalera obtenida funcionará sin problemas en muchos problemas prácticos.

Es importante indicar que hemos presentado la conversión D/A basados en que la señal de tiempo discreto está definida cada T_s segundos (el mismo tiempo de muestreo en la sección anterior). Sin embargo, este tiempo T_s podría ser distinto en ambos casos. Es decir, el convertor puede hacer la conversión A/D con un tiempo de muestreo T_{s1} para ser enviada al computador, y recibir posteriormente una señal de tiempo discreto desde el computador cada T_{s2} [s] para su conversión D/A. Nosotros supondremos que $T_{s1} = T_{s2} = T_s$.

3.3. Equipo disponible

Durante el presente curso ocuparemos para la conexión computador-sistema los siguientes elementos:

- **Computador:** El computador debe tener capacidad de procesamiento y memoria suficientes. El computador disponible en Laboratorio cuenta con sistema operativo Windows 10 (Figura 3.9).
- **Tarjeta de adquisición de datos NI PCI-6221:** Corresponde al dispositivo encargado de la conversión análogo-digital/digital análogo en tiempo real. Existe una variedad de Tarjetas disponibles en el mercado. En el laboratorio se cuenta con la tarjeta National Instrument PCI-6221 (Figura 3.10) que se encuentra inserta dentro del computador y se comunica al exterior a través de un puerto paralelo.

Importante: No manipule dicho puerto, ni el cable correspondiente, pues puede desconectar la tarjeta de la placa madre del computador.



Figura 3.9: Imagen referencial. El equipo en laboratorio podría diferir del de la foto.

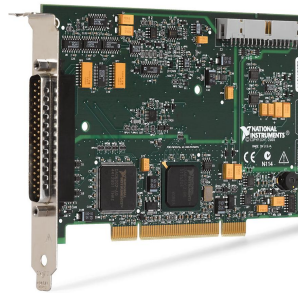


Figura 3.10: Tarjeta National Instrument PCI-6221.

- **Caja Bornes A/D - D/A:** La tarjeta de adquisición de datos se conecta, a través del puerto paralelo del computador, a una caja con bornes que facilita la conexión con el sistema real que desea estudiar (ver Figura 3.11). Esta caja cuenta con bornes para las señales análogas que usted desea enviar al computador, y también para las señales análogas provenientes del computador (son señales análogas luego del proceso de conversión D/A). También usted puede usar un puerto de red para el envío y recepción de dichas señales, en caso que el sistema real cuente con dicha interfaz.



Figura 3.11: Caja con bornes

- **Matlab/Simulink:** Software que permitirá el trabajo con el sistema real desde el computador.
- **Quarc:** Software que permite la comunicación entre Matlab y la Tarjeta de adquisición de datos.

3.4. Configuración Simulink-Quarc

Para conectar la Tarjeta de adquisición de datos con Simulink, usaremos el Software Quarc. Para ello, debemos abrir un modelo en Simulink, ir a la pestaña de librerías (Library Browser) y escoger los 3 elementos mostrados en la Figura 3.12, cuyas ubicaciones están descritas en las Figuras 3.13 y 3.14.

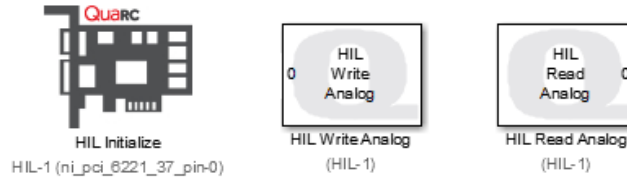


Figura 3.12: Elementos Quarc de Izquierda a derecha: Inicializacion - Escritura - Lectura

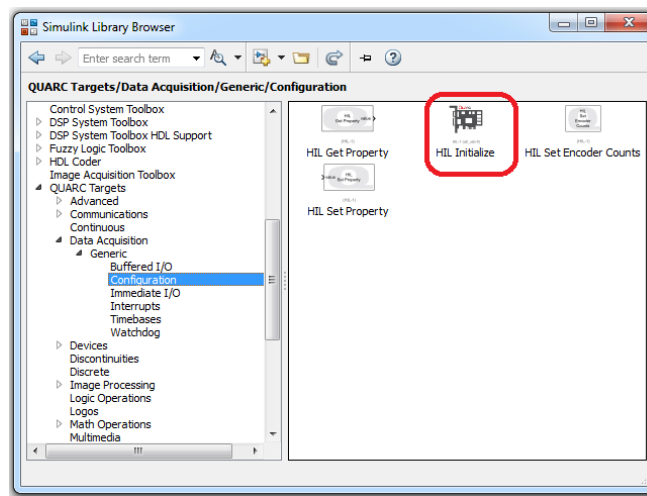


Figura 3.13: Ubicacion bloque HIL Initialize

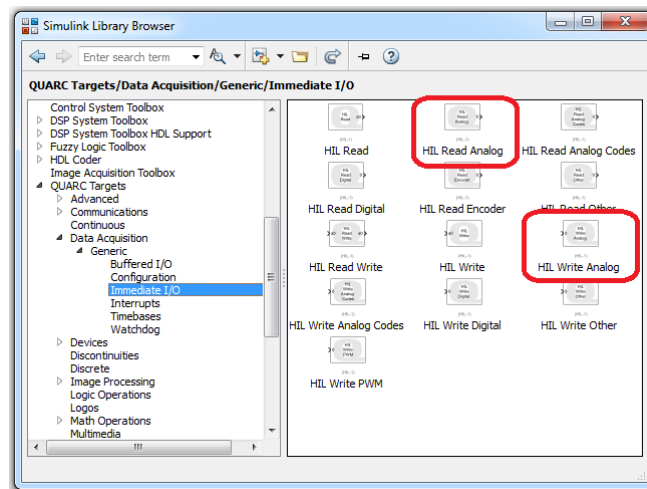


Figura 3.14: Ubicacion bloques HIL Write Analog y HIL Read Analog

En el bloque HIL Initialize se debe seleccionar la tarjeta de adquisición de datos NI PCI-6221 (ver Figura 3.15). Esto conectará la tarjeta con Matlab/Simulink. Debe además definir el número de entradas y salidas (ya sea análogas o digitales). En nuestro caso estamos interesados en entradas y salidas análogas (por qué?). Dichas entradas las puede configurar en las pestañas correspondientes (ver Figura 3.16).

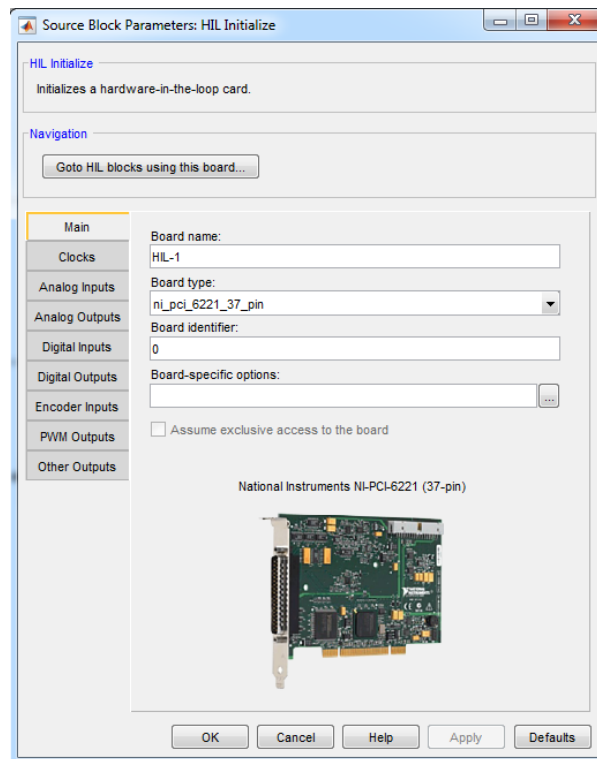


Figura 3.15: Configuración bloque Initialize

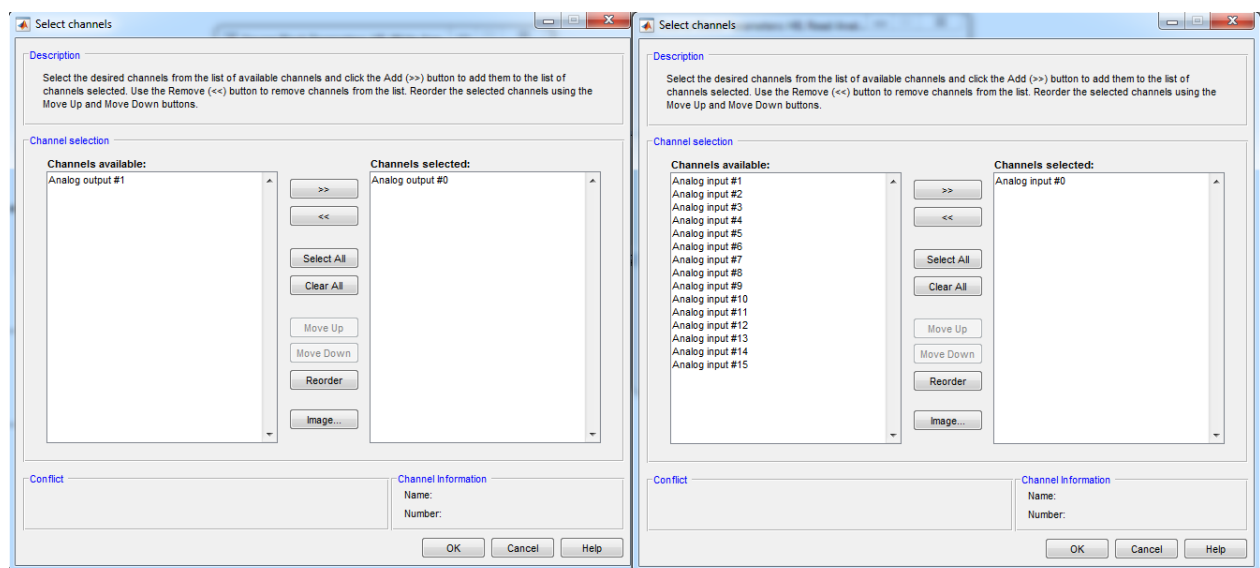


Figura 3.16: Configuración entradas y salidas análogas

Los bloques HIL Write Analog y HIL Read Analog representan las entradas y salidas analógicas de la tarjeta. Así, las entradas que usted conecte al bloque HIL Write Analog serán enviadas a la tarjeta para que se comunique con el sistema real. Por su parte, las señales que salgan del bloque HIL Read Analog en su esquema en Simulink serán las señales que se han leído desde el sistema real a través de la tarjeta de adquisición de datos. La configuración de dichos bloques incluye la definición de las señales que se leen o escriben, así como a la tasa a la que esto se hace, es decir, es donde se define T_s . (ver Figura 3.17)

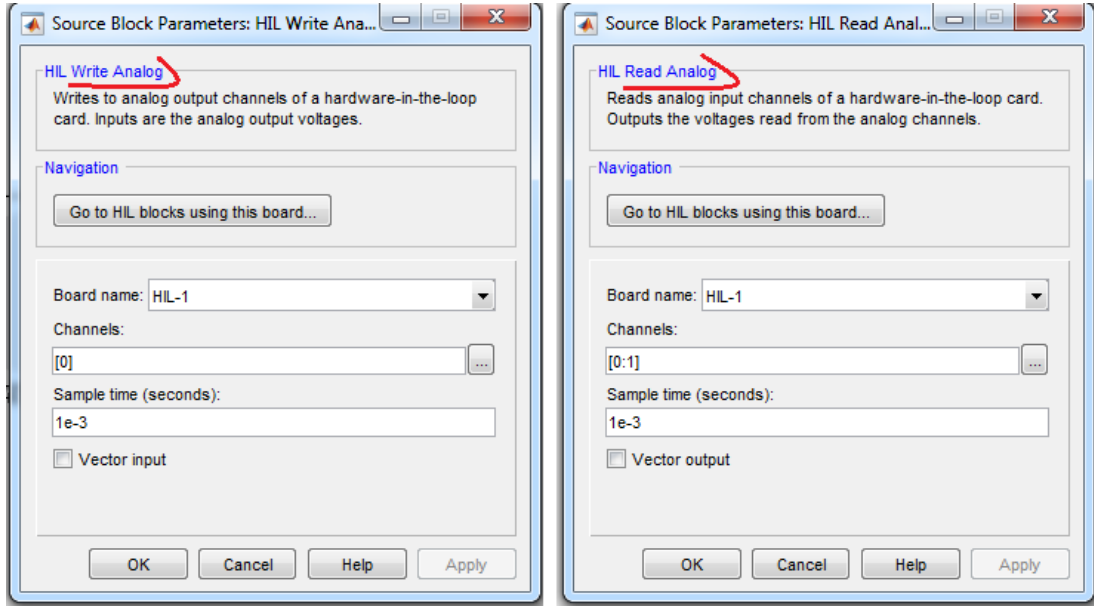


Figura 3.17: Configuración bloques lectura y escritura de entradas y salidas analógicas

Recuerde que además debe configurar el método numérico en Simulink, tal como lo hizo en la Experiencia 1. Usaremos el Método de Euler con $\Delta = 1e^{-3}$ Figure 3.18. Note que el intervalo Δ puede ser distinto de T_s , sin embargo en esta experiencia también consideraremos $T_s = 1e-3$.

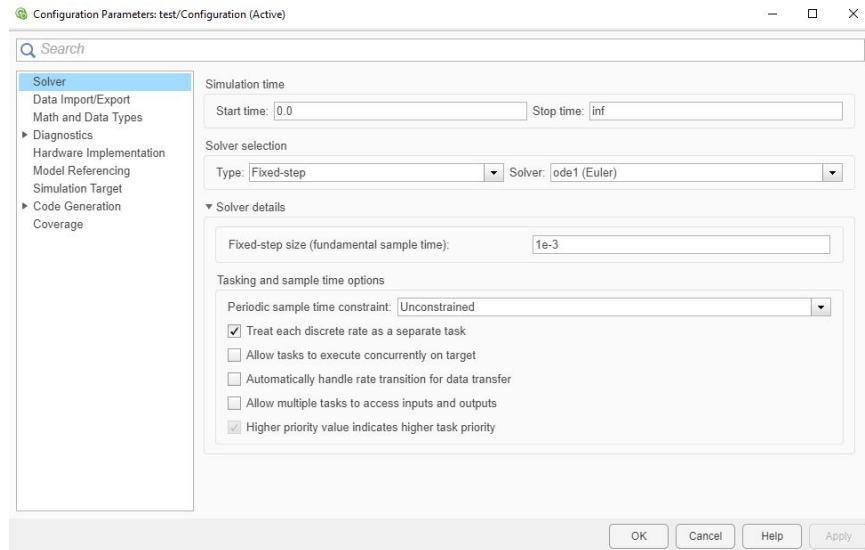


Figura 3.18: Configuración Método de Euler

Una vez que haya implementado el modelo en Simulink, es necesario crear un código de su modelo para ser enviado a la tarjeta. Para ello, usted debe definir la forma en que Simulink genera dicho código a partir del modelo visual de Simulink. Para ello, debe ir a la pestaña configuración y establecer los parámetros en las Figuras 3.19 y 3.20

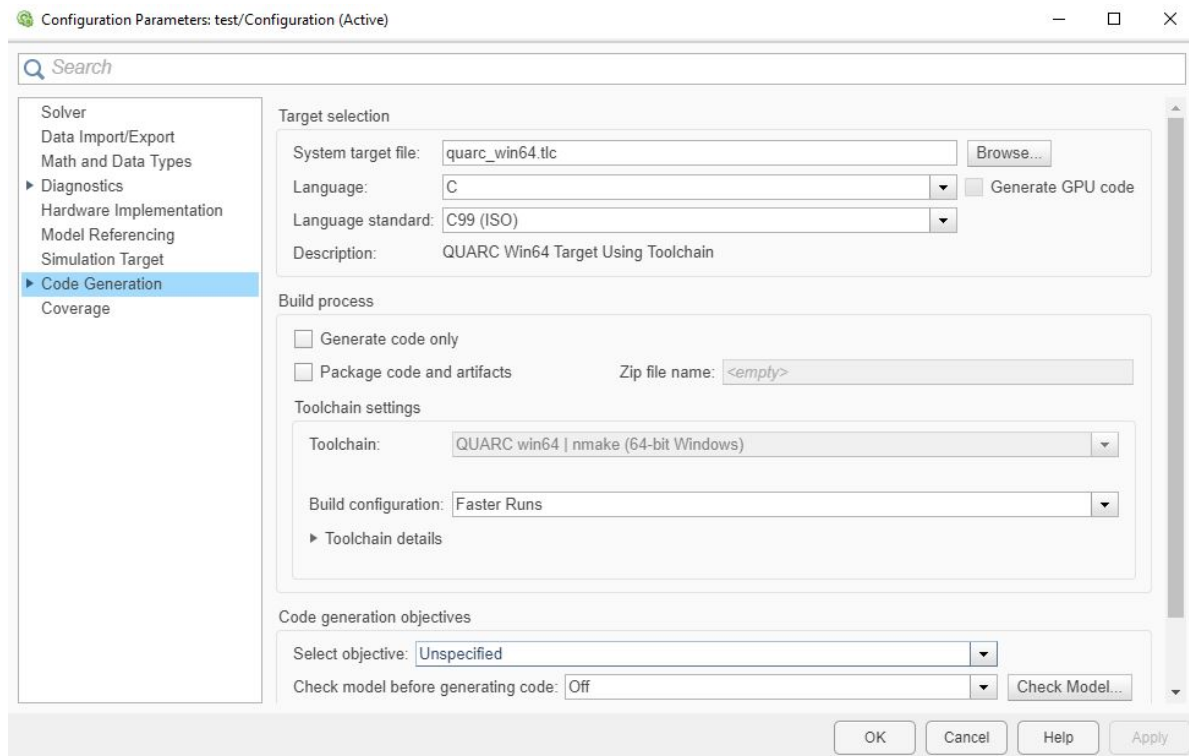


Figura 3.19: Configuración de Código

Luego, usted debe presionar el botón "build model"(con el ícono azul, en la izquierda de la foto en Figura 3.21), que creará el código en lenguaje C correspondiente a su modelo, y luego el botón a la derecha en la Figura 3.21, para enviar dicho código para la comunicación con Quarc y la tarjeta.

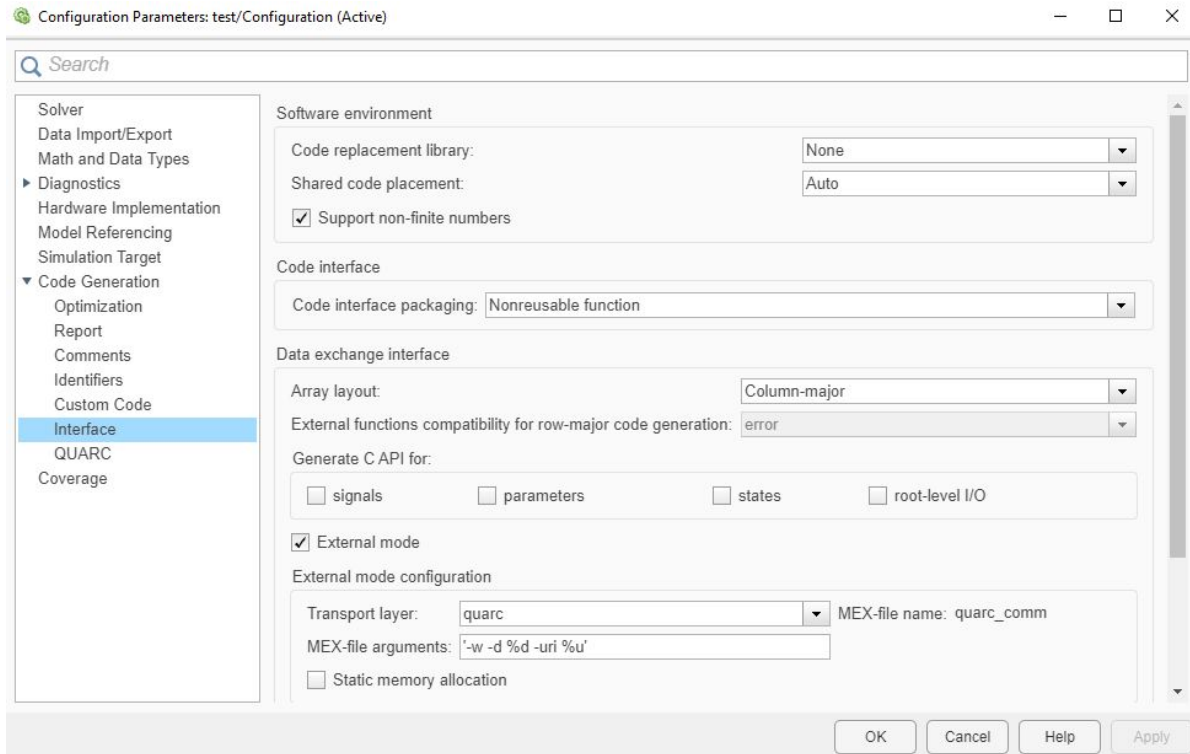


Figura 3.20: Configuración de Código

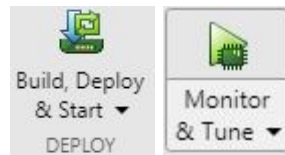


Figura 3.21: Generar código y enviar

4. Trabajo de Laboratorio

El trabajo en laboratorio consiste en 2 actividades. En la primera se busca simplemente implementar y configurar la lectura y escritura de datos usando los softwares descritos anteriormente. En la segunda se busca analizar primero un circuito RC simulado y luego un circuito RC real, usando para ello los bloques de lectura y escritura antes descritos.

4.1. Materiales

En esta experiencia se utilizarán los siguientes componentes que debe pedir en Pañol.

- 1 Osciloscopio.
- 1 Generador de señales digital.
- 1 Resistencia de $10\text{ K}\Omega$.
- 1 Condensador de $47\text{ }\mu\text{F}$.
- 1 Proto-board.
- 1 T-BNC

- 1 BNC-BNC
- 3 BNC-Caimán.
- 6 Caimán-Caimán.
- 1 Caja de herramientas.
- Alambre.

4.2. Actividad 1: Configuración tarjeta de adquisición de datos

1. Conecte la entrada analógica AI0 de la tarjeta NI PCI-6221 a la salida analógica AO0 de la tarjeta NI PCI-6221 (usando la caja de bornes). Ver Figura 4.1.
2. Implemente el diagrama en SIMULINK de la Figura 4.2. Considere para ello la configuración y procedimiento descrito en la sección anterior.

Note que en la Figura 4.2 existen dos bloques *To Host File*. Estos bloques envían la información a un archivo con extensión *.mat*, que puede ser leído por Matlab. En dicho archivo se encontrará la información de la señal conectada al bloque. Así, usted podrá graficar las señales posteriormente para comparación. Este bloque se prefiere sobre el bloque *To workspace*, pues este último presenta dificultades para trabajar en tiempo real con la configuración Quarc. Revise la Figura 4.3 para conocer su ubicación y cómo configurarlo.

3. Seleccione en el generador de funciones de Simulink una señal senoidal de período 10 [s] y amplitud 2.5 [V] ó peak-to-peak de 5[V]. Simule y almacene los datos obtenidos. Repita lo anterior para señales de periodo 1 [s] y 0,01 [s].
4. Grafique y compare las señales generadas por el bloque *Signal Gnerator* y las obtenidas desde el puerto análogo. Comente los resultados.
5. Seleccione en el generador de funciones de Simulink una señal senoidal de período 10 [s] y amplitud 2.5 [V] ó peak-to-peak de 5[V]. Cambie el tiempo de lectura y escritura (sample time) de los bloques HIL Write Analog y HIL Read Analog. Haga las pruebas con los tiempos (para ambos bloques igual): 10[s], 0.5[s] y 1e-3[s]. Almacene los datos en cada caso. Compare gráficamente los resultados y comente.



Figura 4.1: Diagrama de conexión bornes.

4.3. Actividad 2: Análisis circuito RC

1. Implemente en SIMULINK el circuito en la Figura 2.1 utilizando el diagrama de bloque en la Figura 4.4. Considere $R = 10\text{ K}\Omega$, y $C = 47\text{ }\mu\text{F}$ (use el método de Euler con $\Delta = 1e-3$). Para los puntos a) b) y c) a continuación, almacene los datos correspondientes y genere las gráficas necesarias para su análisis.

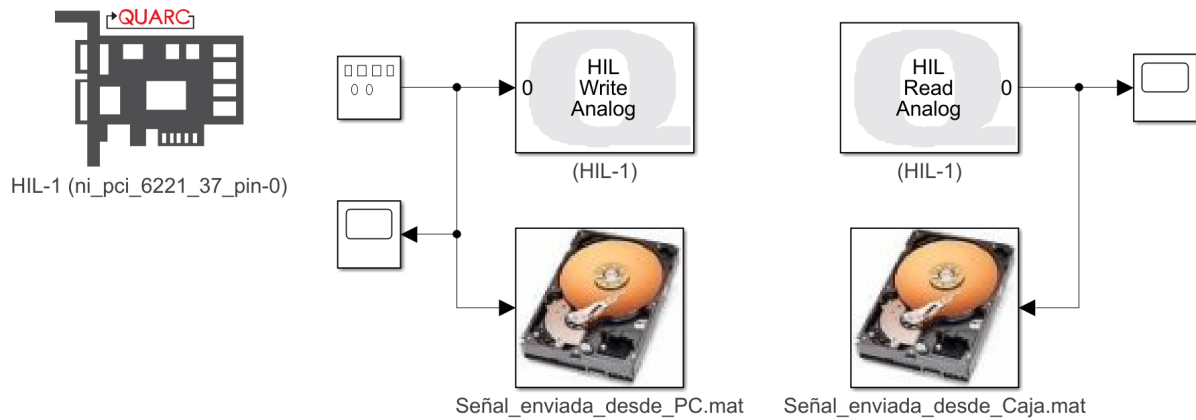


Figura 4.2: Diagrama SIMULINK para lectura y escritura de datos.

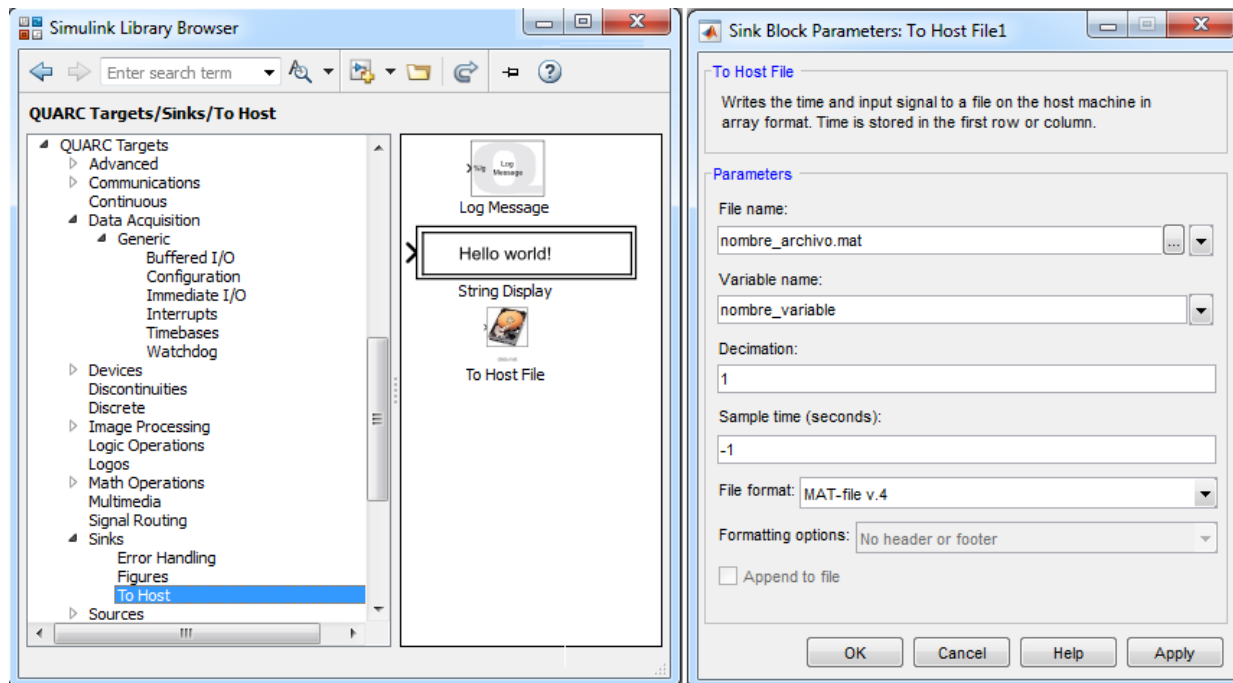


Figura 4.3: Ubicación bloque to host y configuración.

- Genere una señal sinusoidal de Amplitud 2.5[V] (o 5[V] peak-to-peak), y de Frecuencia 0.1[Hz] (o Periodo 10[s]), vea el $V_o(t)$.
 - Genere la misma señal, pero de Frecuencia 1[Hz]. ¿Que cambios se observan?, y ¿por qué?
 - Genere una señal Cuadrada de Amplitud 2.5[V] (o 5[V] peak-to-peak), y de Frecuencia 0.1[Hz] (o Periodo 10[s]), vea el $V_o(t)$ ¿Que cambios se observan?, y ¿por qué?
- Replique la configuración del punto 1.a) de esta actividad, pero ahora utilice el Generador de Señales y el Osciloscopio reales en lugar de los simulados, manteniendo el circuito RC en Simulink. Para ello siga los siguientes pasos:
 - Conecte la salida del generador de señales a la *entrada* análoga A0 de la tarjeta NI PCI-6221 y al canal 1 (CH1) del osciloscopio real. Ver Figura 4.5.

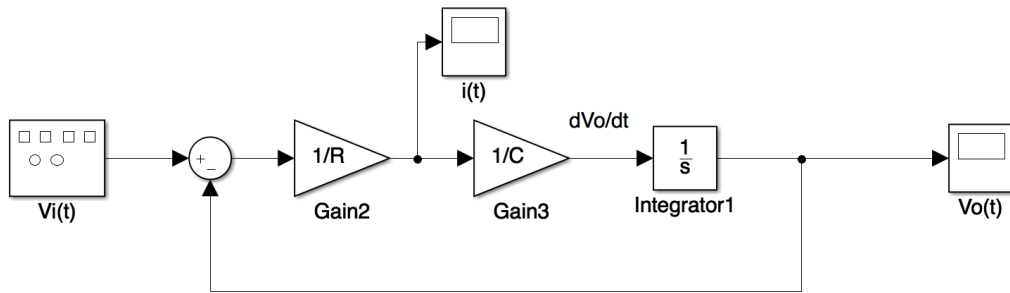


Figura 4.4: Circuito RC simulado

- b) Conecte la salida *salida* analógica A0 de la tarjeta NI PCI-6221 al canal 2 (CH2) del osciloscopio real.
- c) Implemente el circuito RC en Simulink, considerando como entrada la señal del generador real, y envíe el voltaje de salida del circuito al osciloscopio real. Considere para ello la configuración y procedimiento descrito en la sección anterior.
- d) Genere una señal sinusoidal de Amplitud 2.5[V] (o 5[V] peak-to-peak), y de Frecuencia 0.1[Hz] (o Periodo 10[s]) en el generador de señales y almacene los datos de entrada y salida que observa en Simulink en un archivo .mat. Simultáneamente observe el osciloscopio real (puede intentar sacar fotos si hace falta).
- e) Repita lo anterior con los tiempos T_s (para lectura y escritura) iguales a 10[s] y 0.5[s]. Guarde las señales de interés, gráfíquelas y comente. Compare además con los resultados del punto punto 1.a) de la presente actividad.

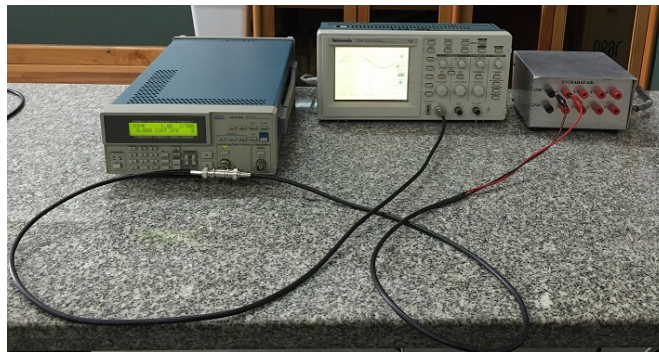


Figura 4.5: Conexiones para el experimento 2.

3. Replique el punto 2 de la presente actividad, pero ahora usando un circuito RC real en vez del simulado, y usando el osciloscopio y el generador de funciones de Simulink en vez de los reales. Para ello siga los siguientes pasos:
 - a) Arme en el proto-board el circuito RC en la Figura 2.1. Corrobore que funcione apropiadamente usando el generador de señales y el osciloscopio reales. Usea las señales definidas en los puntos a), b) y c) del punto 1 de la presente actividad. Luego, desconecte el generador de funciones real del circuito.
 - b) Conecte el voltaje del condensador real a la *entrada* analógica A0 de la tarjeta NI PCI-6221 y al canal 1 (CH1) del osciloscopio real.

- c)* Conecte la salida *salida* analógica A0 de la tarjeta NI PCI-6221 como voltaje de entrada V_i del circuito real y al canal 2 (CH2) del osciloscopio real.
- d)* Implemente en Simulink el esquema de la Figura 4.2.
- e)* Genere en simulink una señal sinusoidal de Amplitud 2.5[V](o 5[V] peak-to-peak), y de Frecuencia 0.1[Hz] (o Periodo 10[s]) en el generador de señales y almacene los datos de entrada y salida que observa en Simulink en un archivo .mat. Observe también el osciloscopio real.
- f)* Repita lo anterior con los tiempos T_s (para lectura y escritura) iguales a 10[s] y 0.5[s]. Guarde las señales de interés, gráfíquelas y comente. Compare además con los resultados del punto 2 de la presente actividad.