UNIVERSIDAD DE SANTIAGO DE CHILE
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA INFORMÁTICA

Control 4

Integrantes	Franco Tapia
RUT:	19.234.165-5
Curso:	Redes de computadores
Profesor:	Carlos González
Ayudante:	Fernanda Muñoz

8 de Agosto de 2020

Tabla de contenidos

1. C4-P1: Canal 1

2. C4-P2: Modulación AM 4

1. C4-P1: Canal

1. Considere el siguiente canal binario donde $P(X1) = P(X2) = 1/2$

   a) Calcule la matriz de canal

   Solución:

   Para calcular la matriz de canal, se debe usar la siguiente ecuación:

$$Q_0 = \begin{bmatrix} P(y_1/x_1) & P(y_1/x_2) \\ P(y_2/x_1) & P(y_2/x_2) \\ P(y_3/x_1) & P(y_3/x_2) \end{bmatrix} \tag{1}$$

   En este caso, se debe considerar que existen dos probabilidades por calcular.

   $P(y_3/x_1) = \frac{4}{8}$ y $P(y_1/x_2) = \frac{2}{8}$

   Remplazando en la ecuación (1) se obtiene la Matriz de Canal solicitada en el enunciado.:

   Resultado

$$Q_1 = \begin{bmatrix} \frac{3}{8} & \frac{2}{8} \\ \frac{1}{8} & \frac{3}{8} \\ \frac{4}{8} & \frac{3}{8} \end{bmatrix}$$

   b) Calcule la información del canal

   Para el calculo de la información de canal, se utiliza la siguiente ecuación:

$$I = H(y) - H(y/x) \tag{2}$$

   Donde:

   $H(y)$ la entropía de y, que se calcula así:

$$H(y) = P(y_0) \cdot log_2(\frac{1}{P(y_0)}) + P(y_1) \cdot log_2(\frac{1}{P(y_1)}) + ... \tag{3}$$

Por otro lado, $H(y/x)$ la entropía de y respecto a x, la cual se obtiene de la siguiente manera:

$$H(\frac{y}{x}) = \sum_{i=1}^{n} \sum_{j=1}^{s} P(x_i) \cdot P(y_j/x_i) \cdot log_2(\frac{1}{P(y_j/x_i)}) \tag{4}$$

Desarrollo:

Para obtener el resultado de (3) se realiza lo siguiente:

- Obtención de $P(y_1)$:

$$P(y_1) = P(x_1) \cdot P(\frac{y_1}{x_1}) + P(x_2) \cdot P(\frac{y_1}{x_2})$$

$$P(y_1) = \frac{1}{2} \cdot \frac{3}{8} + \frac{1}{2} \cdot \frac{2}{8}$$

$$p(y_1) = \frac{5}{16}$$

- Obtención de $P(y_2)$:

$$P(y_2) = P(x_1) \cdot P(\frac{y_2}{x_1}) + P(x_2) \cdot P(\frac{y_2}{x_2})$$

$$P(y_2) = \frac{1}{2} \cdot \frac{1}{8} + \frac{1}{2} \cdot \frac{3}{8}$$

$$p(y_2) = \frac{1}{4}$$

- Obtención de $P(y_3)$:

$$P(y_3) = P(x_1) \cdot P(\frac{y_3}{x_1}) + P(x_2) \cdot P(\frac{y_3}{x_2})$$

$$P(y_3) = \frac{1}{2} \cdot \frac{4}{8} + \frac{1}{2} \cdot \frac{3}{8}$$

$$p(y_3) = \frac{7}{16}$$

- Calculando la Entropía (3)

$$H(y) = \frac{5}{16} \cdot log_2(\frac{16}{5}) + \frac{1}{4} \cdot log_2(4) + \frac{7}{16} \cdot log_2(\frac{16}{7})$$

$$H(y) = 1{,}546$$

Respecto al calculo de (4), se realiza el siguiente procedimiento:

- Calculo de $H(\frac{y}{x})$
  $H(\frac{y}{x}) = \frac{1}{2}\left[\frac{3}{8} \cdot log_2(\frac{8}{3}) + \frac{1}{8} \cdot log_2(8) + \frac{4}{8} \cdot log_2(\frac{8}{4}) + \frac{2}{8} \cdot log_2(\frac{8}{2}) + \frac{3}{8} \cdot log_2(\frac{8}{3}) + \frac{3}{8} \cdot log_2(\frac{8}{3})\right]$

$$H(\frac{y}{x}) = 1{,}483$$

Finalmente, la información obtenida según la ecuación (2) es:

$$I = 1{,}546 - 1{,}483$$

Resultado

$$I = 0{,}063$$

Es decir, se ha obtenido muy poca información del canal binario.

2. C4-P2: Modulación AM

1. Implemente un programa en Python que module en AM la señal de la figura. Use una portadora de 100 Hz.

   a) Grafique en el tiempo y la frecuencia la señal, la portadora y la señal modulada.

   Para gráficar la función se elaboró un programa en Python, con el orden presentado a continuación.

   - Se procede a crear la señal original, definiendo una función llamada signal.

```
def signal(t):
    return cos(2*pi*3*t)+sin(2*pi*2*t)
```

   - Gráfico de la señal original en el tiempo:

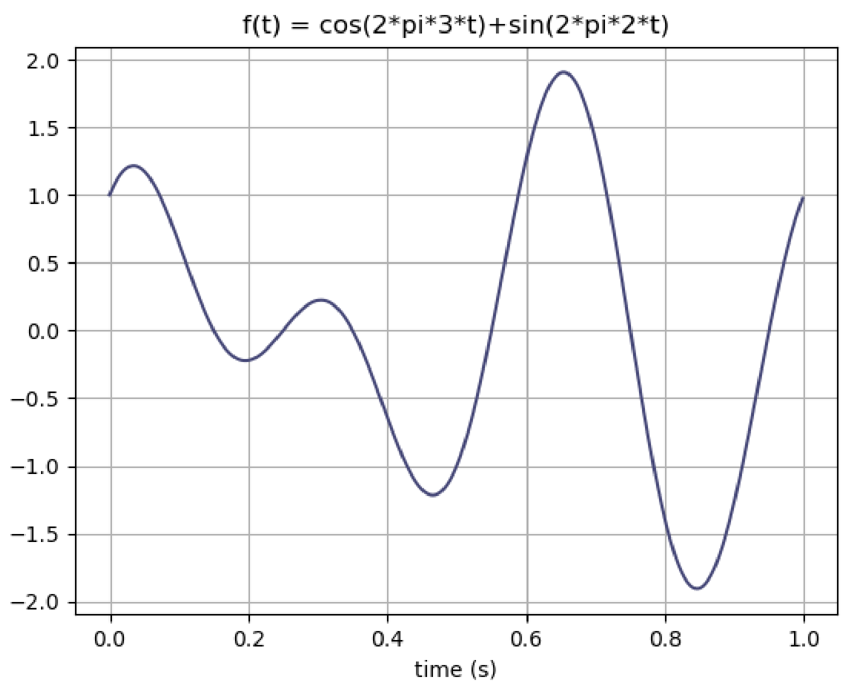

Figura 1: Señal original en el tiempo.

El gráfico presentado en la **Figura 1** tiene un rango de tiempo de 0 a 1 segundos, con un intervalo de $\frac{1}{500}$. Para graficar, se hace uso de la librería matplotlib. A continuación, se muestra el código implementado para este caso.

```
t = np.arange(0, 1, 1/500)
f = signal(t)
plt.plot(t, f,color = '#454777')
plt.grid(True)
plt.title("f(t) = cos(2*pi*3*t)+sin(2*pi*2*t)")
plt.xlabel("time (s)")
plt.show()
```

- Gráfico en la frecuencia: En la **Figura 2** se muestra una función con un ancho de banda (solamente considerando el cuadrante positivo) de 2 Hz.

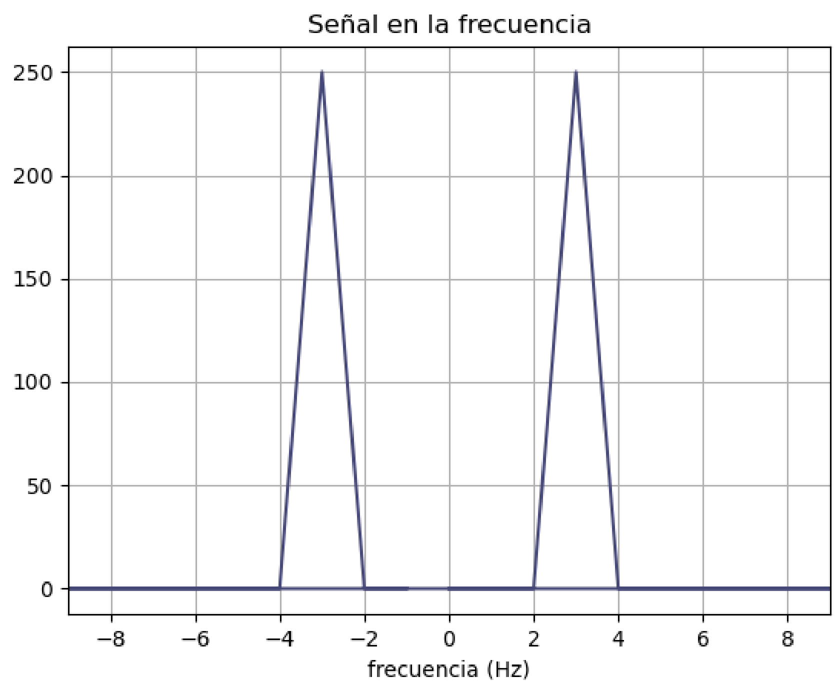

Figura 2: Señal original en la frecuencia.

Para poder obtener la señal original en la frecuencia, se procede a aplicar la Transformada de Fourier sobre la señal original en el tiempo. Para esto, se hace uso de la función fft perteneciente a la librería numpy. Posteriormente, se gráfica la función con un un intervalo de -9 a 9, respecto al eje X.

```python
F = np.fft.fft(f)
freq = np.fft.fftfreq(len(F), 1/500)
plt.figure()
plt.plot(freq, F.real,color = '#454777')
plt.grid(True)
plt.xlim([-9,9])
plt.title("Se al en la frecuencia")
plt.xlabel("frecuencia (Hz)")
plt.show()
```

- Gráfico en el tiempo de la portadora: se obtiene una portadora en el tiempo, específicamente, la representación gráfica de la función $cos(2\pi wt)$. Desde una vista general, no se puede identificar claramente el comportamiento de la función (**Figura 3**). Es por ello, que se procede a hacer zoom al segmento entre 0 y 1 segundos, mostrado en la **Figura 4**.

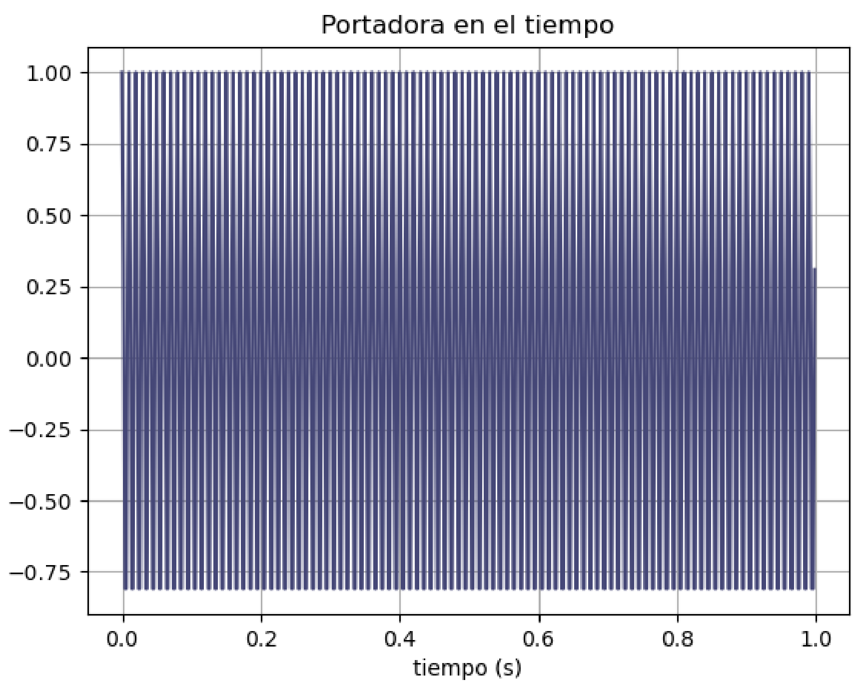

Figura 3: Portadora en el tiempo.

El zoom de la **Figura 4** permite comprobar que la portadora tiene 10 ciclos por por 0.1 segundos, es decir, en 1 segundo realiza 100 ciclos por segundo (recordar que 100 ciclos por segundos es equivalente a 100 Hz). Esto permite explicitar que la portadora obtenida gráficamente, corresponde a la solicitada en el enunciado.

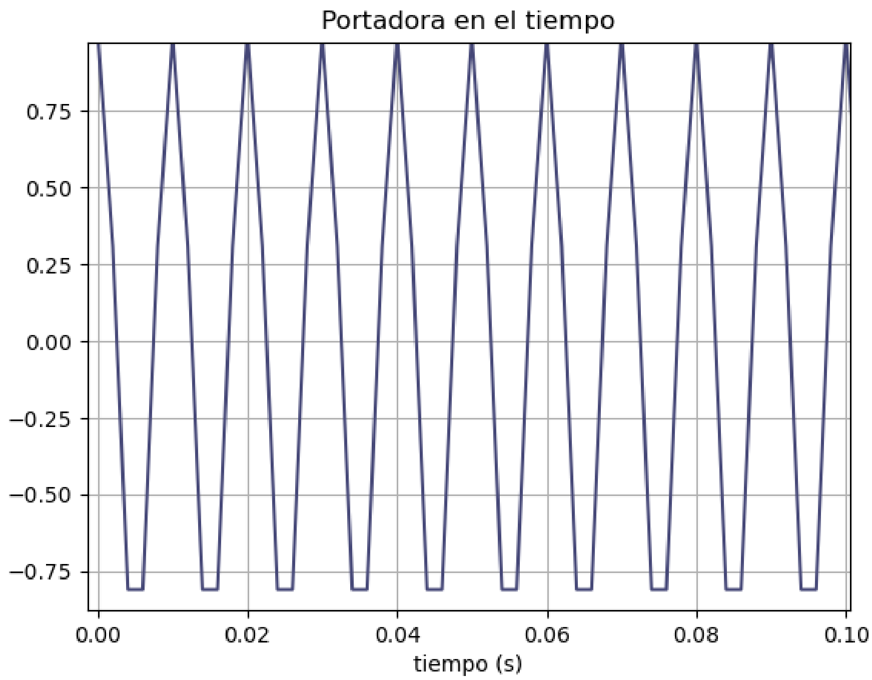

Figura 4: Portadora en el tiempo entre 0 y 1 segundo.

Para poder obtener los gráficos mostrados en la **Figura 3** y **Figura 4**, se escribe un código, el cual implementa la función coseno en una frecuencia de 100 Hz y un tiempo de 0 a 1 segundo.

```
w = 100
f_portadora = cos(2*pi*w*t)
plt.figure()
plt.plot(t, f_portadora,color = '#454777')
plt.grid(True)
plt.title("Portadora en el tiempo")
plt.xlabel("tiempo (s)")
plt.show()
```

- Gráfico en la frecuencia de la portadora: el gráfico de la función portadora (**Figura 6**) en términos de frecuencia, se obtiene gracias a la Transformada de Fourier. A simple vista no es posible ver el ancho de la banda, por lo que, se decide hacer un zoom en el cuadrante positivo de la imagen.

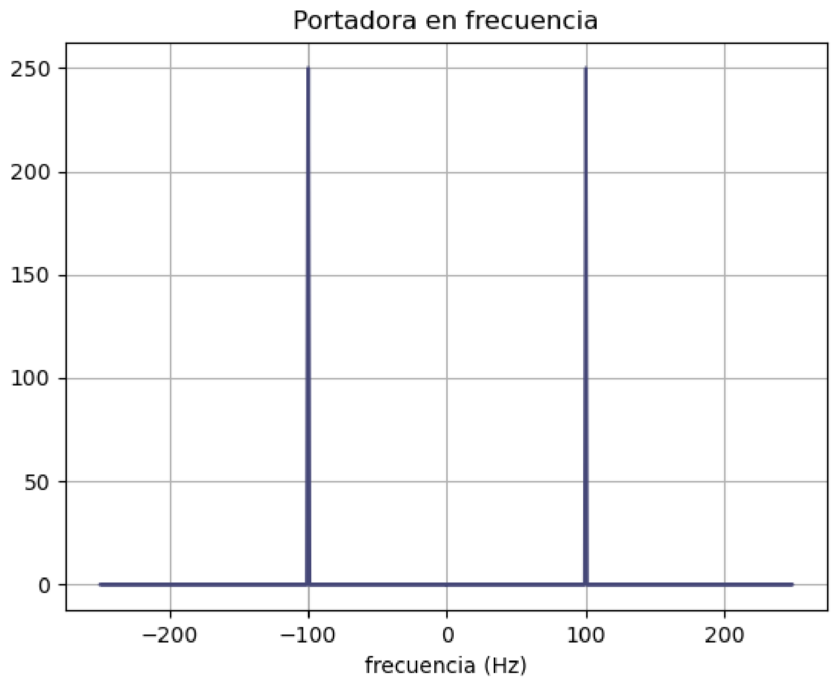

Figura 5: Portadora en la frecuencia.

Con el zoom, es posible notar un ancho de banda de 4 Hz, ubicado entre el rango de 99 y 101 Hz.

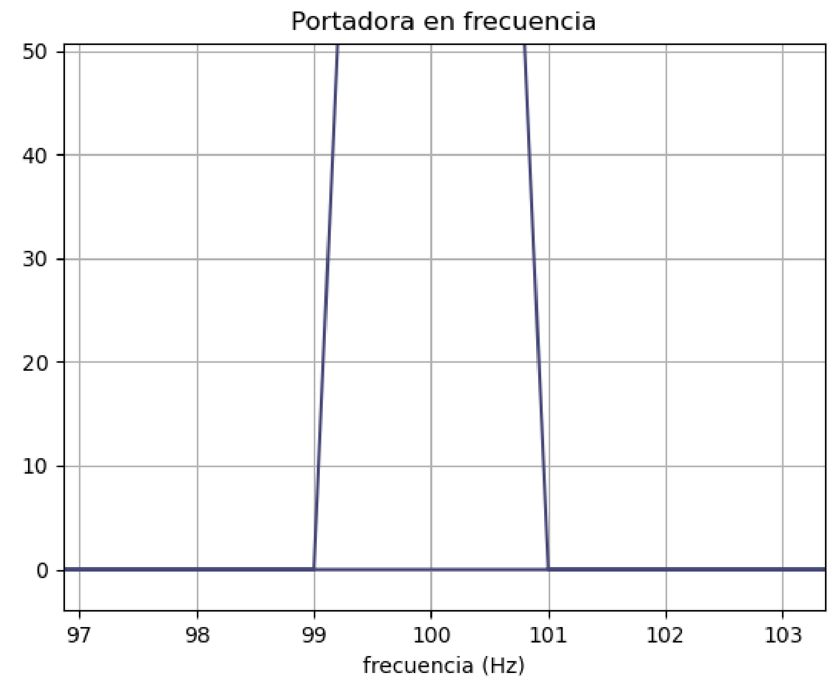

Figura 6: Portadora en la frecuencia en 97 y 103 Hz.

En el código elaborado, se hace uso de fft y fftfreq para obtener la portadora en términos de frecuencia.

```
F_portadora = np.fft.fft(f_portadora)
freq = np.fft.fftfreq(len(F_portadora), 1/500)
plt.figure()
plt.plot(freq, F_portadora.real,color='#454777')
plt.grid(True)
plt.title("Portadora en frecuencia")
plt.xlabel("frecuencia (Hz)")
plt.show()
```

- Gráfico en el tiempo de la señal modulada: la señal modulada en el tiempo (representada con un rojo pálido en la **Figura 7**), tiene las mismas amplitudes de la señal original (representada con un color morado en la **Figura 7**). Esto indica que la modulación se ha realizado correctamente.

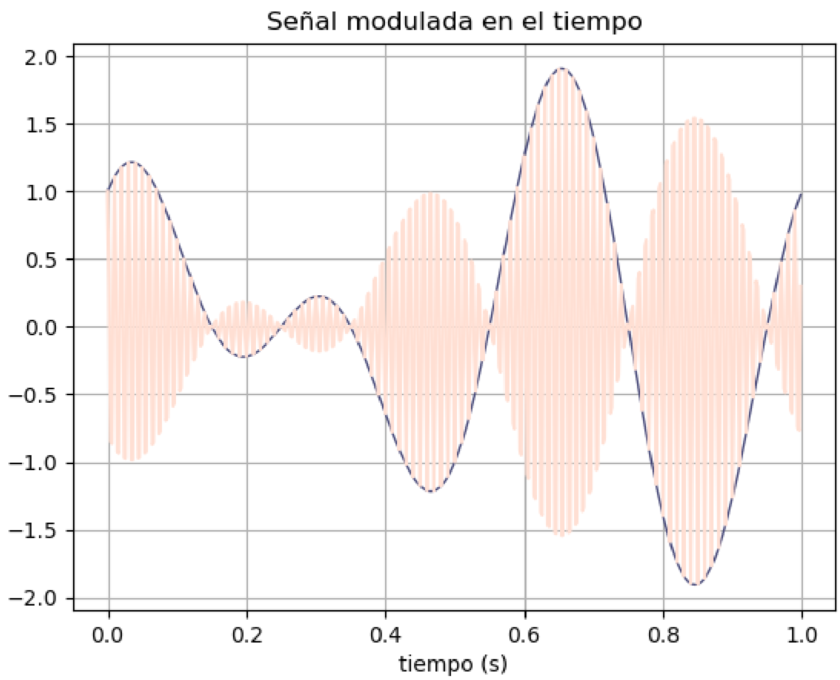

Figura 7: Señal modulada en el tiempo.

Además de la señal modulada en el tiempo, también es posible obtener la señal modulada en las frecuencias. Esto, permite obtener un ancho de banda que será detallado en la respuesta de la pregunta b. Al igual que en los casos anteriores, como en el gráfico de la **Figura 8**) no es posible de interpretar, se realiza zoom sobre el cuadrante positivo, obteniendo la información de la **Figura 9**)

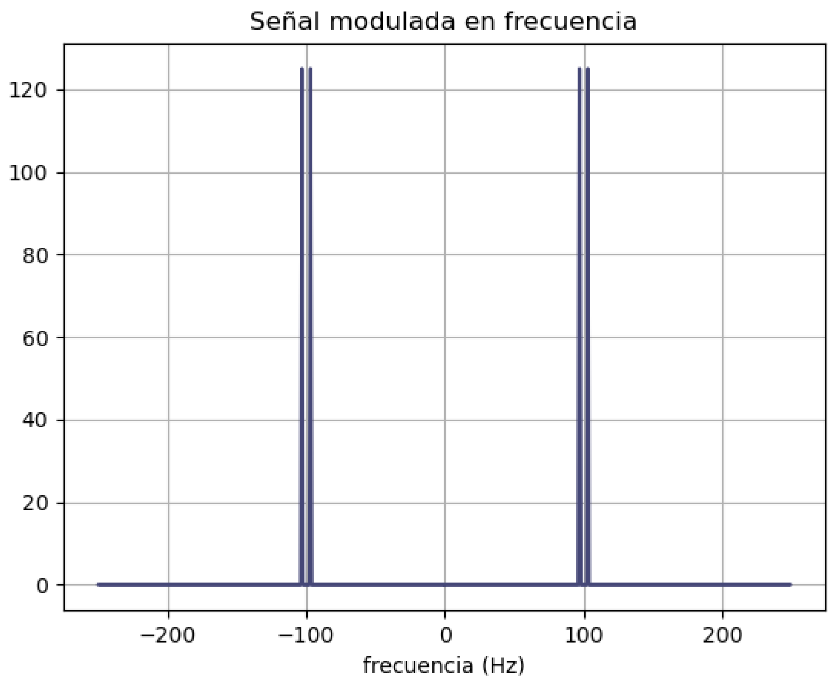

Figura 8: Señal modulada en las frecuencias.

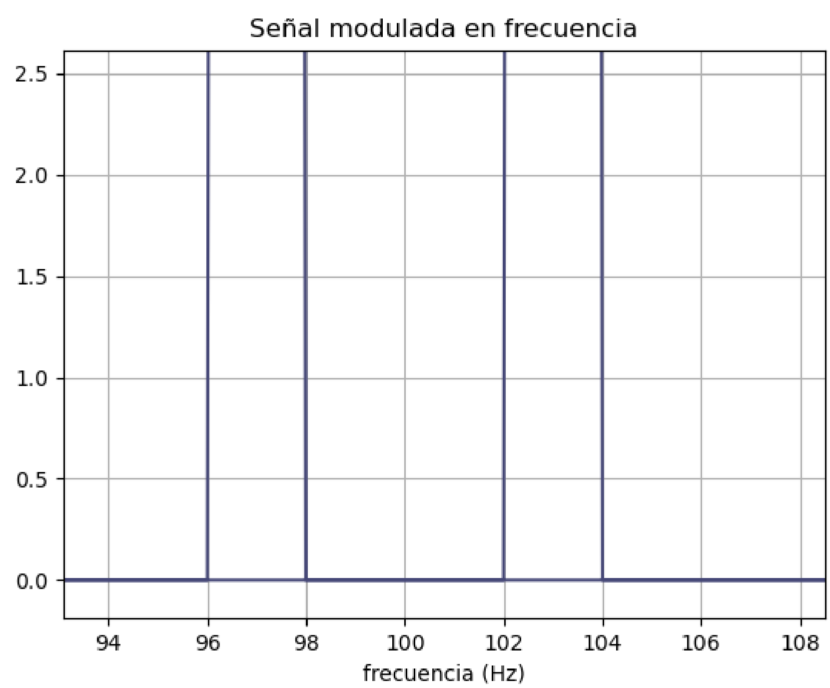

Figura 9: Señal modulada en las frecuencias entre 94 Hz y 108 Hz.

La señal modulada en terminos de frecuencia, también se puede obtener de la Transformada de Fourier, mediante la función fft de la librería numpy.

```
f_modulada = np.fft.fft(f_modulada)
freq = np.fft.fftfreq(len(f_modulada), 1/500)
plt.figure()
plt.plot(freq, f_modulada.real,color = '#454777')
plt.grid(True)
plt.title("Se al modulada en frecuencia")
plt.xlabel("frecuencia (Hz)")
plt.show()
```

b) Compruebe que el ancho de banda de la señal modulada es el doble del ancho de banda de la señal original.

Solución:

La teoría indica que el ancho de banda de la señal modulada, debe ser el doble al ancho de banda de la señal original. Como se puede comprobar en la **Figura 2**, el ancho de banda de la señal original es de 2 Hz, mientras que en la **Figura 9** es posible ver que cada peak presentado, tiene un ancho de banda de 2 Hz cada uno. En otras palabras, el ancho de banda de la señal modulada en terminos de frecuencias, es igual a la suma de los dos peaks presentado, es decir, 4 Hz. Por lo tanto, se concluye que la teoría antes mencionada tiene veracidad.