

PRÁCTICA 3.

Control en tiempo real por computadora (xPCTarget). Control de movimiento ~~-Motion Control-~~ de máquinas eléctricas (*Brushless AC*).

2. 1. Introducción.

Actualmente, se está viviendo un proceso de intenso cambio en el área del control industrial, protagonizado por los denominados PACs (*Programmable Automation Controllers*). Estos dispositivos ofrecen importantes ventajas sobre los clásicos PLCs, fundamentalmente en lo referido a la flexibilidad de programación. No en vano, los sistemas PACs tienen, en casi todos los casos, la estructura interna de una computadora personal, lo que conlleva aprovechar toda la capacidad de programación de esos sistemas. De hecho, pueden programarse exactamente igual que un ordenador personal. Aprovechando esto, hemos implementado un banco de pruebas que permite realizar experimentación remota, a través de redes TCP/IP (Intranet o Internet), sobre algoritmos de control de movimiento de actuadores basados en máquinas eléctricas.

En esta practica, vamos a implementar la programación de un lazo de control RT de posición. Cuando la programación incluye la generación de consignas –trayectorias- se llama también control de movimiento *-motion control-*. En esta práctica se plantea la programación del control de movimiento de un motor *Brushless AC*, que forma parte de un sistema industrial de posicionamiento.

Suponemos que nuestra máquina *Brushless AC* forma parte de cierta articulación de un brazo robótico o de una máquina-herramienta industrial (este tipo de actuador, debido a su compacto tamaño y su potente par, suele usarse en dichas aplicaciones). En dicho contexto, el sistema completo se dedica a mover piezas de un sitio a otro –en el caso del brazo robótico- o a implementar cierta operación sobre materiales en bruto o piezas –en el caso de las maquina-herramienta como, por ejemplo, fresadoras o cizallas-.

En cualquier caso, para cada motor implicado en el sistema, la operación se traduce en implementar diversos posicionamientos angulares –trayectorias- que siguen una secuencia determinada, normalmente cíclica –ciclo de producción- y que debe respetar una estricta sincronización temporal –para conformar correctamente el movimiento del sistema completo-, para cada producto o lote de productos.

Por otro lado, dicha secuencia se debe realizar bajo un par de carga variable, dado que este depende de numerosos factores. En el caso de los brazos robóticos, depende de la posición que tenga en cada momento el brazo y del peso de la pieza que esté trasladando entonces. Por su lado, cuando se trata de una máquina-herramienta, el par de carga depende del paso de la secuencia en el que se encuentre en cada momento. Por ejemplo, para una troqueladora, el par de carga se intensifica durante el corte del metal. Incluso varia mucho con la dureza y grosor del metal utilizado, el estado de la cuchilla y otros detalles del corte. Lo más interesante de todo esto es que, conociendo bien el ciclo de producción y el diseño del sistema completo, podemos modelizar con cierto grado de exactitud la carga que soporta cada actuador implicado, en cada momento del ciclo.

Finalmente, es evidente que, de la celeridad de los posicionamientos depende la duración de cada ciclo de producción y, por lo tanto, la productividad de la planta. Asimismo, está claro que la exactitud de dichos posicionamientos marcará la calidad del producto final.

Es por todo ello que nos planteamos implementar un banco de pruebas que pueda reproducir, con el mayor realismo posible, el ciclo de producción de los diferentes actuadores implicados en el sistema, mientras se le aplica el par de carga correspondiente. Después, estudiaremos el rendimiento del sistema utilizando diversos algoritmos de control, buscando mejorar sobretodo el tiempo de establecimiento y el error en permanente, de modo que pueda ampliarse –en la medida de lo posible, considerando las limitaciones del actuador- la productividad y la calidad final del producto.

2. 2. Hardware.

Tenemos una máquina eléctrica *Brushless* AC autopilotada –ver por ejemplo, Bose (2002), Miller (1993) y Vas (1990)- que se conecta a una etapa de potencia de 1kW gestionada por un puente de IGBTs. Asimismo, se dispone de lazos anidados de control de par y velocidad que admiten una señal externa de consigna.

Dicha máquina se encuentra conectada a un dispositivo de frenado –ver figura 3.1-. Dicho dispositivo genera un par resistivo proporcional a una señal de consigna externa.



Figura 3.1. Bancada de motores y freno.

Dos computadoras –PACs- dotadas de sendas tarjetas de adquisición (*National Instruments* NI-6221M –ver <http://www.ni.com>-) proporcionan las señales de consigna al lazo de velocidad -señal de control de posición- y al par de carga, respectivamente. Asimismo, dichas computadoras están conectadas a la red TCP/IP y tienen instalado un sistema operativo *xPCTarget* en tiempo real, lo que las prepara para ser programadas remotamente y ejecutar algoritmos de control RT.

Nota: La máquina *Brushless* AC puede sustituirse por una máquina CC *Shunt-Compound* -también aparece en la figura 3.1-, alimentada por un drive de 3A, cuando nos diese por estudiar, por ejemplo, aplicaciones de trenes de laminado siderúrgico –ver Álvarez et al. (2002), Lee et al. (2002) y Frayman et al. (2002).

2. 3. Control de movimiento -posición del motor Brushless AC.

2.3.1. Señal de control del lazo de posición.

El equipo del laboratorio dispone de lazos anidados para control de par y velocidad y proporciona la posibilidad de introducir una señal de consigna externa en voltaje. Utilizaremos los dos lazos anidados, de modo que la señal de consigna sea la referencia del lazo de velocidad. La idea es implementar con uno de los PAC el lazo más externo –el de posición–, donde la señal de control sea la consigna de velocidad –ver figura 3.2.–. En dicha figura no aparece representado el lazo más interno –el de par– que debe suponerse dentro del bloque llamado “Motor”.

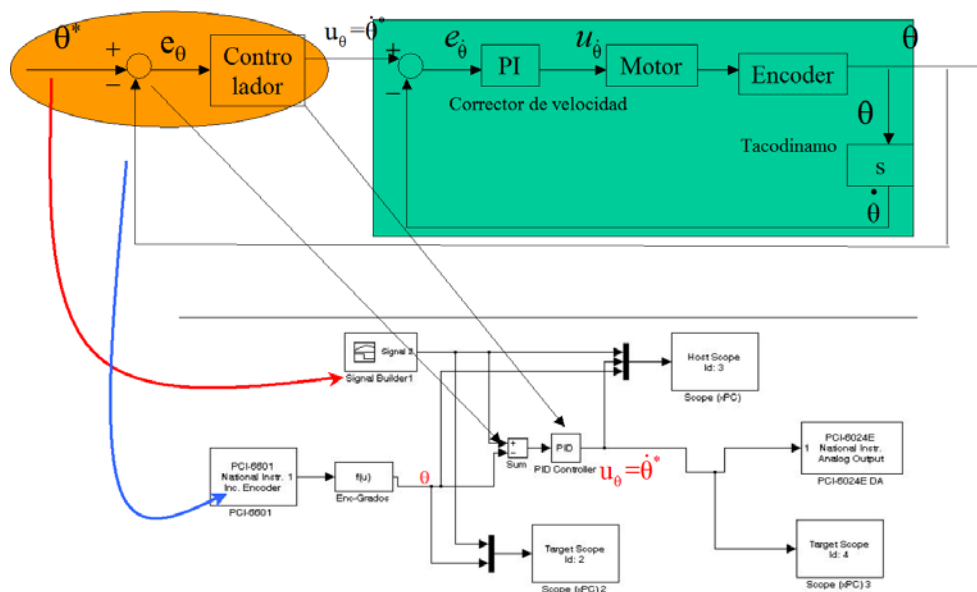


Figura 3.2. Lazos anidados de velocidad y posición.

Por lo tanto, debemos conectar la salida analógica de uno de los *xPCTargets* (PACs) –el que implementará el algoritmo de control de posición– a la entrada de consigna externa, asegurándonos que respetamos el rango del voltaje de entrada. Además, configuramos el equipo para que utilice los lazos de par y velocidad y ajustamos los parámetros de estos lazos para que el comportamiento del equipo sea el adecuado. De hecho, podemos alterar fácilmente la “planta” a controlar, cambiando dichos parámetros.

2.3.2. Señal de control de la carga variable.

Utilizamos el segundo *xPCTarget* para dar la consigna al freno de polvo magnético, conectado al motor que estamos controlando. Para ello, conectamos la salida analógica de la NI6221M a la etapa de alimentación del freno, respetando siempre los rangos de voltaje.

2.3.3. Señales de entrada.

Ambos *xPCTarget* –controlador y carga– necesitan conocer la posición angular del motor en cada instante de tiempo, de modo que conectaremos la señal de salida del *encoder* en cuadratura del Brushless a las entradas “Encoder” de ambos PACs.

2. 4. Implementación del control RT de posición de un motor AC.

Esta práctica se plantea como una secuencia de análisis-síntesis. En la primera parte, se implementa un control de posición, proporcionado por el profesor, para su cuidadoso análisis. Posteriormente, deben proponerse alteraciones o mejoras al control mediante el uso exclusivo de los recursos de programación contenidos en el entorno *MathWorks*.

2.4.1. Implementación y análisis.

En este caso, la implementación inicial se limita a la crear los ejecutables RT, a partir de los esquemas de *Simulink* proporcionados por el profesor, enviarlos al *XPCTarget* correspondiente, realizar la ejecución de los experimentos de control y recoger las señales de interés para su posterior análisis.

El lazo de posición de partida, así como la generación de la consigna de posición durante un ciclo de producción de 50 segundos se implementa en base al esquema “*Control_Mov_PI_1.mdl*”. Podemos ver dicho esquema en la figura 3.3

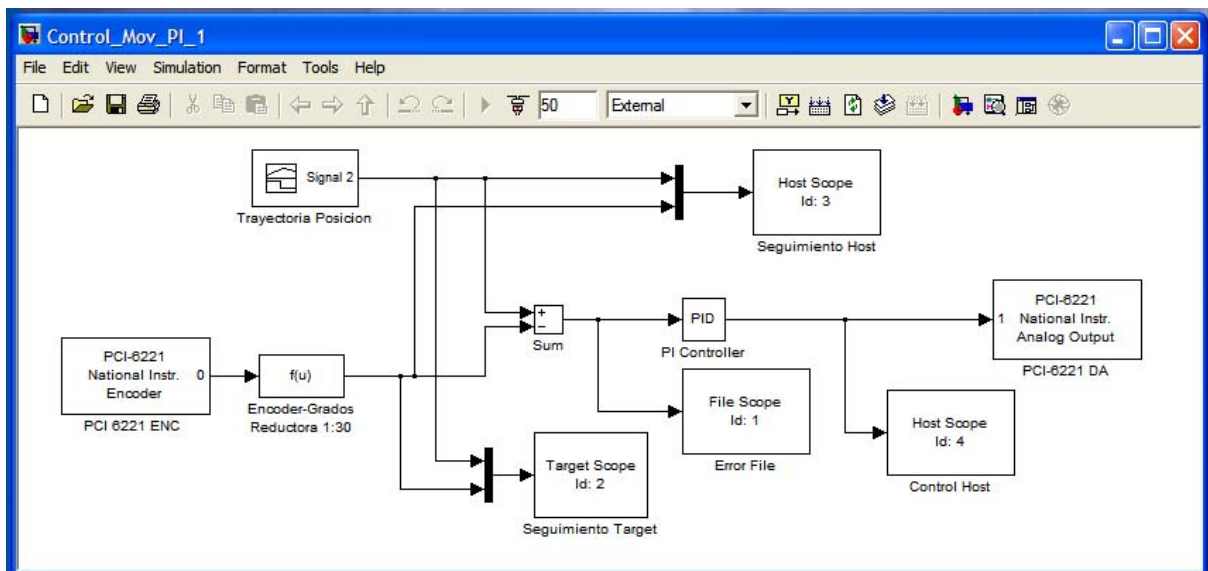


Figura 3.3. Esquema del lazo de control de movimiento.

En dicho esquema tenemos un bloque para convertir la salida del bloque “Encoder” en la posición angular en cada instante y, si es necesario, tener en cuenta la presencia de reductores mecánicos de par –utilizados a menudo para incrementar el par del actuador, “reduciendo” las vueltas que da-. Por otro lado, tenemos el bloque llamado “*Trayectoria_Posición*” que es del tipo “*Signal Builder*” de *Simulink* y que nos permite generar una consigna –de posición- que describe el ciclo de producción de nuestro actuador. Podemos observar dicho ciclo en la figura 3.4., donde vemos que enviamos al motor a la posición 350 grados hasta el segundo 5, luego a la posición 0 hasta el 10,... Evidentemente, existen tipos de consignas mucho más sofisticadas que mejoran el comportamiento del actuador y que pueden implementarse también con bloques de *Simulink*.

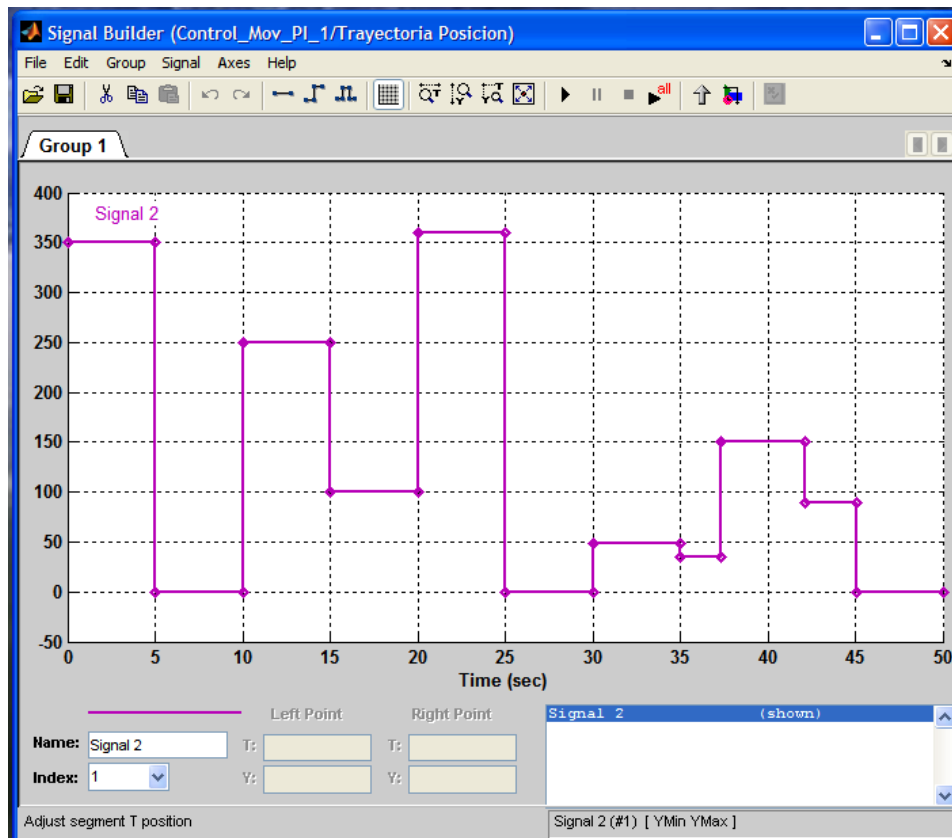


Figura 3.4. “Trayectoria” del actuador durante el “ciclo de producción”.

Podemos observar en la figura 3.3. que en el esquema se incluyen varios “*Scopes (xPC)*”, preparados para monitorizar y estudiar las señales de interés del sistema. Una vez compilado el código generado a partir de este esquema y enviado el ejecutable al xPCTarget “Controlador”, obtenemos la respuesta descrita en la figura 3.5 a partir del “Host *Scope*”, cuando ejecutamos remotamente un primer experimento de control de movimiento-sin carga-.

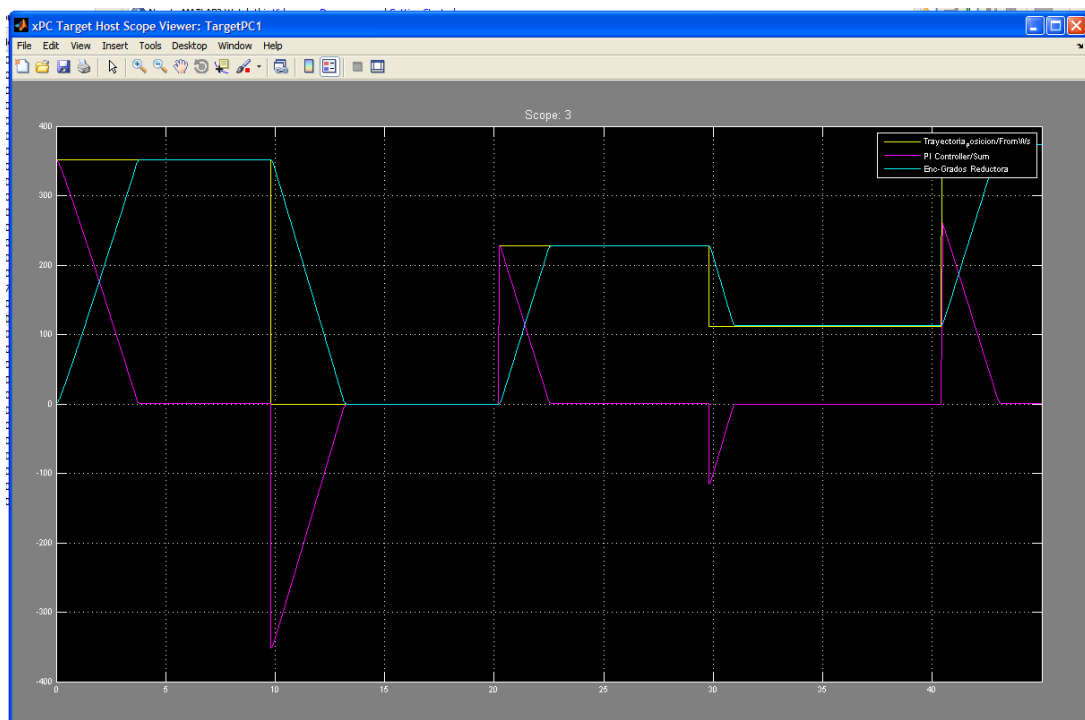


Figura 3.5. Trayectoria, señal de control y posición angular del primer experimento.

Una vez concluida la implementación de este sistema de control de partida, el objetivo es que trabajéis con él para comprender en detalle su funcionamiento en una primera etapa de análisis.

Tenéis que tener en cuenta la descripción que aparece en este guión y las señales de interés que hemos monitorizado. Podéis alterar la señal de consigna –aunque es importante que todas las consignas acaben en la posición 0, para no perder la referencia inicial- para someter al sistema a diversas situaciones de funcionamiento. También puede considerarse cambiar algunos detalles de la programación de los canales de entrada y salida...

Después de esta primera etapa, debéis ser capaces de:

- A. Describir con detalle el funcionamiento de las etapas de entrada y salida.
- B. Describir con detalle el planteamiento del lazo de control.
- C. Estudiar especialmente las características del transitorio (sobretudo, rebose y tiempo de establecimiento) y el error en permanente. ¿puedes medirlos exactamente?
- D. Contestar a las siguientes preguntas. ¿Cómo se hace la conversión de la reductora y para qué?, ¿el rendimiento es el mismo en cada sentido de giro?, ¿es el mismo en cada parte del ciclo de producción?

Una vez que entendemos con detalle cómo funciona el controlador planteado, debemos criticarlo: ¿es bueno el comportamiento dinámico?, ¿son limpias las señales?, ¿por qué? ¿el controlador está bien ajustado?....

Esta etapa es muy importante, ya que abre el camino a posibles mejoras de programación.

2.4.2. Propuesta de mejoras.

En esta última etapa, se deben proponer mejoras a la programación de partida. No hay limitaciones, siempre que no requieran cambios en las conexiones eléctricas o en la configuración inicial del equipo.

Si hace falta cambiar las conexiones o alterar la configuración del equipo, debe justificarse previamente al profesor y recibir su visto bueno y asesoramiento.

Referencias bibliográficas

- Alvarez, A. B. Diez, D. Alvarez, J. A. Gonzalez, F. Obeso, A. F, (2002) Thick unevenness compensation in a hot rolling mill having automatic gage control, *IEEE Transactions on Industry Applications*, Mar-Apr. **38** (2), pp. 559-564.
- Aström K. J. and Wittenmark B. (1990), *Computer-Controlled Systems: Theory and Design*, Prentice Hall.
- Bose B. K. (2002), *Modern Power Electronics and AC Drives*, Prentice Hall.
- Frayman, L.P. Wang, C.R. Wan Y. (2002), Cold rolling mill thickness control using cascade-correlation neural network, *Control and Cybernetics*, **31** (2), pp. 327-342.
- Lee, D. Lee Y. (2002) Application of Neural-Network for improving accuracy of roll-force model in hot-rolling mill, *Control Engineering Practice*, Apr. **10** (4), pp. 473-478.
- Miller T. (1993), *Brushless Permanent-Magnet and Reluctance Motor Drives*, Oxford University Press.
- Vas, P. (1990), *Vector Control of AC Machines*, Oxford University Press.