

# Tratamiento y Transmisión de Señales

## Ingenieros Electrónicos

### EXAMEN CONVOCATORIA ORDINARIA JUNIO 2009

#### TERCERA PARTE: PRÁCTICAS

1 hora y media y 3 puntos.

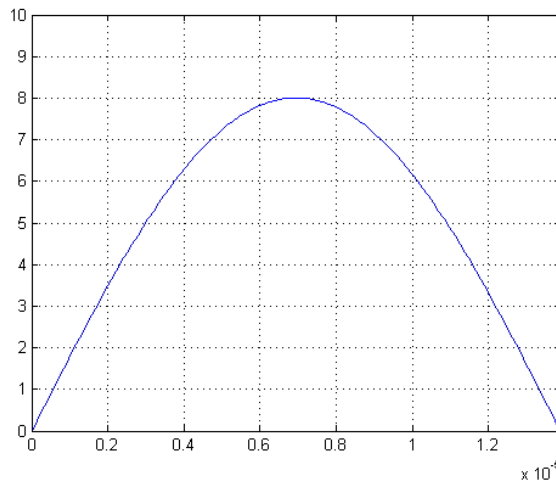
#### Instrucciones

Se deben realizar los pasos que se indican en la sección **Enunciado** partiendo de las prácticas realizadas durante la asignatura. El resultado del examen se debe entregar en formato electrónico utilizando Microsoft WORD respondiendo claramente a lo que se pide en el apartado **Resultados**. Cuando se pidan explicaciones de algún tipo sobre los resultados obtenidos, éstas se deben dar por escrito en el documento WORD haciendo referencia a la pregunta a la que se responde. Si se piden gráficas se deben copiar en el portapapeles desde Matlab y pegarlas en el documento WORD. Una vez terminado el documento (con las respuestas a las preguntas e incluyendo, en su caso, las gráficas correspondientes) se debe entregar al profesor. Incluir el código Matlab utilizado para la realización del examen.

#### Enunciado

Hacer un script de MATLAB que haga lo siguiente:

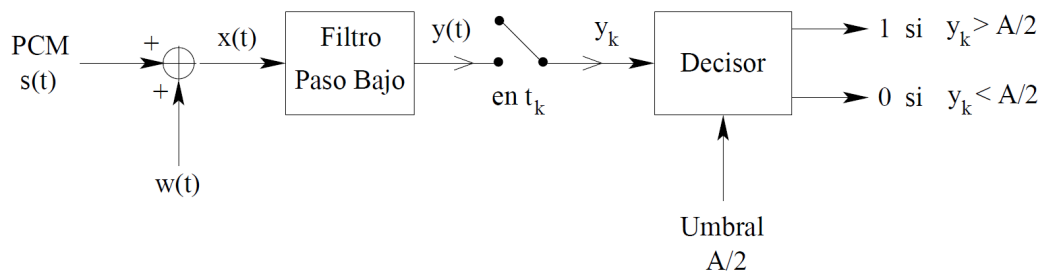
- Considerar  $N=10000$  bits aleatorios. Para ello definir el vector `bitsTx`.
- Considerar una tasa de transmisión  $R_b=72000$  bps. Determinar la duración de un bit  $T_b$ .
- Considerar que cada bit se va a representar en Matlab con  $K=100$  muestras. Determinar en este caso el periodo de muestreo  $T_s$  y la tasa de muestreo  $f_s$ .
- Definir un eje temporal  $t$  comenzando en  $t=0$  segundos y con la duración necesaria para definir la señal que contenga los  $N$  bits ya generados en `bitsTx`.
- Supongamos que la forma básica del bit 1 es con forma de medio seno, según se aprecia en la siguiente figura:



donde la amplitud máxima es  $A=8$ , y que el bit  $0$  corresponde a ausencia de pulso. Definir entonces la señal `s_PCM`.

- Determinar la autocorrelación  $R_s$  de la señal binaria `s_PCM` generada en el apartado anterior. Para ello utilizar el comando de Matlab `xcorr` usando la opción sin sesgo: `unbiased`.
- Determinar la densidad espectral de potencia  $S_s$  de la señal binaria `s_PCM`. Definir el eje de frecuencias  $f$  correspondiente a la densidad espectral calculada.
- Definir los coeficientes  $h$  de un filtro paso bajo con `orden=256` y frecuencia de corte  $B_T=500000$  Hz utilizando el comando de Matlab `fir1`. El parámetro  $B_T$  representa el ancho de banda de transmisión del sistema.

- Determinar la versión filtrada  $s_{\text{PCM}_f}$  de la señal binaria usando el filtro  $h$  definido en el apartado anterior y el comando de Matlab `filtfilt`.
- Determinar la nueva función de autocorrelación  $R_{s_f}$  y densidad espectral de potencia  $S_{s_f}$  para la señal filtrada  $s_{\text{PCM}_f}$ .
- Supongamos ahora que la señal  $s_{\text{PCM}}$  se transmite por un canal ruidoso (ruido Gaussiano, aditivo, blanco, con media cero y densidad espectral  $N_0/2$ ) cuyo nivel de ruido va cambiando. Generar un vector de  $M=10$  valores de densidad espectral de ruido  $N_0$  espaciados logarítmicamente entre los valores  $N_{0\text{min}}=5e-6$  W/Hz y  $N_{0\text{max}}=2e-4$  W/Hz, usando el comando de Matlab `logspace`.
- Hacer lo siguiente para cada nivel de ruido (dentro de un bucle `for`):
  - Determinar la varianza  $\text{var\_ruido}$  del ruido del canal para el nivel de ruido  $N_0/2$  correspondiente.
  - Generar un vector  $w$  de ruido blanco Gaussiano con media cero y varianza  $\text{var\_ruido}$  de la misma longitud que la señal binaria  $s_{\text{PCM}}$ . Determinar la versión ruidosa  $s_{\text{PCM\_ruido}}$  de la señal binaria.
  - Supongamos que usamos el siguiente esquema de receptor:



La primera etapa del receptor es un filtro paso bajo con ancho de banda  $B_T$ . Determinar la señal  $s_{\text{PCM\_ruido}_f}$  a la salida de dicho filtro. Igualmente determinar la versión filtrada  $n$  del ruido del canal  $w$ .

- Determinar los bits a la salida del decisor  $\text{bitsRx}$  suponiendo que el muestreador coja la muestra en la posición  $K/2$  para cada bit.
- Determinar la SNR de pico experimental  $\gamma$  a partir de la versión filtrada de la señal binaria  $s_{\text{PCM}_f}$  y de la versión filtrada del ruido  $n$ , ya calculadas.
- Determinar la probabilidad de error experimental  $P_e$ .

Al final del bucle se tiene que disponer de dos vectores:  $\gamma$  y  $P_e$  con  $M$  componentes cada uno, correspondientes a la SNR de pico y la probabilidad de error, ambas determinadas de forma experimental.

- Determinar la SNR de pico teórica  $\gamma_T$  y la probabilidad de error teórica  $P_{eT}$ , ambas con  $M$  componentes.

## Resultados

Se pide lo siguiente a incluir en el documento WORD:

- Incluir el código Matlab desarrollado comentando las partes más importantes del mismo.
- Una gráfica que incluya la señal binaria  $s_{\text{PCM}}$  (en color azul) y la señal binaria filtrada  $s_{\text{PCM}_f}$  (en color verde) como función del tiempo  $t$  (10 primeros bits adecuadamente escalados para su correcta visualización).
- Una gráfica que incluya la densidad espectral  $S_s$  (en color azul) y la densidad espectral  $S_{s_f}$  (en color verde) correspondientes a la señal binaria y su versión filtrada. Utilizar el comando `semilogy` para dibujar los niveles potenciales en unidades logarítmicas.
- Una gráfica que incluya la señal binaria  $s_{\text{PCM}}$  (en color azul) y la señal binaria ruidosa filtrada  $s_{\text{PCM\_ruido}_f}$  (en color verde) como función del tiempo  $t$  (10 primeros bits adecuadamente escalados para su correcta visualización). Hacerlo para el caso en el que el canal introduce la menor cantidad de ruido.
- Una gráfica en la que se represente la probabilidad de error teórica  $P_{eT}$  como función de la SNR de pico teórica  $\gamma_T$  (en color azul) y la probabilidad de error experimental  $P_e$  como función de la SNR de pico experimental  $\gamma$  (en color verde). Poner los valores de SNR en dB y utilizar escala logarítmica para representar probabilidades de error. Interpretar los resultados obtenidos.