

PRÁCTICA 1.

Simulador Hardware-in-the-Loop y prototipo rápido de control de movimiento de eje en brazo robótico.

1. Introducción.

En esta práctica, vamos a plantear, a partir de un modelo en Simulink del sistema completo, un simulador Hardware-in-the-Loop (HiL) y el prototipo rápido del controlador de movimiento para dos articulaciones de un brazo robótico industrial.

Los robots industriales son sistemas electromecánicos relativamente complejos que suelen utilizar máquinas eléctricas giratorias –ayudadas, o no, por reductoras mecánicas- para crear el movimiento de las estructuras articuladas que conforman dicho robot. Evidentemente, las características de dichas máquinas eléctricas marcan los límites de lo que el robot puede o no puede hacer. Por otra parte, la mecánica de las articulaciones también tiene unos límites que no conviene sobrepasar. Por lo tanto, está claro que, al construir estos sistemas, conviene tener estos aspectos en cuenta, de manera coordinada. Finalmente, lo mejor es considerar también, simultáneamente a lo anterior, lo que el sistema de control que vamos a utilizar es capaz de hacer con los elementos constructivos elegidos y optimizar las características del conjunto, previamente a su fabricación (*control codesign*).

En esta práctica, vamos a partir de un modelo funcional del sistema completo para diseñar la estructura de un sistema de experimentación destinado al codiseño de un brazo robótico industrial. Dicho sistema de experimentación va a constar de un simulador HiL al que se aplica un prototipo de control de movimiento de modo que, en cierto sentido, tendremos que decidir qué elementos del modelo numérico completo del que partimos van a formar parte del simulador HiL y qué elementos se implementarán en el prototipo de control. Sin embargo, ocurre que algunos elementos del modelo que forman parte del simulador HiL son muy complicados de implementar en la práctica. En concreto, la electrónica de potencia de las máquinas eléctricas tiene dinámicas de muy alta velocidad, lo que obliga a utilizar periodos de ejecución muy pequeños (1e-6 s.) y como, además, dichos modelos numéricos son relativamente complicados, es difícil lograr una implementación para su ejecución en tiempo real. Eso nos lleva a utilizar máquinas eléctricas reales, acompañadas de su electrónica de potencia (*motor drives*), en la implementación del simulador HiL. De este modo, podemos usar periodos de ejecución suficientemente bajos como para implementar correctamente los experimentos de control en tiempo real. Eso sí, esto nos obliga a utilizar dispositivos adicionales –no recogidos en el modelo original- para aplicar, sobre cada eje, la carga que debe experimentar cada máquina eléctrica en cada momento.

Antes de seguir nos interesa recordar algunas cosas sobre las características típicas de los brazos robóticos industriales. Normalmente, son dispositivos de 6 grados de libertad (3 para el posicionamiento y 3 para la orientación), donde la posición angular de las 6 articulaciones en cada momento define el punto del espacio en el que se encuentra la herramienta y con qué orientación. Por regla general, la operación del robot consiste en seguir secuencias de movimientos para actuar sobre cierto objeto (imaginemos un robot pintando un coche, o moviendo objetos entre una posición y otra, ...) y conviene que dicha secuencia de movimientos se realice de la manera más rápida y exacta posible, respetando siempre las limitaciones de la mecánica.

Por regla general, el diseño del brazo robótico es tal –**Figura 1**- que la dimensión de los actuadores –máquinas eléctricas- acoplados a las articulaciones se va reduciendo según dichas articulaciones están más cercanas a la herramienta. Ello se debe, obviamente, a que la carga que sufre una articulación es mayor cuanto más lejos este de la herramienta y, además, dicha carga varía más en función de la posición del resto de articulaciones durante los movimientos. Por lo tanto, nuestro interés va a centrarse en las dos articulaciones que sufren más carga. Dichas dos articulaciones están actuadas por *motor drives* Brushless DC conectados a través de reductoras mecánicas de velocidad angular. Los *motor drives* ya incluyen los lazos anidados de torque y velocidad, así que el sistema de control de movimiento a implantar se reduce a dos lazos de control de posición angular. Finalmente, la carga que supone, para cada articulación, el resto del brazo robótico y su carga en cada instante del movimiento se calcula mediante un modelo altamente no lineal y extraído normalmente a partir de datos experimentales medidos durante la operación del sistema.



Figura 1. Brazo robótico industrial GMF S-360.

1. Modelo del brazo robótico y del sistema de control.

El robot que vamos a utilizar como base es uno de propósito general (GMF S-360) con 6 ejes y estructura mecánica en forma de paralelogramo –ver Figura 2-. Los tres primeros ejes (θ_1 , θ_2 , θ_3) se representan en la figura 2 y se encargan de los posicionamientos. Como decíamos antes, las articulaciones se mueven mediante motores *Brushless* DC modelados como máquinas de imanes permanentes síncronos alimentados por inversores PWM de IGBTs. Nosotros nos centraremos en las dos primeras articulaciones (eje 1 y eje 2). El eje 1 usa un motor de 2 kW y una reductora de ratio 1:130. El eje 2 está acoplada a un motor de 1 kW mediante una reductora 1:100.

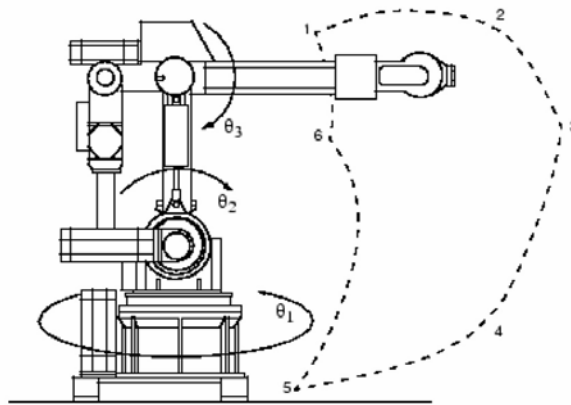


Figura 2. Estructura del brazo robótico industrial GMF S-360.

Como suele implementarse en el contexto industrial, el sistema de control consiste en tres lazos anidados en cascada: el de posición, más externo, que da la consigna al interno de velocidad, quien a su vez da la consigna al más interno, de torque. El motor síncrono de imanes permanentes se alimenta con un inversor PWM trifásico, operando en modo controlado por corriente. Se utiliza un control por orientación de campo electromagnético que utiliza la medida de posición angular del rotor en cada instante para desacoplar las variables de interés, de manera que el flujo y el torque pueden controlarse, de forma independiente, por la corriente de estator en el eje directo i_{ds} y por la corriente en el eje en cuadratura i_{qs} , respectivamente. En este caso, utilizamos la referencia de la corriente en el eje en cuadratura i_{qs}^* como variable de control (referencia de torque) y que se calcula en el algoritmo de control del lazo interno de velocidad –ver Figura 3-. Como puede verse, este planteamiento de control requiere de un sensor de posición/velocidad angular del rotor, como señal de realimentación de los lazos de posición y velocidad. Además, la medida instantánea de posición angular del rotor también es necesaria, como se dice arriba, para hacer la conversión de coordenadas (dq a abc) y poder desacoplar las variables de la maquina eléctrica mediante orientación de campo.

Todo lo descrito sobre los *motor drives* está modelado utilizando un bloque AC6 (*PM Synchronous Motor Drive*) de la librería “Electric Drives” de *Simscape*. Las entradas del AC6 son las consignas de velocidad y el torque que aparece en el eje de alta velocidad del motor. La salida principal es la velocidad que tenemos en cada instante en el eje de altas.

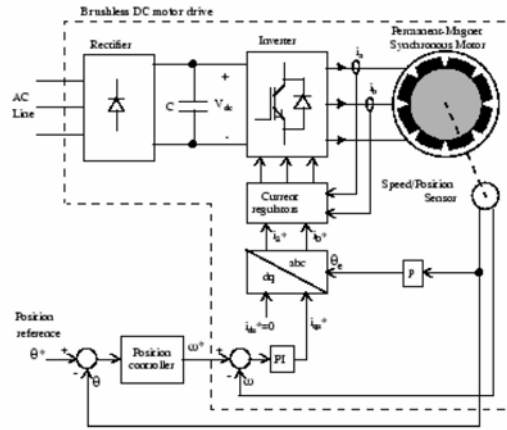


Figura 3. Esquema del control de los *motor drives*.

Las reductoras de velocidad que acoplan los motores a las articulaciones se modelan mediante otro bloque (*Speed Reducer*) de la librería “Electric Drives” de *Simscape*. Este bloque necesita, como entradas, las velocidades medidas en el eje de altas y en el eje de bajas velocidades (a ambos extremos de la reductora) y proporciona como salidas los torques en los ejes de altas y bajas velocidades. El torque aparecido en el eje de bajas es el que aplicamos al modelo de la estructura mecánica del robot para provocar su movimiento y obtener así la velocidad en el eje de bajas. El torque en el eje de altas se pasa, como entrada necesaria, al modelo del motor drive. En estas interrelaciones se observa el nivel de acoplamiento que tiene el modelo matemático utilizado. El bloque *Speed Reducer* se parametriza por su ratio y por la rigidez y amortiguamiento que presentan los ejes de entrada (altas) y salida (bajas). Estos últimos parámetros marcan las principales características oscilatorias del tren de transmisión.

Finalmente, los torques que aparecen en el eje de baja velocidad son los que provocan el movimiento en las articulaciones y, por tanto, la trayectoria dinámica del brazo. Eso sí, necesitamos calcular el torque efectivo reflejado por cada articulación. Para ello, para cada i articulación consideramos globalmente el efecto de las otras articulaciones sobre ella como una sola carga que supone un torque compuesto por tres términos

$$T_i = J_i \cdot \frac{d^2 \Theta_i}{dt^2} + C_i \cdot \frac{d \Theta_i}{dt} + G_i \cdot \Theta_i$$

donde Θ_i es la posición angular de la articulación i , J_i es la inercia, C_i en el coeficiente centrífugo y de *Coriolis* y G_i es el coeficiente gravitacional. Los parámetros J_i , C_i y G_i son funciones altamente no lineales, dependientes de la posición y/o velocidad angular de las articulaciones, que se suelen implementar utilizando polinomios o *lookup tables* –ver figura 4-. Obviamente, el torque aplicado se iguala con el torque reflejado en cada momento en cada articulación.

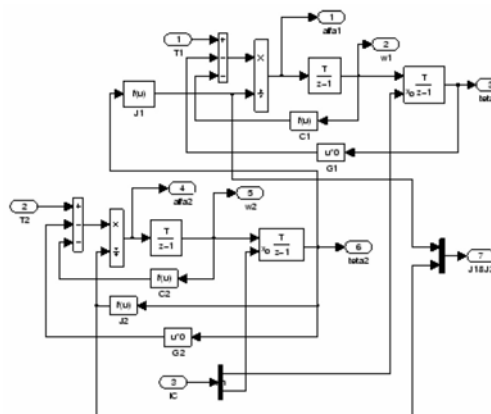


Figura 4. Esquema interno de la dinámica del Robot.

Las posiciones angulares Θ_1 y Θ_2 de las dos primera articulaciones están controladas por el lazo externo, que las obliga a seguir unas referencias calculadas por el generador de trayectorias del robot, para lograr que este implemente ciertas trayectorias. Diferentes algoritmos pueden usarse tanto para el controlador de posición, como para el generador de trayectorias. En aplicaciones de control de movimiento para robótica, se suelen usar generadores de trayectorias polinómicas que garantizan un movimiento suave del dispositivo. Por su parte, los algoritmos de control de posición

tienen como principal objetivo el *tracking* de la referencia, proporcionando la mayor robustez posible ante cargas variables. Los más utilizados son el proporcional-derivativo, el de torque computado y los adaptativos. En este caso, usaremos un proporcional-derivativo para el lazo de posición para controlar la posición de ambos ejes y un generador de trayectorias polinómicas cúbicas.

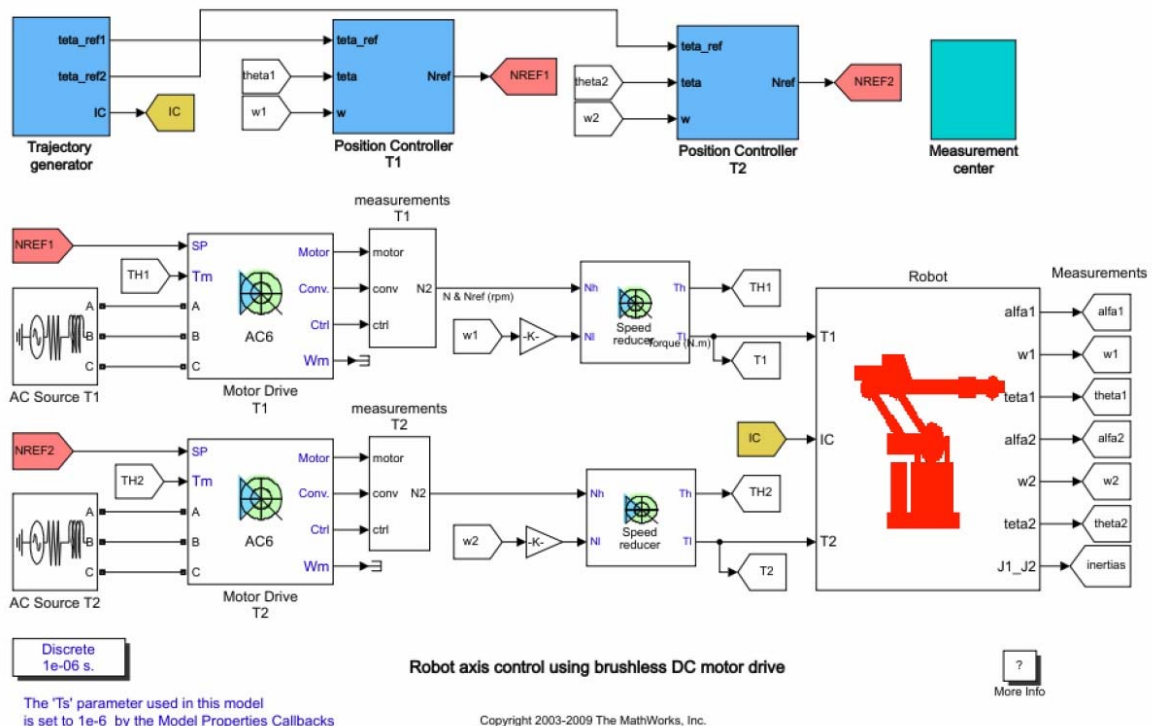


Figura 5. Esquema de *Simulink* del sistema completo.

El esquema general del esquema de *Simulink* que reúne todos los elementos se puede ver en la Figura 5 y será utilizado para probar el rendimiento del sistema controlado ante trayectorias de test. Para este experimento en particular vamos a elegir un movimiento del eje 2 desde la posición 6 a la posición 3 de la Figura 2 (Θ_2 desde $-\pi/4$ a $\pi/4$), mientras que el eje 1 rota (Θ_1 desde $-\pi/6$ a $\pi/6$) –ver Figura 6-. La duración del movimiento es de 1.5 segundos, lo que requiere de altos niveles de aceleración angular y torque. Por otra parte, la variación de la inercia que “vemos” desde el eje 1 es rápida y de valores elevados, debido al cambio de posición del eje 2. Dicha variación se modela como una función de Θ_2 que pasa de $J_1=215 \text{ kg.m}^2$ a $J_1=340 \text{ kg.m}^2$ pasando por un mínimo de $J_1=170 \text{ kg.m}^2$. Por su parte, consideramos que la inercia que “ve” la articulación 2 se mantiene constante $J_2=50 \text{ Kg.m}^2$.

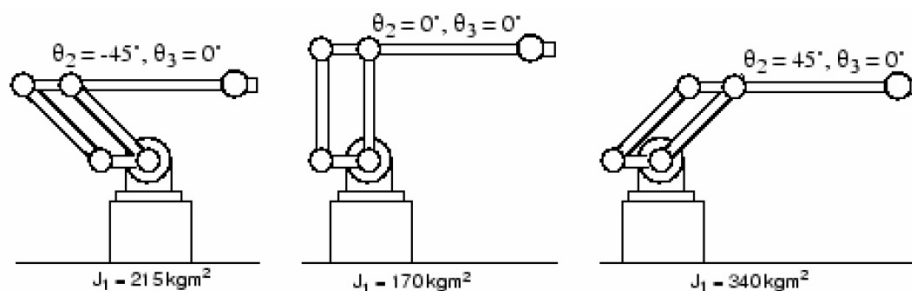


Figura 6. Cambios de estructura durante el movimiento de test.

La trayectoria de test descrita constituye una de las más exigentes para los *motor drives* de las articulaciones 1 y 2. En concreto, queremos comprobar el comportamiento del nivel de voltaje en el bus DC durante las aceleraciones y deceleraciones de los motores. También queremos ver si los torques aplicados por los motores se mantienen proporcionales a las corrientes circulando por el estator, de manera que podemos analizar el rendimiento de los algoritmos de control vectorial de orientación de campo. Finalmente, queremos estudiar la influencia del diseño de las reductoras de velocidad angular involucradas en la transmisión de las articulaciones en la aparición de oscilaciones en el brazo robótico y la posible reducción de dichas oscilaciones mediante mejoras en el algoritmo de control de posición de las mismas, usando el conocimiento de la dinámica del sistema.

3. Hardware disponible.

Como se decía al principio, en este caso vamos a utilizar hardware real como parte del simulador hardware-in-the-loop. En particular, vamos a usar dos *motor drives* reales acoplados a otros dos que les aplican el torque reflejado que podemos calcular a partir del modelo numérico comentado previamente.

El hardware que usaremos en esta práctica son equipos didácticos de *Alecop*, pero que tienen características compatibles con los dispositivos reales instalados en el brazo robótico GMF S-360. Se trata de dos equipos RMBR-900 (*Brushless*) acoplados mecánicamente a dos RMCC-900—ver manuales, para detalles—. Ambos cuatro equipos llevan lazos de par y velocidad integrados y admiten consignas externas de velocidad (o par, si el lazo de velocidad se desactiva).

La idea es utilizar los RMCC-900 en control de par y proporcionarles, como consigna externa, el torque reflejado por cada articulación y calculado, en tiempo real, a partir del modelo numérico comentado en el apartado anterior. Por su parte, los RMBR-900 se configuran en control de velocidad y se les proporciona, como consigna externa, la señal de control que genera el prototipo de los controladores de posición. Dichas señales analógicas se intercambiarán mediante el uso de sendas tarjetas NI-6024E, instaladas en dos PC targets de *Simulink Real-time*.

Además de las señales analógicas de consigna mencionadas arriba, también tenemos que intercambiar, entre simulador HiL y prototipo, las posiciones y velocidades angulares de las articulaciones 1 y 2, en cada instante, que son calculadas por el simulador y requeridas por los prototipos controladores de posición. Para dicho intercambio de señales, se propone el uso de una conexión *Raw Ethernet* entre las computadoras que hacen de *targets* de *Simulink Real-time* para la implementación de la parte numérica de simulador y prototipo. En ese sentido, ambas computadoras están conectadas mediante un cable directo RJ-45 y sendas tarjetas Ethernet, alternativas a las que utiliza la conexión con el PC *Host*, utilizado para su programación y gestión de los experimentos.

4. Objetivos de la práctica.

Los objetivos de la practica son principalmente:

- Analizar en profundidad la operación del sistema completo y estudiar los detalles del modelo que se utiliza para las simulaciones numéricas en *Simulink*.
- Extraer del modelo anterior los esquemas a utilizar para implementar el simulador HiL, por un lado, y el prototipo de control, por el otro. Obviamente, aparte del modelo utilizado para las simulaciones, necesitaremos drivers I/O, disponibles en las librerías de *Simulink Real-time*.
- Describir con detalle las conexiones a implementar con el hardware disponible para poner en marcha lo anterior.
- Valorar el uso de elementos adicionales que puedan mejorar el rendimiento, por ejemplo, filtros.
- Proponer posibles mejoras, extesiones, cambios en el enfoque, etc,...

Con todo lo anterior, hay que escribir un pequeño informe descriptivo que se entregará, a través de eGela, **antes del 6 de febrero**.