

Avance Trabajo Práctico Final

Control Digital - MSE

Alumno: Carlos Herrera Trujillo

I. Propuesta de Trabajo

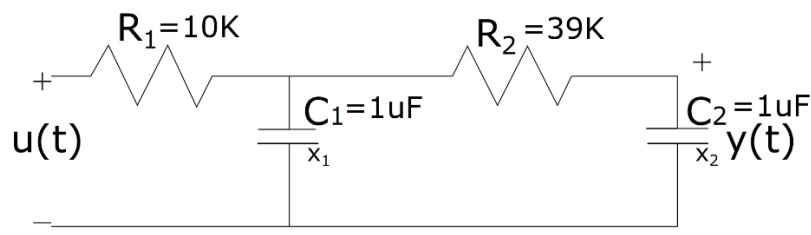


Figura 1: Circuito rc-rc

1. **Armado del circuito:** Para el armado del circuito los valores de resistencia y capacidad serán dados a cada alumno/a de manera particular.

En la Figura 2 se observa el diseño del circuito rc-rc en Altium haciendo uso de los componentes disponibles. Posteriormente se fabricó para su uso, obteniendo el circuito físico mostrado en la Figura 3.

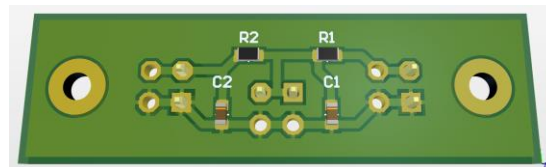


Figura 2: Diseño del circuito

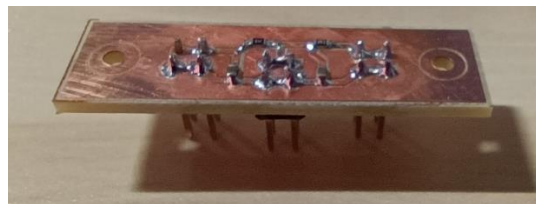
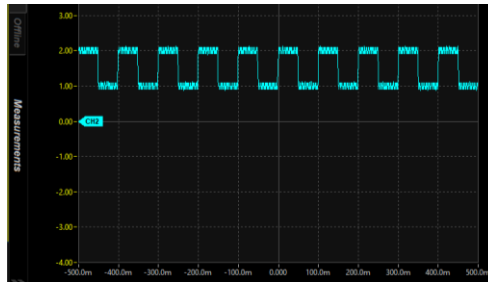


Figura 3: Circuito físico

2. **Análisis:** Generar una señal cuadrada que vaya de 1V a 2V con una frecuencia de 10Hz como señal de referencia (utilizar el DAC). **Sin aplicar control**, obtener la respuesta al escalón del sistema a lazo abierto. Medir el **tiempo de subida** (t_r) y mostrar los **gráficos** que se generan.

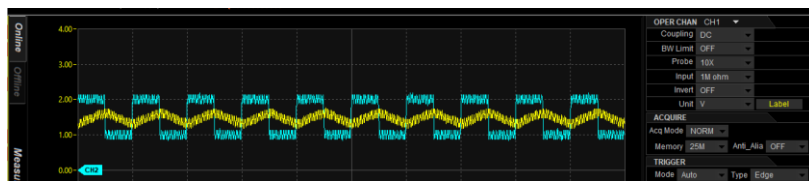
Archivos: TPF_p2_UDPreceiver.py, TPF_p2b_CSVsignalsplot.py



Señal de onda cuadrada 1V – 2V generado con DAC

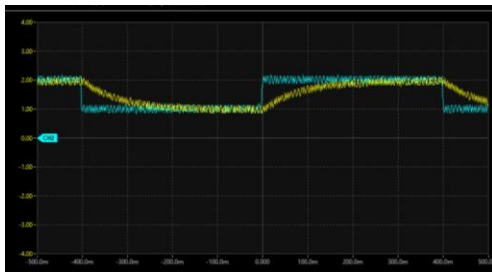
Frecuencia de oscilación 10 Hz (100ms)

A una frecuencia de 10 Hz se obtiene la siguiente gráfica, no obstante, se reducirá la frecuencia a fin de poder observar mejor el tiempo de subida de la señal resultante “y(t)” (señal amarilla).

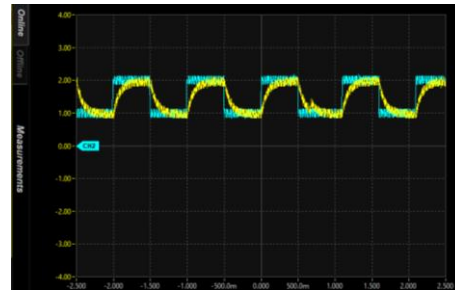


Resultados con frecuencia de onda cuadrada de 10 Hz

En las siguientes figuras se puede observar que tanto a una frecuencia de 1.25Hz y a una frecuencia de 1 Hz la señal obtenida es adecuada. Por tanto, para el presente trabajo se escogió la frecuencia de 1 Hz, por lo que la frecuencia de muestreo aplicado en el ADC debe ser mayor al doble de la frecuencia de la señal.

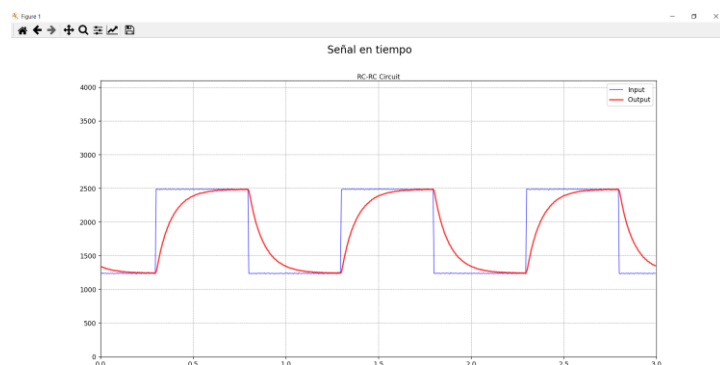


Frecuencia de oscilación 1.25 Hz (800ms)



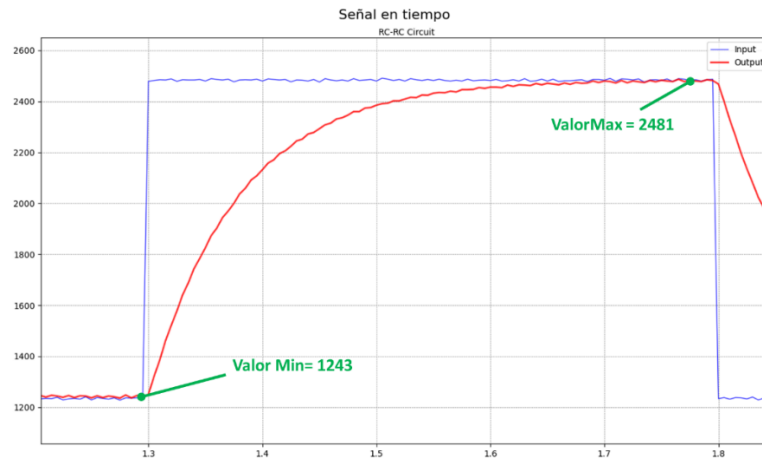
Frecuencia de oscilación 1 Hz (1000ms)

Para visualizar los datos tomados por el ADC, **en formato de entero de 16 bits**, se enviarán a través de ethernet por protocolo UDP y luego se realiza un gráfico en Python de los resultados.



Gráfica en Python

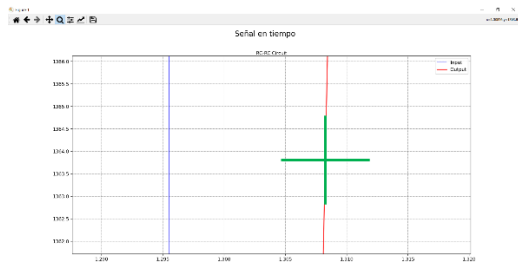
El tiempo de subida se calcula como el tiempo que tarda la señal en ir del 10 % al 90 % de su valor ($t_r = t_{90\%} - t_{10\%}$).



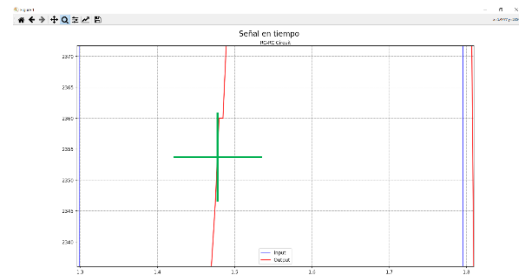
Gráfica en de la señal en valor mínimo y máximo

$$\begin{aligned} valor_{90\%} &= (2481 - 1243) \times 0.9 + 1243 \\ &= 1366.80 \end{aligned}$$

$$\begin{aligned} valor_{90\%} &= (2481 - 1243) \times 0.9 + 1243 \\ &= 2357.20 \end{aligned}$$



$$t_{10\%} = 1.31 \text{ seg}$$



$$t_{90\%} = 1.48 \text{ seg}$$

$$\begin{aligned} (t_r &= 1.48 - 1.31) \\ (t_r &= 0.17 \text{ segundos} \end{aligned}$$

3. Identificación: Determinar el valor de los polos del sistema mediante identificación, considerando al sistema como una caja gris.

$$G(s) = Y(s)/U(s)$$

Modelo matemático:

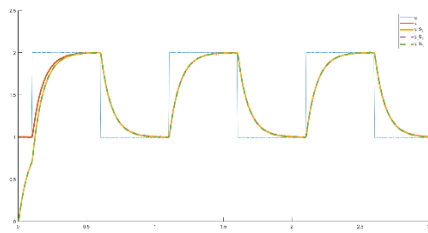
$$\frac{Y(s)}{U(s)} = \frac{1}{R_1 C_1 R_2 C_2 s^2 + [R_1 C_1 + R_1 C_2 + R_2 C_2]s + 1}$$

Se procedió a realizar la identificación con el algoritmo “offline” LS (“TPF_p2_Identificacion.m”) tanto para una señal de entrada cuadrada como para una señal aleatoria. No obstante, el error se mantenía alto tanto para el primer, segundo y tercer orden. Como paso a futuro se utilizará una mayor cantidad de muestras.

Los archivos rrcDatos0.csv a rrcDatos4.csv contienen capturas de las diversas señales generadas con los cuales se intentó obtener la identificación del sistema, no obstante, no se logró obtener un error mínimo.

Archivos: TPF_p3_Identificacion.m

Señal cuadrada

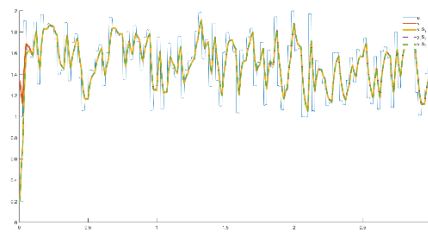


J_1 = 4.1858
J_2 = 4.1889
J_3 = 4.1895
numz_LS_1 = 0.0140 0.0464
denz_LS_1 = 1.0000 -0.9396

numz_LS_2 = 0.0152 0.0434 0.0456
denz_LS_2 = 1.0000 -0.2380 -0.6579

numz_LS_3 = 0.0154 0.0487 0.0547 0.0315
denz_LS_3 = 1.0000 0.1083 -0.5876 -0.3704

Señal Aleatoria



J_1 = 1.6021
J_2 = 1.6040
J_3 = 1.6034
numz_LS_1 = 0.1987 0.0339
denz_LS_1 = 1.0000 -0.7675

numz_LS_2 = 0.1983 0.0716 0.0022
denz_LS_2 = 1.0000 -0.5899 -0.1381

numz_LS_3 = 0.1989 0.0963 0.1368 0.0198
denz_LS_3 = 1.0000 -0.4612 0.4117 -0.4989

No obstante, el modelo matemático de la función de transferencia discreta de la planta calculada con un tiempo de muestreo de 0.005 segundos es:

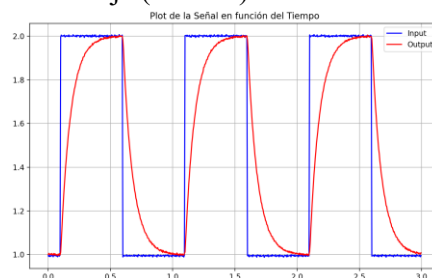
$$Hz = \frac{0.2517z + 0.01957}{z^2 - 1.425z + 0.4693}$$

4. **Controlador PID:** Diseñar y aplicar un control PID para cumplir con las siguientes especificaciones:

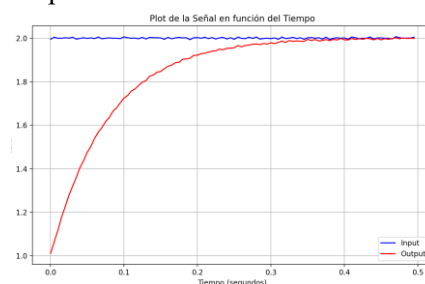
Archivos: TPF_p4_PID.py

- Tener error de estado estacionario nulo.
- Mejorar el tiempo de subida, t_r , un 30 % con respecto a la respuesta a lazo abierto.
- Tener un sobrepico menor al 8 %.

Se utiliza el método respuesta al escalón Ziegler – Nichols, así mismo, los valores en formato decimal del ADC se convierten a voltaje (3.3Vref).



Dado la señal obtenida del microcontrolador, con muestreo en la salida del DAC, se procede a utilizar la sección que sería la respuesta al escalón.

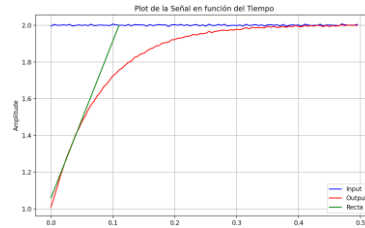


Se procede a dibujar la recta tangente con los valores de

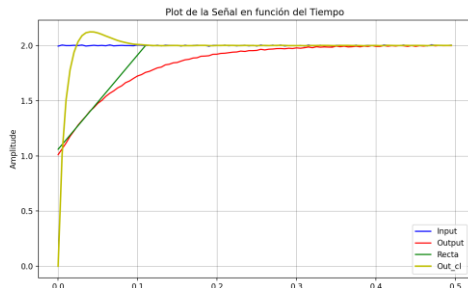
$$\begin{array}{ll} x1 = 0.015 & y1 = 1.188 \\ x2 = 0.11 & y2 = 2 \end{array}$$

Obteniendo de esta forma:

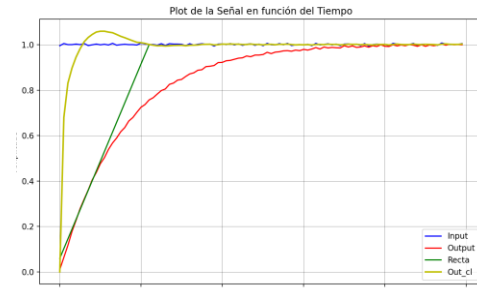
$$\begin{array}{ll} L = x1 & T = x2 - x1 \\ A = B = 2 \end{array}$$



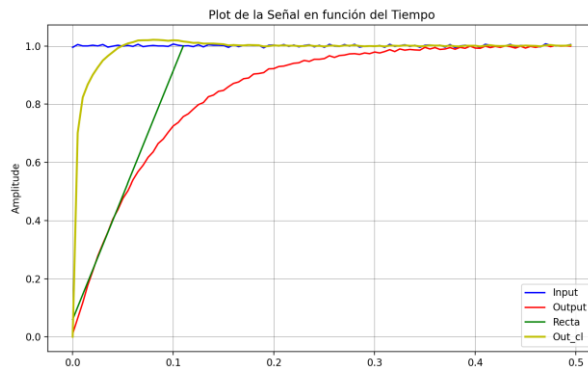
Una vez implementado el controlador PID se tiene el siguiente resultado



Se tiene un 6% con Kp, Ki y Kd sin alterar



Un sobre pico de 5.9% con 1.1Kp, 0.8Ki y 1.8 Kd



Con 1.1Kp, 1.4Ki y 2Kd se obtiene un sobrepico de **aproximadamente 2.13%**

Así mismo, se observa que el error de estado estacionario se elimina.

Y el tiempo actual de subida es de

$$t_{90\%} = 0.02 \text{ secs} \quad t_{10\%} = 0.0007 \quad \Rightarrow tr = 0.020 \text{ segundos}$$

Por lo tanto, dado que el tiempo de subida anterior era de 0.17 segundos, el nuevo tiempo de subida es de 0.02 segundos siendo una mejora de más del 30%.

5. **Período de Muestreo:** Seleccionar un tiempo de muestreo que cumpla con todas las restricciones prácticas vistas en clase.

A fin de poder seleccionar un buen periodo de muestreo, en el diseño del filtro PID se tuvo en cuenta el tiempo de subida de la señal no sea demasiado corto.

Frecuencia de muestreo mínima: Con un tiempo de subida de 20 milisegundos podemos tener como mínimo una frecuencia de muestreo de 200 Hz (20 milisegundos / 4 muestras)

Frecuencia de muestreo máxima: Para lo cual se debe tener en cuenta el tiempo que demora en muestrear, el tiempo de procesamiento y el tiempo del conversor digital/analógico. El periodo mínimo de muestreo será 20 veces la sumatoria de todos los tiempos mencionados.

En el microcontrolador STM32 F429 una vez obtenido la suma de todos los retardos provocados, estos se suman y el periodo mínimo de muestreo será 20 veces el valor de dicha sumatoria.

$$\begin{aligned}T(ADC) &= 84 \text{ ciclos de reloj} \\T(\text{Procesamiento}) &= \text{Medido en ciclos de reloj} \\T(DAC) &= \text{microsegundos}\end{aligned}$$

