



El trabajo consta de dos partes: Primero, se trata de utilizar la aplicación Pascal, Figura 2, para obtener el modelo empírico de velocidad del motor hps5130 en dos pasos: a) Se debe realizar un experimento y obtener datos del motor CD; b) Se deben procesar los datos con la herramienta *ident* de Matlab. Segundo, diseñar un regulador PI en *sisotool*, Figura 8; para luego probar el diseño funcionando con la aplicación Pascal, Figura 12, en una placa de control prototipo basada en el kit059.

La primera parte del trabajo, la obtención del modelo SIMO del motor CD, ya ha sido realizada con la estructura mostrada en la Figura 2, en la cual los generadores trapezoidales junto a los sumadores han sido configurados para producir una forma de onda no periódica como se muestra en la Figura 3. El ADC_0 ha sido configurado como se muestra en la Figura 4. El ADC_1 ha sido configurado como el ADC_0; pero, con una ganancia de 0.1. Se estimuló la planta y se obtuvo la respuesta mostrada en la Figura 5. Después de procesar los datos y escalando adecuadamente las señales, se obtiene con ident de Matlab el modelo de referencia de velocidad mostrado:

$$Motor(s) = \frac{14.87}{(s + 10.4)}$$

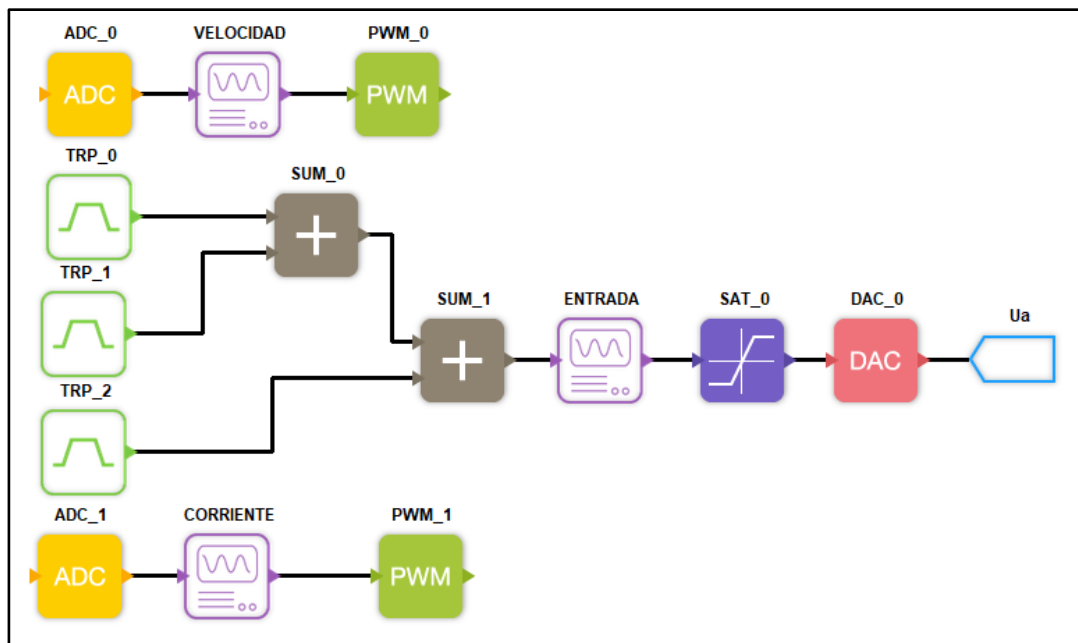


Figura 2: Estructura para estimular el motor y obtener la respuesta de velocidad y corriente

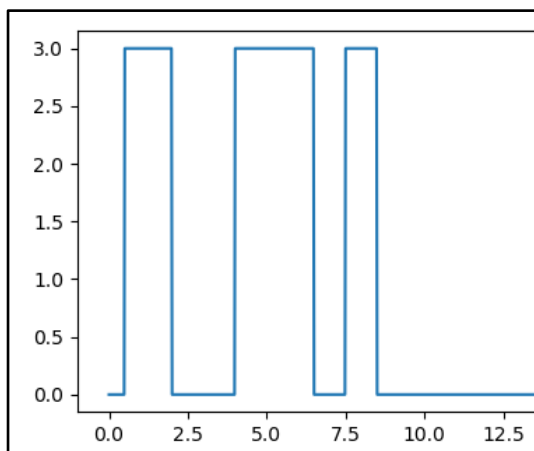


Figura 3: Estimulación a la entrada al motor con 3 V

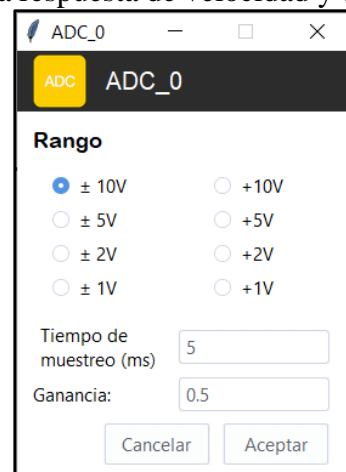


Figura 4: Configuración del ADC con entrada bipolar +/- 10 V, atenuación 2:1 y con periodo de muestreo a 5ms

Para esta segunda parte del proyecto, sus tareas consisten en:

- Utilizar la herramienta *sisotool* de Matlab para diseñar un regulador PI que haga que el sistema se estabilice en **300ms**, con un sobreimpulso M_p inferior al **3%**, error de estado estacionario de 0 ante una entrada escalón y con capacidad para eliminar las perturbaciones de entrada o salida a la planta.
- Verificar el diseño usando *simulink* ante entrada escalón y perturbaciones.
- Descomponer el regulador obtenido a forma paralela y encontrar las constantes K_P y K_I .
- Probar el control de velocidad angular constante ante una entrada de prueba escalón de 4krpm y verificar la eliminación del efecto de las perturbaciones introducidas.

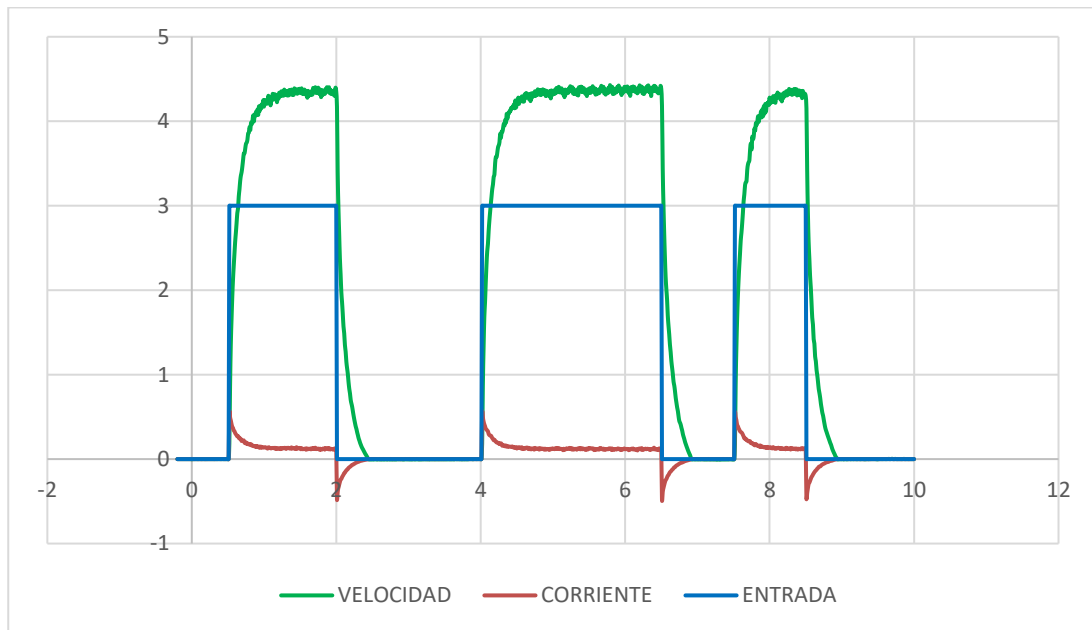


Figura 5: Captura de la entrada, u_a en [V], medida en el punto (3), antes del amplificador; y la velocidad angular, ω en [1V/krpm], medida con un atenuador 2:1 en el punto (6) y la corriente i_1 , medida en el punto (7) del sistema motor hps5130 de la Figura 1. Esto con el interruptor (4) hacia la derecha.

Parte I: Descripción del entorno del regulador PI

Objetivo: Conocer el entorno del regulador PI

$$PI(s) = \frac{K_c(s-z_0)}{s} = K_p + \frac{K_I}{s}$$

El sistema microprocesador kitprog059 a utilizar ha sido adaptado para medir señales de +/-10 V y también puede producir señales bipolares de +/-10 V. Se utilizará la planta hps5130, controlando el motor con una señal analógica de +4 V; y se usará el amplificador integrado de la planta, el cual posee una ganancia de 2.4 desde la entrada de +/-5 V. La salida del tacómetro tiene una ganancia de 2V/krpm, por lo que introducimos un atenuador de 0.5 en el sistema, éste está definido en el ADC_0 a utilizar; de tal manera, que las señales de hasta 10V producidas por el tacómetro sean presentadas por el ADC al sistema microprocesador como señales de hasta 5V como se muestra en la Figura 19.

En la Figura 6 se muestra el sistema equivalente, con realimentación unitaria, y para el cual se diseñará el regulador PI digital.

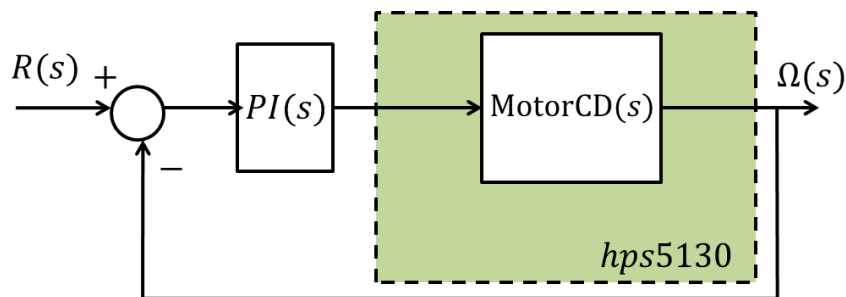


Figura 6: Diagrama de control simplificado del sistema motor hps5130

Parte II: Diseño y simulación de un control PI para el motor CD

Objetivo: Calcular un regulador PI para el motor CD que elimine las perturbaciones

Se utilizará un regulador PI con un tiempo de muestreo de **5ms**, ajustado por el sistema microprocesador a utilizar para el control, al ser definido en la configuración del ADC_0, Figura 4 o en la configuración del PID, Figura 15, y que ha sido determinado por la condición de tiempo de estabilización requerida de **300ms**. Con este periodo de muestreo se tendrán 60 muestras durante el tiempo de estabilización. Los valores usuales van desde 32 hasta 75 muestras durante el tiempo de estabilización, siendo usual un valor intermedio de 40.

Procedimiento:

Paso 1) **Creación del modelo en Matlab.** Cree primero la variable "s" con la orden `tf('s')`. Luego escriba el modelo para el motor CD mostrado en la ecuación.

NOTA: Todos los equipos usarán este modelo para poder detectar diferencias en el control diseñado por cada uno, si es que existen y para evitar arrastrar problemas de estimación o identificación desde la primera parte del proyecto.

$$Motor(s) = \frac{14.87}{(s + 10.4)}$$

Paso 2) **Apertura de la aplicación sisotool en Matlab.** Ejecute la orden

`sisotool('rlocus', MotorCD) .`

Paso 3) **Establecimiento de las condiciones de diseño en sisotool.**

a) En la ventana del *sisotool* haga **click** derecho sobre el área blanca, luego seleccione un nuevo requisito de diseño. Ajuste el requisito de tiempo de estabilización del 2% a $< 300\text{ms}$.

b) Repita el procedimiento para establecer un nuevo requisito de diseño; pero, esta vez seleccione el % de sobreimpulso y ajústelo a $< 3\%$.

Como se muestra en la Figura 7, las condiciones de diseño serán representadas por una recta perpendicular al eje real negativo, para el tiempo de estabilización y por una recta radial, para el sobreimpulso.

Paso 4) **Sintonización del compensador.** Agregue un regulador PI formado por un polo en $s = 0$ y un cero en los alrededores del polo dominante del sistema en lazo abierto, (hay tres opciones: un poco a la derecha; exactamente sobre el polo; o un poco a la izquierda del polo dominante de lazo abierto). Puede editar el compensador de forma gráfica, insertando y luego arrastrando con el ratón los polos y los ceros; o dando **click** derecho en el área blanca, abriendo la ventana de edición del compensador y escribiendo el valor del polo o cero directamente en forma numérica.

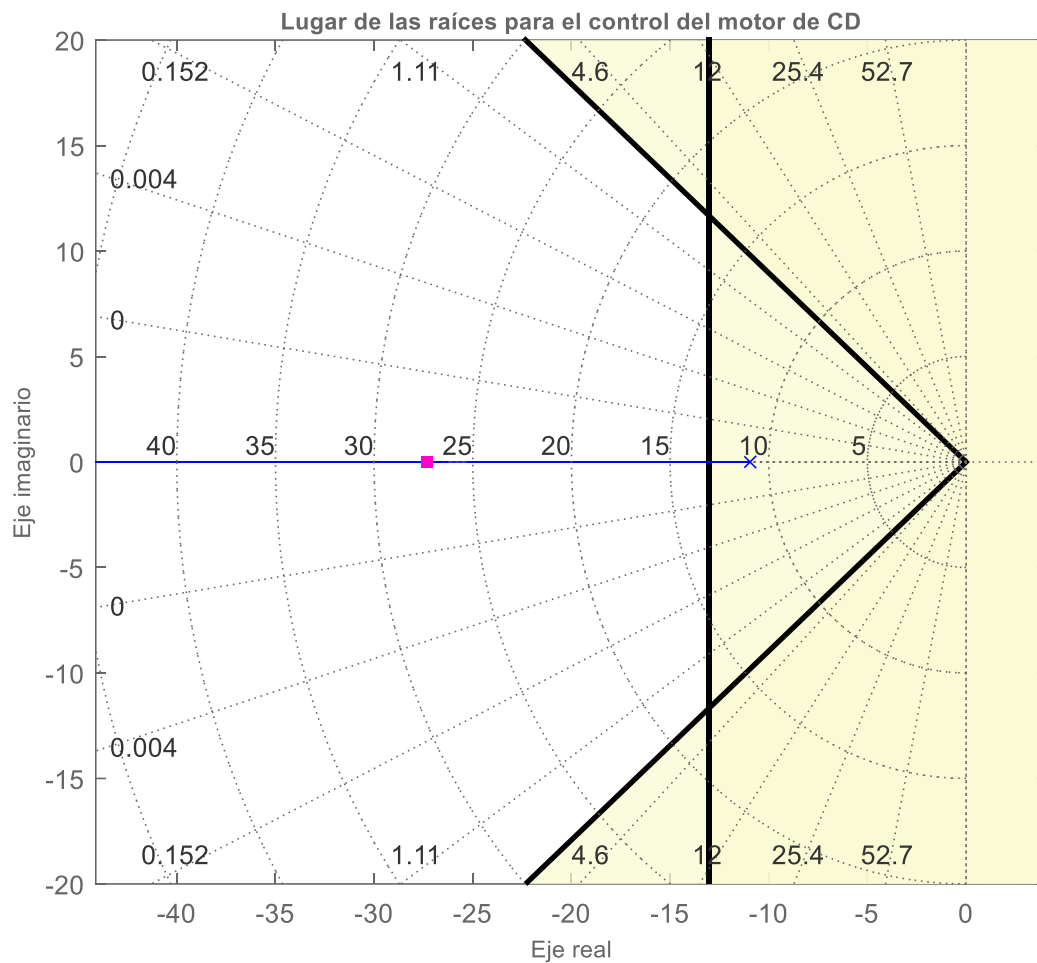


Figura 7: Definición de las condiciones de operación ($t_{s2\%} < 0.3$ s y $M_P < 3\%$) deseadas para el motor CD en *sisotool*

Paso 5) **Ajuste la ganancia del compensador.** Puede hacerlo de forma gráfica arrastrando uno de los polos de lazo cerrado del sistema, representados por un rectángulo “rosado”, puede ser necesario que haga una ampliación (*zoom*) del área de interés del lugar de las raíces. Para obtener un sistema resultante con un tiempo de estabilización del 2% menor a 300ms, como se muestra en la Figura 8, arrastre el polo de lazo cerrado hacia la izquierda de la recta que define el tiempo de estabilización. Observe, mientras arrastra el polo de lazo cerrado en el lugar de las raíces, la simulación de la respuesta ante escalón del sistema, como se muestra en la Figura 8, hasta obtener el tiempo de estabilización requerido. Alternativamente puede ajustar la ganancia de forma numérica en la ventana del editor del compensador.

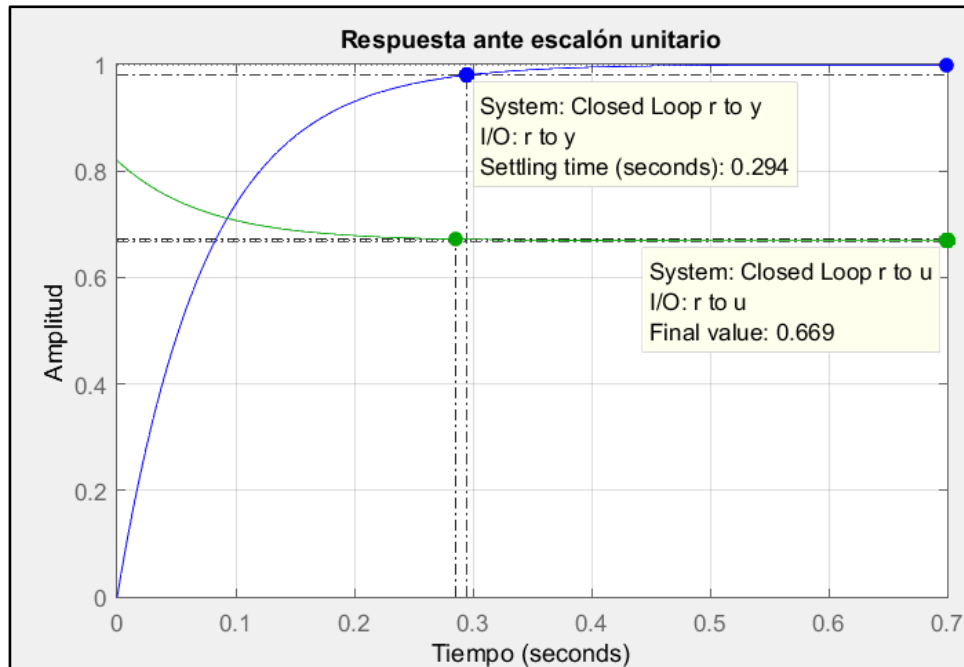


Figura 8: Simulación en *sisotool* del control con PI(s) para la planta hps5130. Note el tiempo de estabilización de 300ms para la velocidad y el valor final de la acción de control en 0.67V.

Paso 6) **Exportar el diseño a Simulink**. Cuando haya finalizado con la sintonización del regulador PI(s), exporte el diseño en *sisotool* al entorno de *Simulink* como se muestra en la Figura 9. Responda si a la pregunta de si desea escribir al entorno de Matlab las variables del diseño. Estas variables son llamadas F, H, C y G para el pre-regulador, el sensor, el regulador y la planta respectivamente.

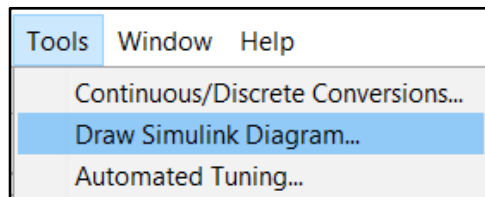


Figura 9: Exportar diseño de *sisotool* a *Simulink*

Paso 7) **Simulación**. Simule su sistema regulado con PI(s) en *Simulink* y verifique que se mantiene el sistema bien compensado aun en presencia de perturbaciones de entrada y de salida (relativas a la planta). En la Figura 11 se muestran los resultados de la simulación. Note que *Simulink* produce figuras con el fondo negro. Para graficar con el fondo blanco debe ajustar las propiedades del *scope* de *Simulink*.

Paso 8) **Descomposición en paralelo**. Finalmente descomponga el regulador PI(s) a forma paralela; a mano o usando la función *residue* de Matlab. Luego simule en *Simulink* el compensador PI(s) paralelo así obtenido. Verifique como en el paso 7. Vea la Figura 10 para una implementación paralela de un PI (si usa otra implementación para el PI, asegúrese de que sea equivalente a la forma mostrada). Las constantes mostradas **Kp** y **Ki** representan las que Ud. habrá encontrado en el paso 5 y que habrá exportado en el paso 6.

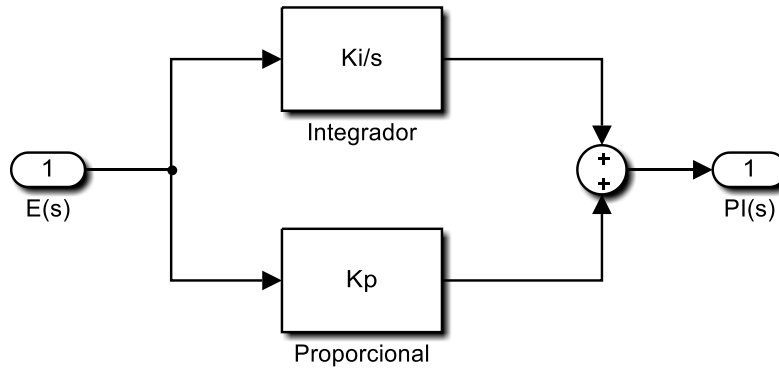


Figura 10: Implementación de un regulador PI(s) en forma paralela.

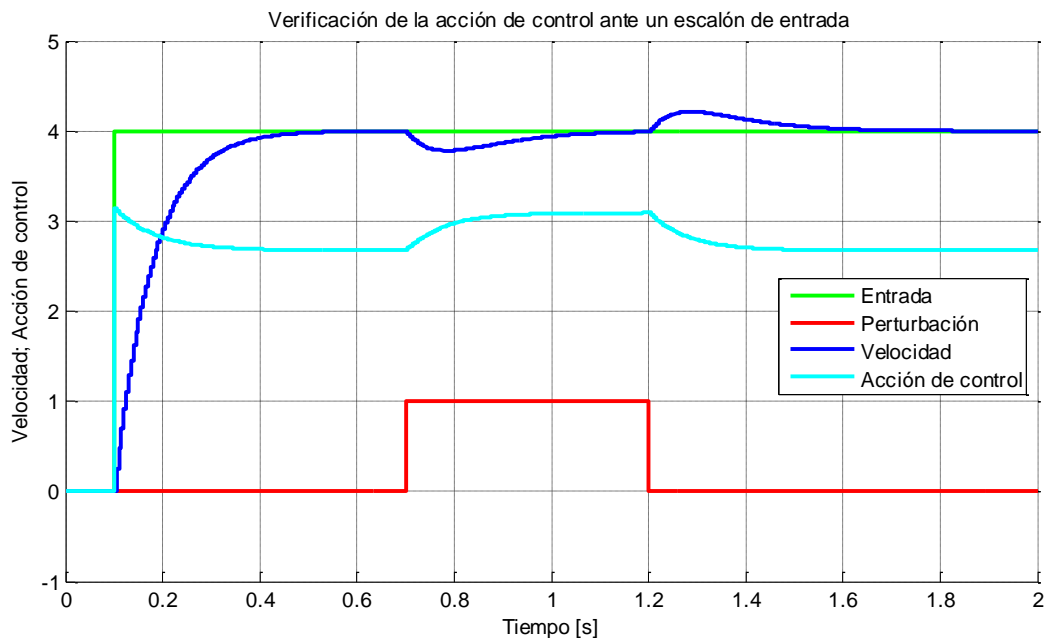


Figura 11: Simulación de: a) la respuesta con control PI(s), en *simulink*, ante una entrada escalón de 4krpm en 0.1s (la estabilización se leerá ahora en $0.1s + 0.3s = 0.4s$) y b) la respuesta ante perturbaciones en 0.7s y 1.2s.

Parte III: Implementación del diseño en el sistema PSoC

Paso 1: Alimente el kit PSoC 059

Conecte el kit059 al puerto USB de una PC. Determine con el programa "Administrador de dispositivos" de Windows, a cuál puerto COM se encuentra conectada la placa. Ese puerto COM debe configurarlo en la aplicación como se muestra en la Figura 16.

Paso 2: Construcción de la estructura de control

Abra la aplicación del Pascal y construya o cargue la estructura de control que se muestra en la Figura 12. La configuración del ADC_0 es iguales a la mostradas en la Figura 4. Las

configuraciones para los bloques SAT_0 y DAC_0 se muestran también respectivamente en la Figura 13 y en la Figura 14.

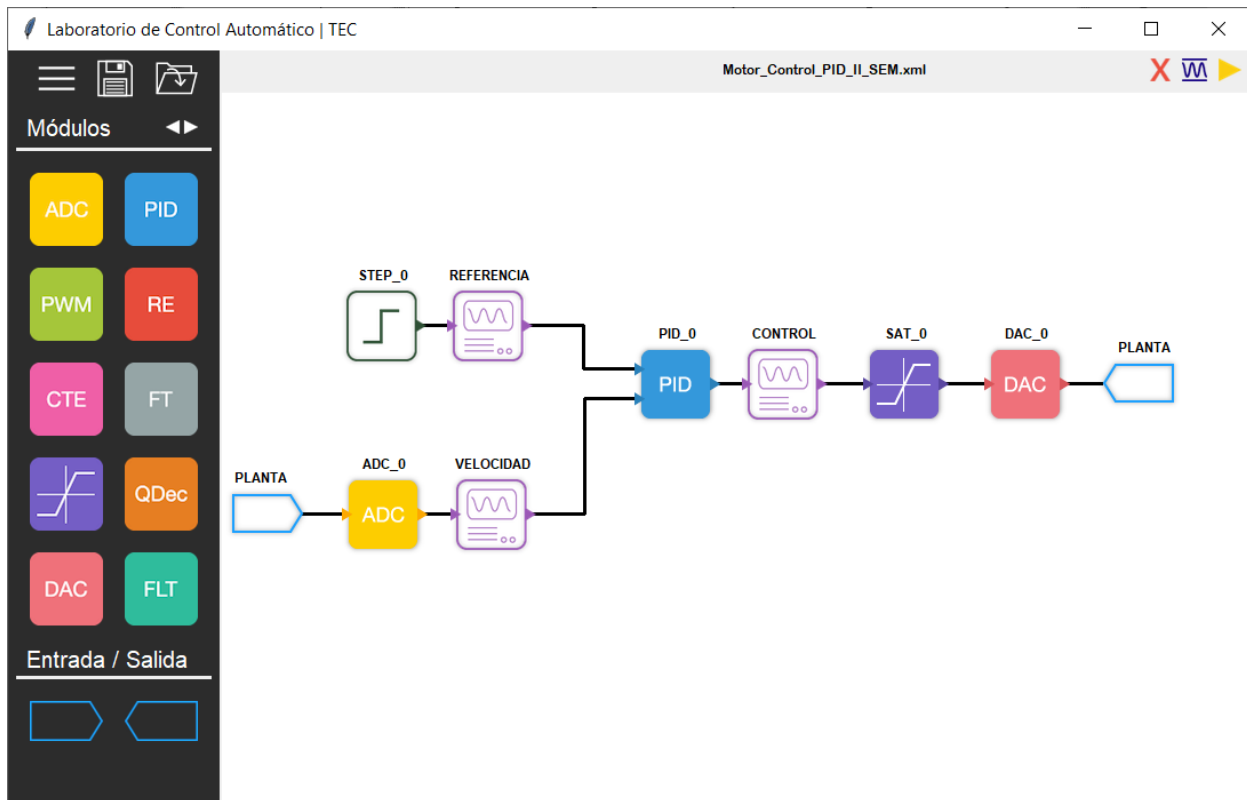


Figura 12: Estructura de control con PID (PI) para el motor de CD. El escalón (STEP_0) está configurado para iniciar en $t = 1$ s.

Figura 13: Configuración del saturador a 4.0V positivo y 0V negativo.

Figura 14: Configuración del DAC a +5V

Paso 3: Ajuste de las constantes K_P y K_I obtenidas en la etapa de diseño.

Abra el bloque PID_0 y coloque todos los ajustes mostrados en la Figura 15. En la sección resaltada en amarillo, coloque sus constantes P, I, calculadas durante la descomposición a PI paralelo y oprima “Aceptar” para guardar los cambios.

PID_0

Control

- ☒ PID
- ☐ I_PD
- ☐ PI_D

Parámetros

Proporcional (P): 1

Integral (I): 1

Derivativo (D): 0

Coeficiente Filtro (N): 1

Setpoint weight (b): 1

Setpoint weight (c): 1

Método Integrador: Backward Euler

Método Filtro: Forward Euler

Tiempo de muestreo (ms): 5

☒ Saturador

Límite Superior: 3.9

Límite Inferior: -3.9

$$P(b \cdot r - y) + I \cdot T_s \frac{z}{z-1} (r - y) + D \frac{N}{1 + N \cdot T_s \frac{1}{z-1}} (c \cdot r - y)$$

Cancelar Aceptar

Figura 15: Ajustes del bloque PID como PI

Configuración

Comunicación Serial

Puerto: COMX

Velocidad: 115200

Simulación

Tiempo de muestreo (ms): 20

Número de muestras: 1000

Cancelar Aceptar

Figura 16: Ajuste del puerto de comunicaciones y la velocidad de comunicación en bps

Paso 4: Ajuste de la comunicación en serie con el kit059

Abra la ventana de configuraciones (ícono de tres rayas horizontales en la parte superior izquierda de la Figura 12) y haga los ajustes de velocidad a 115200 bps y del puerto de comunicaciones al determinado en el paso 1.

Paso 5: Descarga de la estructura de control y configuración a la placa

Asegúrese de tener abierta y visible en segundo plano la ventana de comunicaciones, Figura 18, de la aplicación Pascal. Oprima en la aplicación Pascal, el botón de descargar (ícono de mensajes de correo o "TRANSMITIR") en la parte superior derecha de la Figura 12, espere unos segundos y al aparecer la ventana mostrada en la Figura 17, que indica el fin de la transmisión, verifique que en la ventana de comunicaciones, se muestran los datos transmitidos, como en la Figura 18.

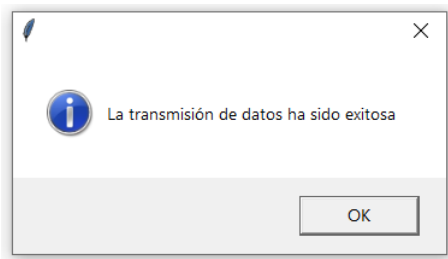


Figura 17: Ventana que indica el fin de la transmisión

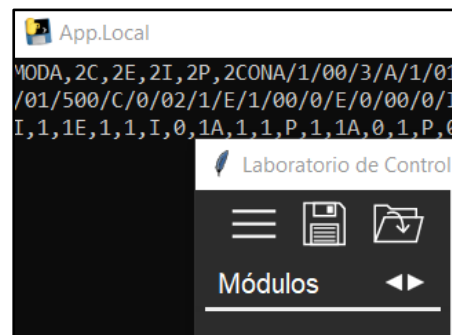


Figura 18: Ventana de comunicaciones que muestra que se envió exitosamente la configuración

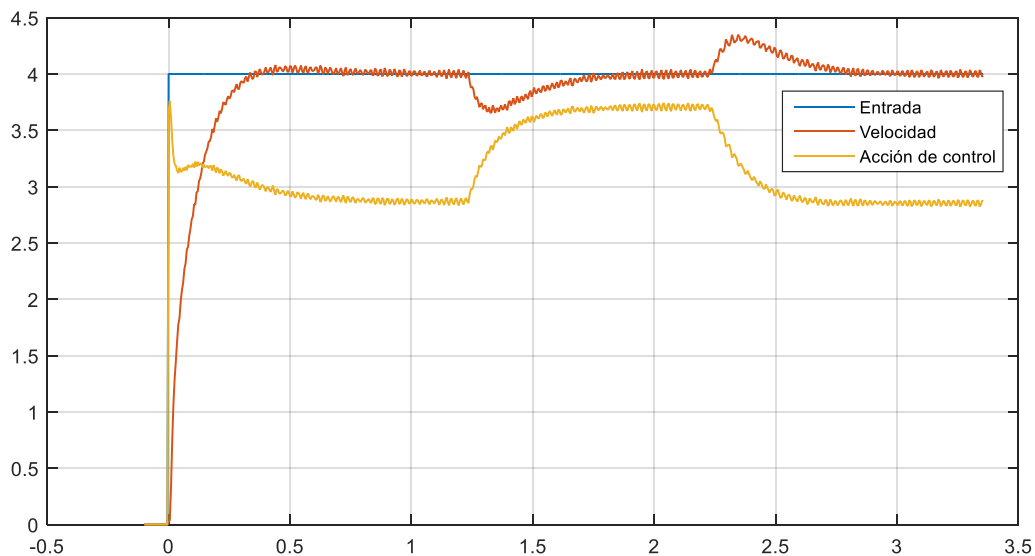


Figura 19: Captura de la respuesta del sistema hps5130 regulado con un PI, ante un escalón equivalente a 4000 rpm (4V) y ante perturbaciones. La acción de control está en voltios y la referencia y velocidad en krpm

Paso 6: Medición de la respuesta ante un escalón y ante perturbaciones del sistema regulado.

Abra la ventana del programa PUTTY. Cargue una sesión guardada o haga una nueva sesión y ajuste el puerto de comunicaciones serie y la velocidad a los mismos valores que en la aplicación Pascal. En la Figura 20 se muestra la configuración de la sesión. También verifique o haga los ajustes necesarios para que todos los datos de la sesión sean guardados en un archivo con la extensión CSV, digamos en el escritorio de la PC. En la Figura 21 se muestra la configuración de la captura en PUTTY.

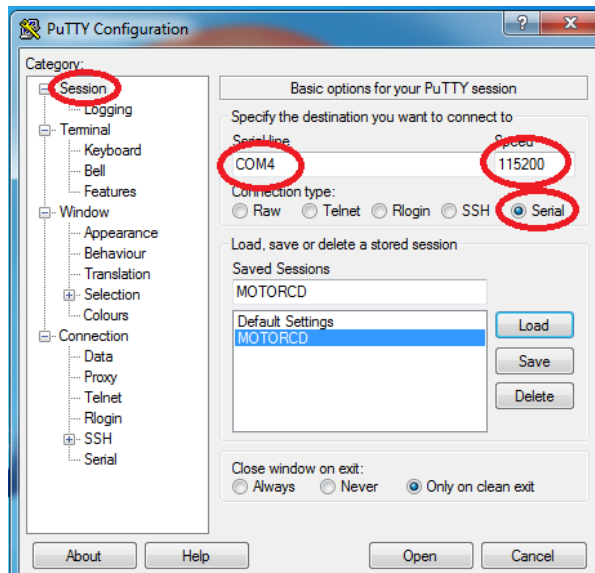


Figura 20: Configuración de la sesión en PUTTY.

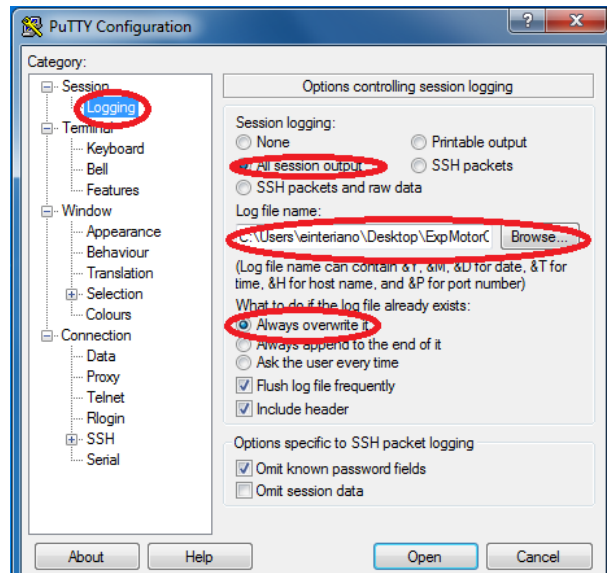


Figura 21: Configuración de la captura en PUTTY

Abra la sesión de PUTTY y utilizando el teclado, ejecute (**r** o **R**), aplique una perturbación (**e** o **E**), detenga la perturbación (**e** o **E** nuevamente) y termine la ejecución (**t** o **T**). Finalmente, cuando haya terminado el experimento cierre el programa PUTTY para que éste guarde los datos recopilados en el archivo y en la ubicación por Ud. seleccionada. Preserve este archivo de datos para el análisis posterior.

En la Figura 19 se puede apreciar la respuesta de velocidad del sistema motor CD, (naranja), ante una entrada interna tipo escalón en la referencia (azul) equivalente a 4000 rpm. También se muestra la acción de control (amarillo) y puede apreciarse como ésta cambia para corregir las perturbaciones. Al analizar la gráfica de velocidad se puede determinar que el tiempo de estabilización es aproximadamente 300ms; que el sobreimpulso es muy pequeño y que el error de estado estacionario es cero. Además, se puede ver que el regulador PI utilizado es capaz de cancelar las perturbaciones aplicadas al sistema y que lo hace en aproximadamente 300ms. Por lo que se puede dar por concluido el diseño e implementación.

Entrega: El paper debe subirse a más tardar una semana después de realizada la revisión del control al TecDigital en formato PDF, la nota para cada integrante del equipo será proporcional a la nota obtenida en la calificación de pares para todas las subnotas relacionadas a este proyecto corto.

EIS,ARB

2023