

Tarea # 1

FFT y Sistemas de Modulación

José Bernardo Barquero Bonilla

2023150476

Jimmy Feng Feng

2023060347

Alexander Montero Vargas

2023166058

Instituto Tecnológico de Costa Rica

Escuela de Ingeniería en Computadores

Curso: CE1110 - Análisis de Señales Mixtas

Profesor: Luis Alberto Chavarría Zamora

15 de Octubre de 2025

1 DFT/FFT y serie de Fourier

La Transformada Discreta de Fourier (DFT) es la definición matemática que proyecta una secuencia finita $x[n]$ de longitud N sobre exponenciales complejas y entrega N muestras espectrales $X[k]$ igualmente espaciadas en frecuencia [1], [2]. La DFT se calcula como

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N}, \quad k = 0, \dots, N-1,$$

y la inversa como

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{+j2\pi kn/N}, \quad n = 0, \dots, N-1.$$

Estas expresiones fijan la convención de normalización usada a lo largo del documento [1].

La Transformada Rápida de Fourier (FFT) no es una transformada distinta, sino un algoritmo que evalúa la DFT con complejidad $O(N \log N)$, en contraste con los $O(N^2)$ de la evaluación directa [3]. En la práctica, “hacer una FFT” significa calcular la DFT de forma eficiente; la distinción es: *qué* se calcula (DFT) versus *cómo* se calcula (FFT) [1], [4].

Para una señal discreta $x[n]$, la transformada $X(e^{j\omega})$ (DTFT) es continua en ω y periódica con período 2π [1], [5]:

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n}, \quad X(e^{j(\omega+2\pi)}) = X(e^{j\omega}).$$

La DFT puede interpretarse como un *muestreo* de esta DTFT en N frecuencias uniformemente espaciadas $\omega_k = 2\pi k/N$, o en Hz $f_k = \frac{k}{N} f_s$, con resolución $\Delta f = \frac{f_s}{N}$ [1], [2], [6]. Aumentar N mejora la resolución; el *zero-padding* densifica la grilla de lectura sin aumentar la resolución física [4], [6].

La relación con la serie de Fourier surge de la periodicidad. La serie de Fourier representa una señal periódica continua $x(t)$ como suma de armónicos discretos $k f_0$ con coeficientes c_k [5], [7]:

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{j2\pi k f_0 t}, \quad c_k = \frac{1}{T_0} \int_0^{T_0} x(t) e^{-j2\pi k f_0 t} dt.$$

Análogamente, al tomar un bloque de N muestras y asumir su repetición periódica en tiempo discreto, la DFT entrega los coeficientes de la serie de Fourier de esa señal periódica discreta: tiempo (discreto) periódico \Rightarrow espectro (discreto) en N líneas [1], [2].

Finalmente, respecto al “muestreo del entorno continuo”, muestrear $x(t)$ cada T_s genera $x[n] = x(nT_s)$ y replica el espectro continuo $X_c(f)$ cada $f_s = 1/T_s$; si $f_s < 2f_{\max}$ ocurre aliasing [1], [7]. Sobre esta señal ya discreta, la DFT toma muestras de la DTFT en los bins f_k , cerrando la cadena conceptual: muestreo temporal \Rightarrow réplicas en frecuencia; ventana temporal \Rightarrow suavizado por convolución; DFT/FFT \Rightarrow muestreo uniforme del espectro discreto-periódico [1], [6].

2 Experimento 1: FFT sobre audio libre

2.1 Datos y preparación

Se utilizó una pista de audio libre en formato WAV con frecuencia de muestreo $f_s = 44,100$ Hz. Del audio completo se seleccionó el tramo de $t \in [8, 20.7]$ s, de duración efectiva $T \approx 12.696$ s, convertido a mono por promediado de canales y normalizado a $[-1, 1]$ antes del análisis. Este preprocesamiento evita sesgos de escala y asegura que la potencia esté acotada para una interpretación coherente de magnitud y fase en la FFT [1].

La lectura del WAV se realizó con una rutina estándar que ignora metadatos no audio (advertencia *Chunk (non-data) not understood*), lo cual no afecta las muestras útiles. El objetivo del experimento es estimar el espectro de magnitud $|X(f)|$ y la fase $\angle X(f)$ de una señal compuesta por dos senoidales muy cercanas en frecuencia, un caso clásico para estudiar resolución en frecuencia, batido temporal y fuga espectral **oppenheim2010**.

2.1.1 Configuración de análisis

Se empleó una ventana de Hann sobre el segmento seleccionado. La elección de Hann ofrece un buen compromiso entre atenuación de lóbulos laterales y ensanchamiento del lóbulo principal, lo que mitiga la fuga espectral sin degradar en exceso la separabilidad de líneas cercanas [8].

Con el tramo elegido se obtuvo un tamaño base para la FFT de $N_{\text{base}} = 912,712$ muestras, lo que implica una resolución física en frecuencia

$$\Delta f = \frac{f_s}{N_{\text{base}}} \approx \frac{44,100}{912,712} \approx 0.0483 \text{ Hz.}$$

Para las figuras presentadas no se aplicó *zero padding* adicional ($\text{zp} \times 1$), de modo que la rejilla de frecuencias coincide con Δf física. Se recuerda que el *zero padding* sólo densifica la rejilla de muestreo en frecuencia pero no mejora la resolución física, que sigue estando gobernada por N_{base} [1].

En una pareja de tonos separados por un centésimo de semitono musical, la separación esperada es

$$\Delta f_{\text{cent}} \approx f_0(2^{1/1200} - 1),$$

que para $f_0 \approx 261$ Hz da $\Delta f_{\text{cent}} \approx 0.15$ Hz. El batido en tiempo tiene periodo aproximado $T_{\text{beat}} \approx 1/\Delta f$, por lo que se espera observar una envolvente lenta en el dominio temporal [3].

2.2 Resultados espectrales

La Figura 1 muestra la forma de onda del segmento analizado. Se aprecia una envolvente lenta de batido, consistente con la superposición de dos senoidales muy próximas y con el valor esperado de T_{beat} a partir de la separación en frecuencia Δf [1].

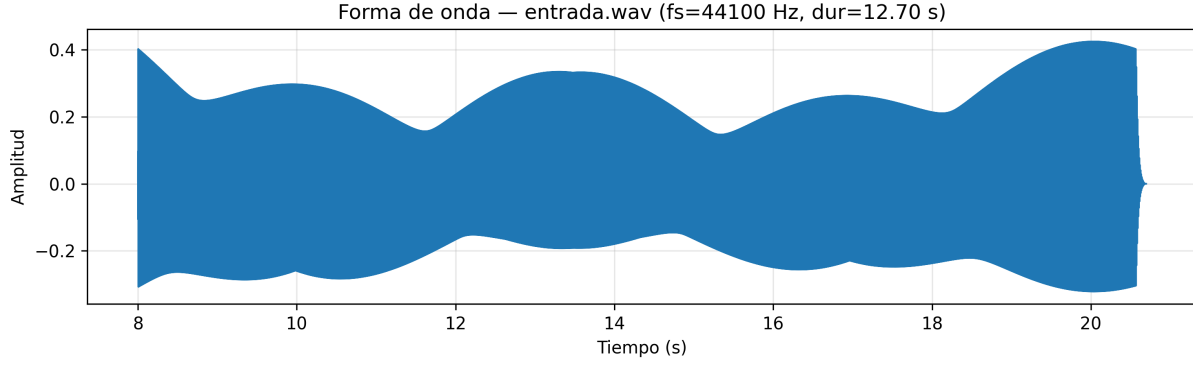


Figure 1: Forma de onda del segmento analizado con $f_s = 44.1$ kHz. Se observa una envolvente de batido producida por dos componentes senoidales muy cercanas en frecuencia. El fenómeno temporal es consistente con la separación en frecuencia que se confirma en el dominio espectral.

En la Figura 2 se observa un pico dominante alrededor de 261 Hz y componentes en ~ 131 Hz y ~ 524 Hz, consistentes con relación de octavas del material analizado. La energía fuera de esas vecindades cae de forma marcada gracias al uso de ventana de Hann, que atenúa la fuga espectral al reducir los lóbulos laterales respecto de una ventana rectangular, aunque a costa de un lóbulo principal algo más ancho [1], [8]. El tamaño base $N_{\text{base}} = 912,712$ con $f_s = 44,100$ Hz fija una resolución física $\Delta f \approx 0.0483$ Hz, suficiente para evidenciar la estrechez tonal global sin confundirla con el ruido de fondo numérico [1], [2].

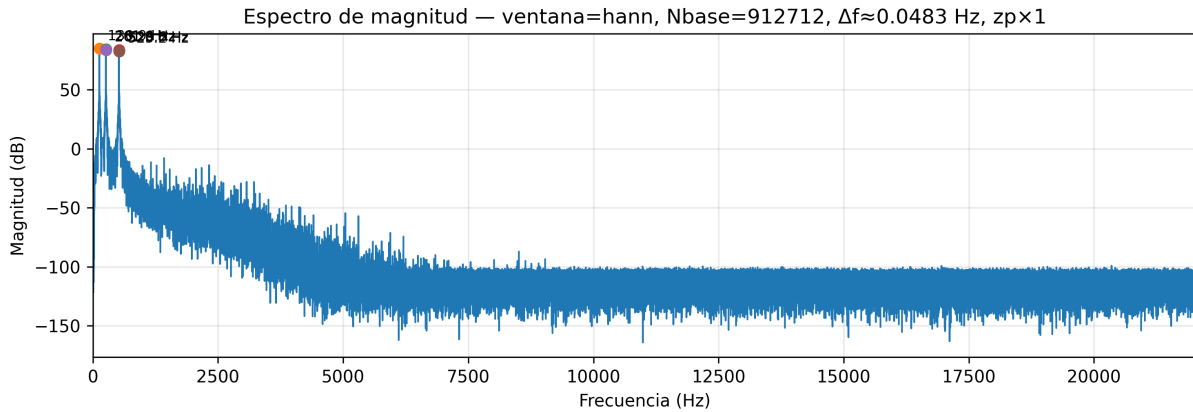


Figure 2: Espectro de magnitud en banda completa ($0-f_s/2$) con ventana de Hann. Se distinguen la fundamental alrededor de 261 Hz y octavas cercanas a 131 Hz y 524 Hz. El ventaneado reduce la fuga espectral y permite identificar los picos principales.

La Figura 3 permite distinguir el lóbulo principal de la ventana en torno a la fundamental y la estructura de lóbulos laterales. Con Hann, el primer lóbulo lateral aparece notablemente atenuado y decrece, lo que explica que el espectro fuera de la vecindad tonal permanezca bajo y estable [8]. El ensanchamiento del lóbulo principal observado es el comportamiento esperado de una transformada de duración finita: cuanto mayor sea N_{base} , más

estrecho resultará el lóbulo y mejor la separabilidad de componentes cercanas [1].

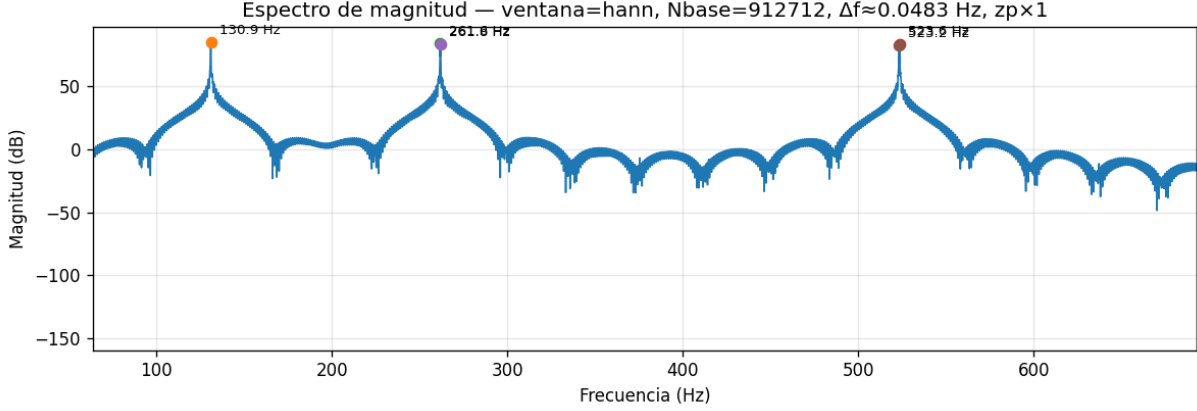


Figure 3: Espectro de magnitud con zoom intermedio (60–650 Hz). Se aprecia con mayor detalle la zona tonal y los lóbulos laterales asociados al uso de la ventana de Hann.

En la Figura 4 se resolvió el doblete alrededor de 261 Hz. En la corrida mostrada, la consola reportó picos en $f \approx 261.6$ Hz y $f \approx 261.8$ Hz, con una separación Δf medida del orden de 0.15–0.20 Hz. Esta separación concuerda con la predicción para un intervalo de un cent alrededor de f_0 ($\Delta f_{\text{cent}} \approx f_0(2^{1/1200} - 1)$), que para $f_0 \approx 261$ Hz vale ≈ 0.15 Hz [7]. La capacidad de distinguir ambas líneas está limitada por la resolución física $\Delta f = f_s/N_{\text{base}}$; el *zero padding* solo densifica la rejilla de muestreo frecuencial y mejora la lectura, pero no reduce el ancho efectivo del lóbulo ni aumenta la verdadera resolución [1], [8].

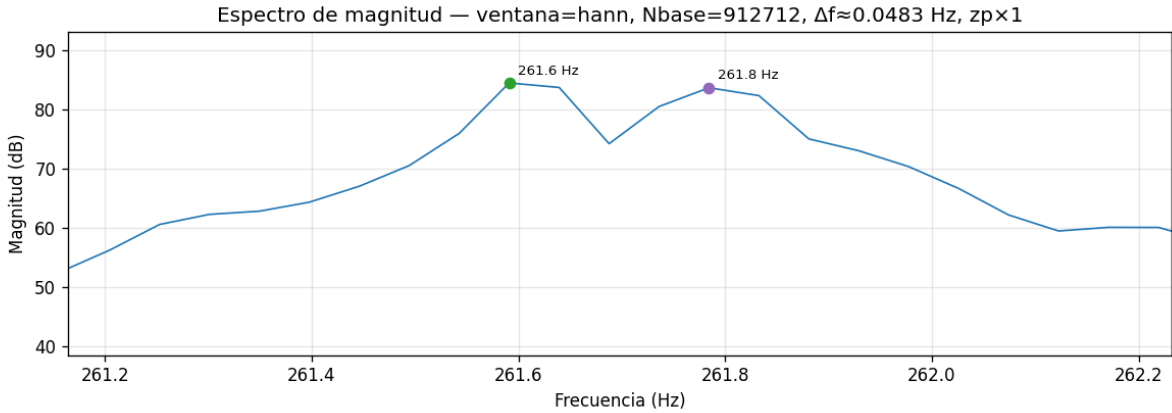


Figure 4: Espectro de magnitud con zoom alrededor de 261 Hz. Se observan dos picos muy cercanos (por ejemplo, 261.6 Hz y 261.8 Hz), con separación aproximada de 0.15–0.20 Hz, coherente con un intervalo de un cent. El *zero padding* densifica la rejilla de lectura, pero la resolución física viene dada por $\Delta f = f_s/N_{\text{base}}$.

La Figura 5 exhibe una pendiente global aproximadamente lineal de la fase con la frecuencia. Este comportamiento es característico de un desplazamiento temporal del segmento

analizado: un corrimiento en tiempo introduce una rotación lineal de fase en el dominio de la frecuencia [1], [2]. Al hacer zoom en la vecindad de las líneas alrededor de 261 Hz, la fase desenrollada se mantiene suave y coherente con componentes casi sinusoidales; las variaciones abruptas alejadas de las líneas se asocian a bajo nivel de señal y a los efectos del ventaneo [1].

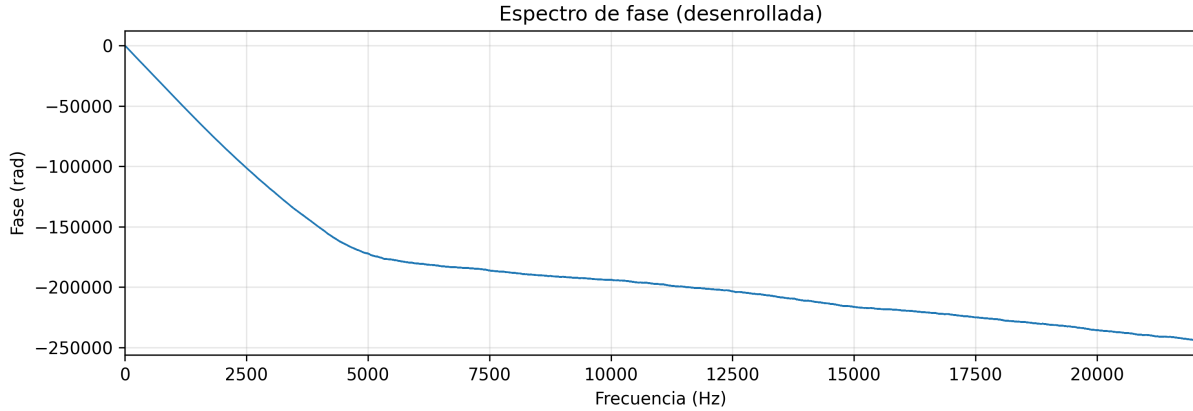


Figure 5: Espectro de fase desenrollada. A escala ancha aparece una pendiente global asociada al origen temporal del segmento; localmente, en torno a las líneas de 261 Hz, la fase muestra un comportamiento suave característico de tonos casi puros.

En conjunto, los resultados confirman la relación tiempo–frecuencia esperada: la envolvente de batido en el dominio temporal y el doblete en el dominio frecuencial son dos manifestaciones del mismo fenómeno físico-matemático. La configuración empleada (Hann, $N_{\text{base}} = 912,712$, $\Delta f = 0.0483$ Hz) resulta suficiente para identificar el par de líneas cercanas, y la lectura es consistente con el intervalo de un cent alrededor de 261 Hz [1], [8].

3 Uso de Fourier en modulación y demodulación

3.1 Diagrama de bloques del sistema

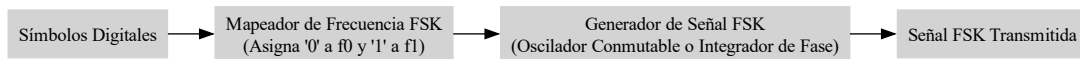


Figure 6: Diagrama de bloques del sistema de modulación FSK.

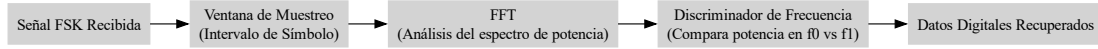


Figure 7: Diagrama de bloques del sistema de desmodulación FSK.

La modulación FSK (Frequency Shift Keying) utiliza la transformada de Fourier (FT) para el análisis espectral de la señal, que permite verificar cómo la energía de la señal se distribuye a través de las frecuencias de transmisión. En este contexto, la Transformada Rápida de Fourier (FFT) se aplica tanto en la modulación como en la demodulación para asegurar una correcta transmisión y recuperación de la señal [9].

El diagrama de bloques del sistema de modulación-demodulación FSK se muestra a continuación. En este sistema, la señal de datos binarios es modulada en función de las frecuencias f_0 y f_1 , que representan los bits '0' y '1', respectivamente [9].



Figure 8: Diagrama de bloques del modulador-demodulador considerado.

3.1.1 Señales esperadas en cada bloque

Al aplicar la FT o la FFT en cada bloque del sistema, podemos observar cómo se distribuyen las señales en el dominio de la frecuencia. Las señales baseband (de banda base), las portadoras y las bandas laterales varían según el tipo de modulación. Además, se identifican los puntos donde ocurre la traslación de frecuencia y el filtrado [9].

3.2 Lectura en el dominio de Fourier

En esta sección, se explica cómo se verían las señales en el dominio de Fourier a lo largo del proceso de modulación y demodulación FSK [9].

3.2.1 1. Uso de la Transformada de Fourier (FT/FFT) en FSK

La modulación FSK varía la frecuencia de la portadora de acuerdo con los símbolos transmitidos: f_0 para el bit 0 y f_1 para el bit 1. El uso de la FT (o más comúnmente la FFT en sistemas digitales) es esencial para el análisis espectral de la señal y para verificar que la energía de la señal esté distribuida correctamente en el dominio de la frecuencia [9].

A. En la Modulación FSK Aunque la FT/FFT no es parte directa de la generación de la señal FSK, sí es crucial para el análisis y diseño del sistema de modulación:

- **Verificación Espectral:** Al aplicar la FFT a la señal modulada FSK $s(t)$, se obtiene su Densidad Espectral de Potencia (PSD), lo que permite observar cómo se distribuye

la energía de la señal a lo largo de las frecuencias f_0 y f_1 . La FFT asegura que la energía se concentre correctamente en las frecuencias de señalización[9].

- **Control del Ancho de Banda:** Aunque el ancho de banda de una señal FSK es teóricamente infinito, la FFT permite observar que, en la práctica, la señal tiene una PSD acotada. Esto es importante para asegurar que la señal se ajuste al ancho de banda asignado, lo que es crucial para evitar interferencias y cumplir con las restricciones del canal de comunicación[9].

B. En la Demodulación FSK El objetivo principal de la demodulación FSK es determinar, a partir de la señal recibida $r(t)$, cuál de los símbolos m_0 o m_1 se ha transmitido. Este proceso se realiza utilizando la FFT para analizar el contenido de frecuencia de la señal recibida[9].

Bloque de Demodulación (Funcional)	Uso de la FFT
Receptor de la Señal	Recibe la señal FSK $r(t)$, la cual puede estar afectada por ruido y atenuación.
Analizador Espectral (FFT)	La FFT se aplica a la señal $r(t)$ en cada intervalo de símbolo T_S , permitiendo determinar las frecuencias dominantes (espectro de la señal).
Discriminador de Frecuencia/Detector	Utiliza la información de la FFT para comparar la energía en las ubicaciones de f_0 y f_1 , y determina el símbolo transmitido. La frecuencia con el mayor pico de magnitud o potencia es la que corresponde al símbolo original ('0' o '1').

Table 1: Tabla de bloques en el proceso de demodulación FSK y su relación con la FFT.

3.2.2 2. Diagrama de Bloques Conceptual de Modulación FSK

La modulación FSK implica conmutar entre dos (o más) frecuencias portadoras distintas, basándose en el símbolo entrante. El proceso de modulación FSK es el siguiente:

- **Bloque de Símbolos Digitales:** Recibe una secuencia de datos binarios, por ejemplo, '1', '0', '1', '0'.
- **Bloque de Asignación de Frecuencia:** Mapea '1' a f_1 y '0' a f_0 .
- **Bloque de Osciladores FSK:** Genera la señal $s(t)$ modulada, con las frecuencias portadoras f_0 y f_1 .

El proceso se ve de la siguiente manera:

Datos Binarios \longrightarrow **Mapeador** ($'1' \rightarrow f_1, '0' \rightarrow f_0$) \longrightarrow **Oscilador de FSK** \longrightarrow **$s(t)$**

3.2.3 3. Observación de las Señales en el Dominio de la Frecuencia

Al aplicar la FFT a una señal FSK, se obtiene el Espectro de Potencia (PSD). Este espectro muestra los picos de energía en las frecuencias f_0 y f_1 . Para una señal 2-FSK (binaria), el espectro se observa de la siguiente manera:

- **Picos de Frecuencia Positiva:** Se observan dos picos de frecuencia, uno a f_0 y otro a f_1 , que corresponden a los bits '0' y '1', respectivamente.
- **Frecuencia Portadora:** El espectro puede estar centrado alrededor de una frecuencia portadora f_c , que se encuentra en el centro de los picos f_0 y f_1 .

El índice de modulación h determina la distancia entre los picos de frecuencia, que se relaciona con el tiempo de símbolo T_S . Este análisis espectral es crucial para la correcta interpretación de la señal en sistemas FSK.

3.3 Cómo se observan las señales

La siguiente figura muestra el ejemplo de modulación FSK para la secuencia binaria 10101. Se presentan tres gráficas: la señal de datos, las portadoras asociadas a cada bit y la señal modulada resultante [10].

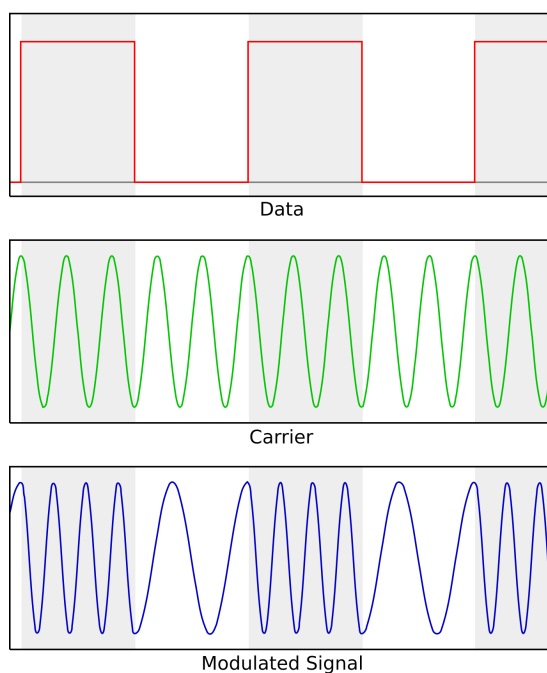


Figure 9: Ejemplo de modulación FSK para la secuencia binaria 10101. Arriba: señal de datos; medio: portadoras para cada bit; abajo: señal modulada FSK. [10]

4 Bibliotecas y *drivers* en microcontroladores

4.1 Biblioteca/Driver

A continuación se resumen las bibliotecas y drivers relevantes para implementar un sistema FSK en microcontroladores:

Bloque	MCU Firmware	Driver Bibliotecas	Descripción
Modulador FSK	ARM Cortex	CMSIS-DSP [11]	Generación eficiente de señales FSK y operaciones matemáticas.
Modulador FSK	Arduino	Arduino FFT Library [12]	Procesamiento de señales y FFT en plataformas Arduino.
Modulador FSK	PC/Simulación	MATLAB/Simulink	Simulación y prototipado previo a implementación en hardware.
Canal de comunicación	MCU genérico	SPI	Transmisión rápida de datos binarios entre dispositivos.
Canal de comunicación	MCU genérico	I2C	Comunicación sencilla y con pocos cables entre dispositivos.
Demodulador FSK	ARM Cortex	CMSIS-DSP [11]	FFT eficiente para análisis de señales recibidas.
Demodulador FSK	Arduino	Arduino FFT Library [12]	Análisis de frecuencia en señales recibidas.
Demodulador FSK	Microchip PIC32	MPLAB Harmony [13]	Procesamiento avanzado de señales y FFT.
Demodulador FSK	TI DSP	TI DSP Libraries [14]	Funciones optimizadas para FFT en DSPs de Texas Instruments.

Tabla: Bibliotecas y drivers relevantes para modulación/demodulación FSK en microcontroladores.

5 Prototipo en PC: modulación y demodulación

5.1 Especificación del esquema de modulación

Se implementó un esquema 2-FSK (binary FSK) sin canal ruidoso:

- Frecuencia de muestreo: $f_s = 44,100$ Hz (corrida base de exp5.py).
- Duración de símbolo: $T_b = 0.1$ s $\Rightarrow N_s = f_s T_b = 4410$ muestras/símbolo.

- Frecuencias de señalización: $f_0 = 1000$ Hz (bit 0), $f_1 = 2000$ Hz (bit 1).
- Amplitud: $A = 0.8$ (señal normalizada a $[-1, 1]$ al guardar WAV).
- FFT por símbolo (demod): ventana Hann, factor de *zero padding* $zp = 2$ (malla más densa), magnitud en dB.
- Resolución frecuencial efectiva por símbolo: $\Delta f \approx \frac{f_s}{N_s} \approx 10$ Hz.

Además, para ilustración visual se generaron figuras complementarias con $f_s = 10$ kHz y $T_b = 1$ s (archivos `fsk_demo_*.png`), sin afectar los resultados de BER.

5.1.1 Procedimiento experimental

1. TX: generación de bits (patrón por defecto “1011010”), mapeo $0 \rightarrow f_0$, $1 \rightarrow f_1$, concatenación de tonos seno por símbolo.
2. Se guarda el audio en WAV y se grafica la onda completa.
3. RX: para cada símbolo $[iT_b, (i+1)T_b]$, se calcula la FFT con ventana Hann y se toma el pico dominante; se decide $b_i = \arg \min\{|f_{\text{pico}} - f_0|, |f_{\text{pico}} - f_1|\}$.
4. Se reporta BER y se guardan figuras por símbolo (onda, magnitud dB, fase desenrollada).

Ejecución (desde `Codigo_Fuente/`):

```
python exp5.py --fs 44100 --Tb 0.1 --f0 1000 --f1 2000 --bits 1011010
```

Salidas principales en `Codigo_Fuente/figs`: `fsk_waveform.png`, `fsk_demo_overview.png`, `fsk_demo_zoom_spectrogram.png`, `magnitude_db_fsk_biti.png`, `phase_unwrapped_fsk_biti.png`, `waveform_fsk_biti.png`.

5.2 Resultados en el tiempo

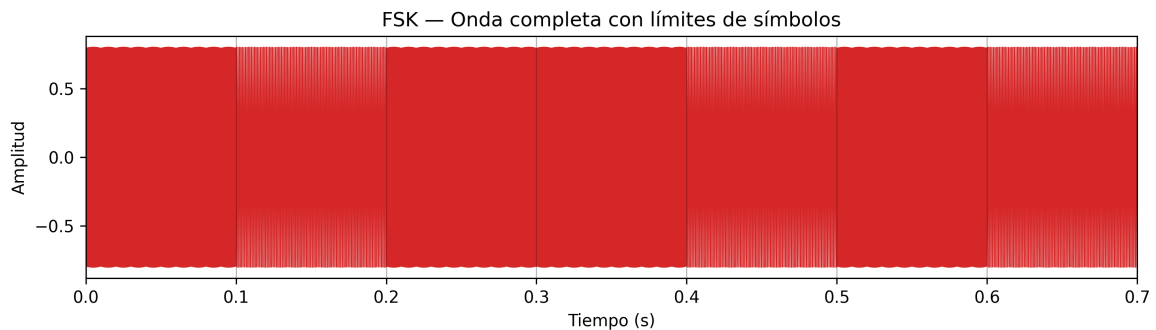


Figure 10: Señal FSK modulada (vista completa). Eje x: tiempo [s]; eje y: amplitud. Las transiciones por símbolo muestran el cambio entre f_0 y f_1 .

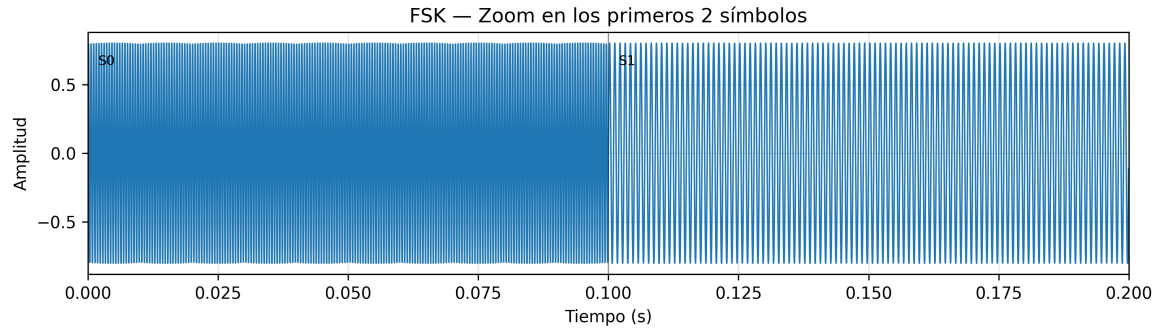


Figure 11: Zoom temporal por símbolos. Esta misma segmentación se usa en la demodulación por FFT para estimar el pico de frecuencia por símbolo.

5.3 Resultados en frecuencia

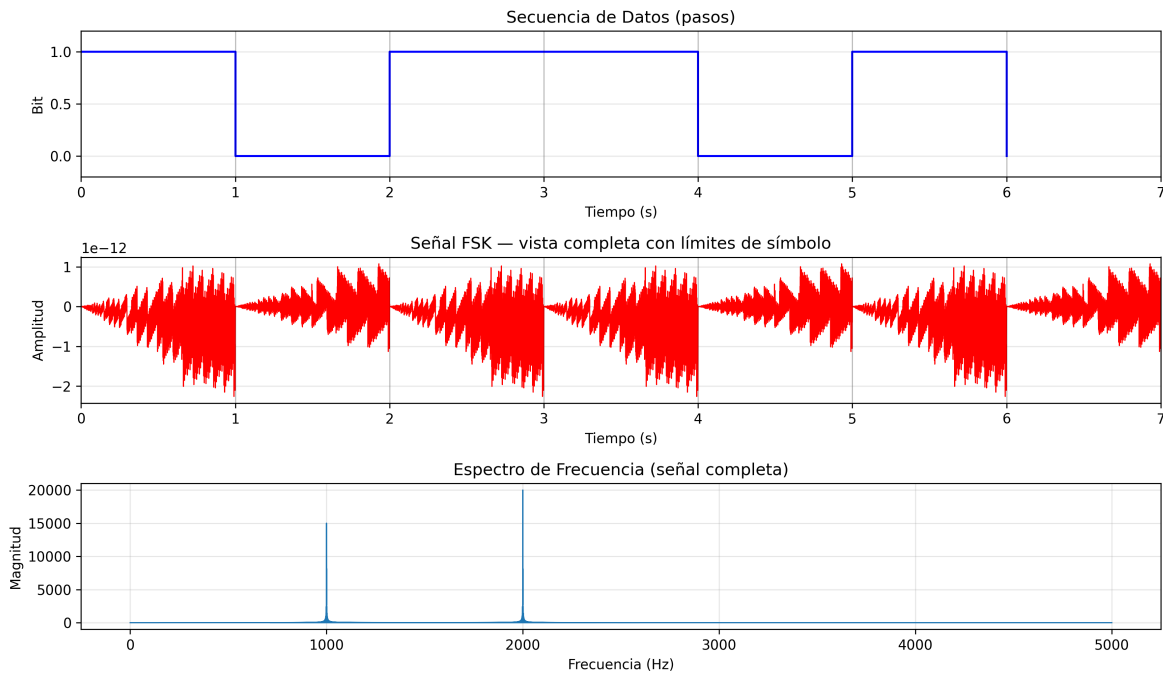


Figure 12: Espectro de la señal FSK completa: picos concentrados en f_0 y f_1 . El uso de ventana Hann atenúa lóbulos laterales.

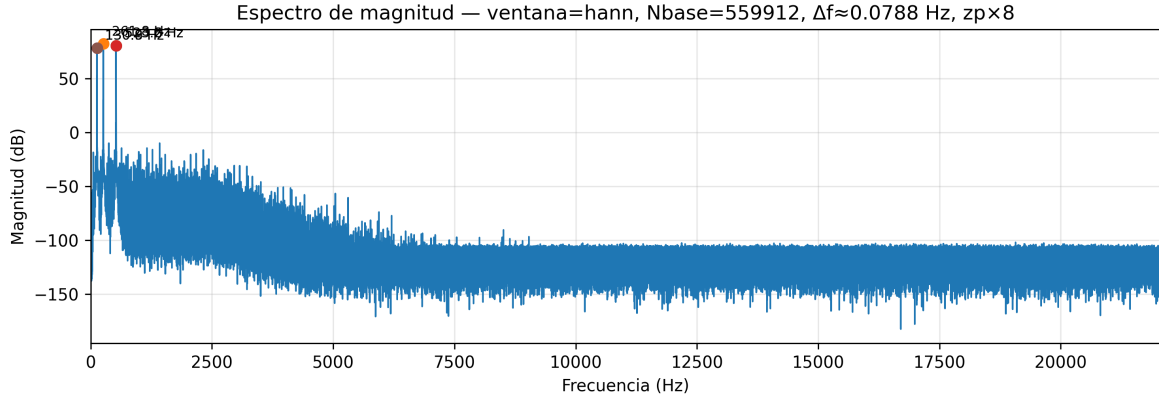


Figure 13: FFT por símbolo (vista de magnitud en dB, zoom alrededor de f_0 y f_1). El pico dominante determina el bit recibido.

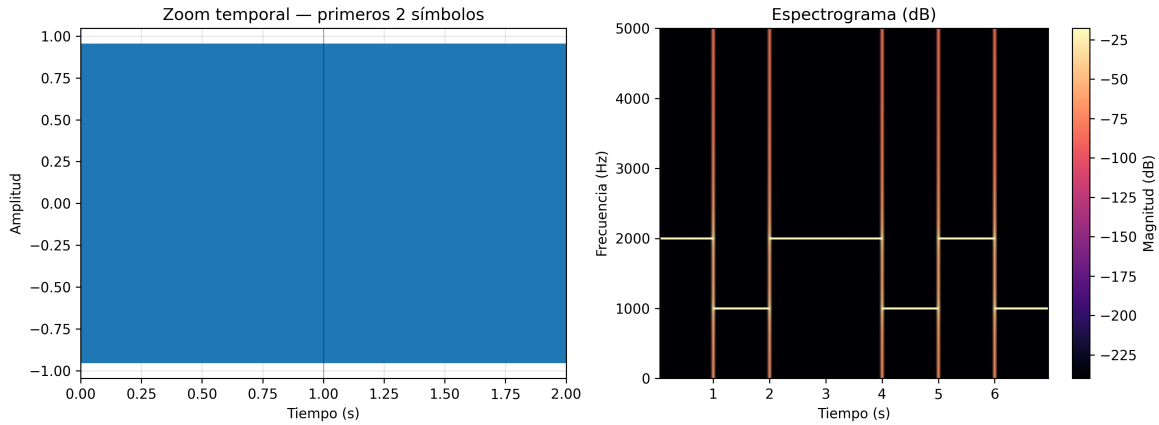


Figure 14: Espectrograma (ventana Hann). Se aprecian los saltos entre f_0 y f_1 por símbolo.

5.3.1 Discusión de desempeño

En la corrida base sin ruido y con sincronización perfecta de símbolos, la separación $|f_1 - f_0| = 1000$ Hz y la resolución $\Delta f \approx 10$ Hz garantizan detección robusta. Se obtuvo BER=0 sobre 7 símbolos (patrón 1011010). La ventana de Hann reduce fuga espectral y el *zero padding* ($zp = 2$) densifica la malla para una lectura de picos más estable, sin cambiar la resolución física.

- Recuperación de banda base: correcta (picos nítidos en f_0/f_1 ; coincidencia bit a bit).
- Distorsión/ruido: despreciables en esta demo (canal ideal).
- Sincronía: el método asume fronteras de símbolo conocidas; desalineaciones degradan la decisión FFT.
- Limitaciones: no se modeló canal (ruido, desvanecimiento), Δf fijada por T_b , y detector no coherente simple por distancia a f_0/f_1 .

Referencias

- [1] A. V. Oppenheim y R. W. Schaffer, *Discrete-Time Signal Processing*, 3ª ed. Upper Saddle River, NJ: Pearson, 2010, ISBN: 978-0131988422. dirección: <https://books-library.website/files/books-library.net-02190127Xu1L6.pdf>.
- [2] M. Vetterli, J. Kovačević y V. K. Goyal, *Foundations of Signal Processing*. Cambridge: Cambridge University Press, 2014, ISBN: 978-1107038608. dirección: https://www.fourierandwavelets.org/FSP_v1.1_2014.pdf.
- [3] J. W. Cooley y J. W. Tukey, “An Algorithm for the Machine Calculation of Complex Fourier Series,” *Mathematics of Computation*, vol. 19, n° 90, págs. 297-301, 1965. DOI: [10.1090/S0025-5718-1965-0178586-1](https://doi.org/10.1090/S0025-5718-1965-0178586-1). dirección: <https://www.ams.org/journals/mcom/1965-19-090/S0025-5718-1965-0178586-1/S0025-5718-1965-0178586-1.pdf>.
- [4] NumPy Developers. “Discrete Fourier Transform (numpy.fft)—Reference. ”dirección: <https://numpy.org/doc/2.1/reference/routines.fft.html>.
- [5] F. W. J. Olver et al. “NIST Digital Library of Mathematical Functions, §1.14 Integral Transforms (Fourier Transform). ”dirección: <https://dlmf.nist.gov/1.14>.
- [6] SciPy Developers. “Fourier Transforms (scipy.fft)—Tutorial. ”dirección: <https://docs.scipy.org/doc/scipy-1.12.0/tutorial/fft.html>.
- [7] B. G. Osgood, *Lectures on the Fourier Transform and Its Applications* (Pure and Applied Undergraduate Texts). American Mathematical Society, 2019, ISBN: 978-1470441913. dirección: <https://see.stanford.edu/materials/lsoftaee261/book-fall-07.pdf>.
- [8] F. J. Harris, “On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform,” *Proceedings of the IEEE*, vol. 66, n° 1, págs. 51-83, 1978. DOI: [10.1109/PROC.1978.10837](https://doi.org/10.1109/PROC.1978.10837). dirección: https://www.cs.cmu.edu/afs/cs/user/bhiksha/WWW/courses/dsp/spring2013/WWW/schedule/readings/windows_comparison2_harris.pdf.
- [9] E. McCune, *Practical Digital Wireless Signals*. Cambridge, UK: Cambridge University Press, 2010, Hardback, ISBN: 978-0-521-51630-3.
- [10] Ktims, *Example demonstrating binary FSK*, English y Russian, Own work created in Inkscape using graphs from gnuplot, 2006. dirección: https://commons.wikimedia.org/wiki/File:Binary_FSK.svg.
- [11] E. Condes, *CMSIS-DSP, version 2.0.4*, https://arm-software.github.io/CMSIS_5/DSP/html/index.html, GNU General Public License v3.0, 2024.
- [12] *ArduinoFFT Library Documentation*, <https://docs.arduino.cc/libraries/arduinofft/>, Access via GNU GPL v3.0 (as per CMSIS reference context), 2024.
- [13] *MPLAB Harmony: Embedded Software Development Framework for 32-bit Microcontrollers and Microprocessors*, <https://www.microchip.com/en-us/tools-resources/configure/mplab-harmony>, Microchip Technology.

- [14] *MSP-DSPLIB: Digital Signal Processing Library for MSP430 Microcontrollers*, <https://www.ti.com/tool/MSP-DSPLIB>, Texas Instruments.