Colocación de las preguntas en cada parte del informe:

**Análisis del Sps:**

g. Se supone que una señal binaria aleatoria de forma rectangular tiene un ancho de banda infinito, ¿coincide esto con lo observado en GNU Radio y por qué?

h. ¿Qué fórmula podría ayudar a calcular el número de lóbulos de la PSD de señal binaria aleatoria de forma rectangular cuando se conoce la frecuencia de muestreo y Sps? Nota: el lóbulo de la mitad se cuenta como dos porque tiene el doble de ancho que los demás.

i. ¿Cómo se calcula todo el rango de frecuencias que ocupa el espectro cuando se conoce Rb y Sps?

o. ¿Para qué caso de Sps la PSD de una señal binaria aleatoria bipolar es similar a la PSD de ruido blanco?

**Análisis de Ruido:**

f. Se supone que el ruido blanco tiene un ancho de banda infinito, ¿coincide esto con lo observado en GNU Radio?, ¿por qué?

**Análisis de Imagen RGB:**

**Análisis de Audio:**

**Contraste de Audio e Imagen:**

c. ¿Por qué razón la PSD de las señales binarias que provienen de una señal de audio es diferente a la que proviene de una foto siendo ellas igualmente señales binarias bipolares de forma rectangular?

**Preguntas de Control:**

* **Funcionamiento del sistema:**

a. ¿Qué papel juega la siguiente combinación de bloques?

b. ¿Qué papel juega el bloque “Interpolationg FIR Filter”, ¿cómo funciona?

1. ¿Por qué el parámetro “Interpolation” en el bloque vale “SPS” y qué pasa si se coloca otro valor?

2. Si tuviese que analizar la señal en p3, ¿qué cambios realizaría en la instrumentación (esquema de GNU Radio)?

3. ¿Qué fórmula permite conocer el ancho de banda de la señal en p4 si se conoce Rb y Sps?

4. ¿Qué fórmula permite conocer la frecuencia de muestreo en p3, si se conoce la frecuencia de muestreo en p4 y Sps?

d. ¿Qué papel juega el bloque “Throttle”?

e. ¿Qué pasaría con la PSD si no se hace la conversión a señal bipolar, sino que la señal binaria en p4 solo tiene valores de 0 ó 1 en lugar de -1 ó 1?

**- Preguntas Extras:**

j. ¿Cómo se calcula la resolución espectral del analizador de espectros, cuando se conoce N y la frecuencia de muestreo?

k. ¿Qué pasaría si en el bloque “Unpack K Bits” se configura el parámetro K como 16?

l. ¿Cómo calcularía la frecuencia de muestreo a la entrada del bloque “Unpack K Bits” si conoce el número de lóbulos de la PSD y el ancho de banda de la señal?

m. ¿Cómo calcularía la frecuencia de muestreo a la salida del bloque “Unpack K Bits” si conoce la frecuencia de muestreo a la entrada?

n. ¿Cómo calcularía la frecuencia de muestreo a la salida del bloque “Char to Float” si conoce la frecuencia de muestreo a la entrada?

**Graficas:**

Cuando interpolas con las muestras **1,1,1,1,2,1,1,1,1**, el valor "2" en el centro de la secuencia crea un cambio repentino o "pico" en la señal. Este cambio brusco interrumpe la continuidad de los valores "1", lo que ocasiona que la señal final tenga un salto visible en ese punto.

Este tipo de variación rápida provoca que se añadan **frecuencias más altas** en la señal, ya que, en el procesamiento de señales, los cambios abruptos en el tiempo tienden a generar componentes de alta frecuencia. El resultado es una señal que no es completamente suave y presenta una pequeña distorsión o "sobresalto".

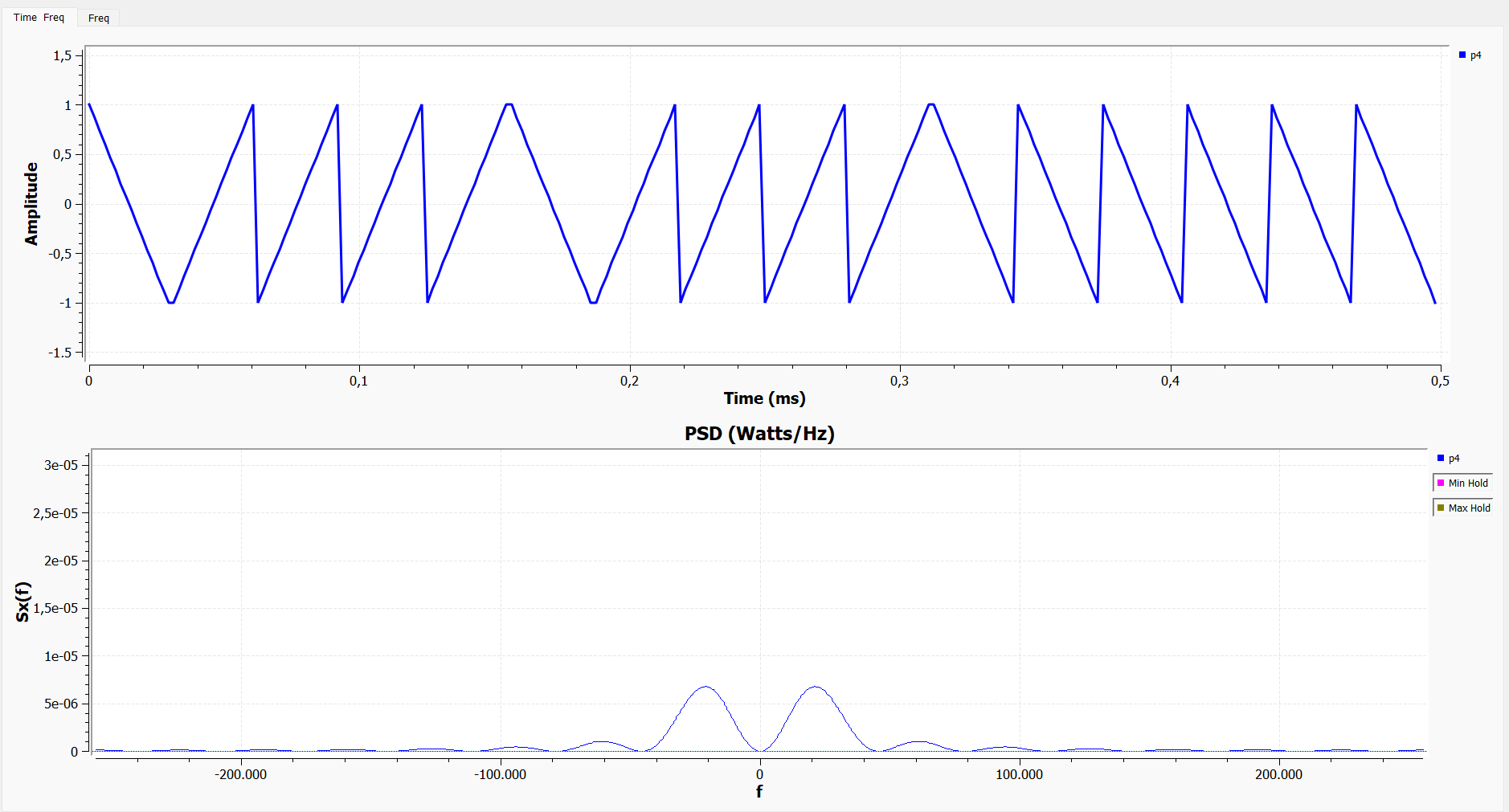
En resumen: la muestra "2" rompe la uniformidad de los valores "1" y provoca un cambio brusco en la señal, lo que genera efectos no deseados en su comportamiento.

Es como multiplicar con la señal, ya que no es un valor fijo o suave, sino con cambios repentinos que se ven en la señal.

|  |  |  |  |
| --- | --- | --- | --- |
| Señal | Valor de [h] | Constant Source | Interpolating FIR |
| Dientes de Cierra | *np.array*  *([-1, -0.866, -0.733, -0.6,*  *-0.466, -0.333, -0.2, -0.066,*  *0.0666, 0.2, 0.3333, 0.4666,*  *0.6, 0.7333, 0.8666, 1])* | -500m | 16 |
| Unipolar RZ | np.array  ([0.5, 0.5, 0.5, 0.5,  0.5, 0.5, 0.5, 0.5,  0, 0, 0, 0,  0, 0, 0, 0]) | 0 | 16 |
| Manchester NRZ | np.array  ([1, 1, 1, 1,  1, 1, 1, 1,  -1, -1, -1, -1,  -1, -1, -1, -1]) | -500m | 16 |
| OOK | np.sin(np.linspace(0, 4\*np.pi, 32)) | 0 | 32 |
| BPSK | np.sin(np.linspace(0, 8\*np.pi, 64)) | -500m | 64 (por resolución) |
| Latidos del Corazón | np.array([0, -0.25, -0.5, -0.25, 0, 0.25, 0.5, 0.25, 0, 0, 0, 0, 0, 1, 2, 1, 0, -1, -2, -1, 0, 0.5, 1.0, 0.5, 0, 0.25, 0.5, 0.25, 0, -0.25, -0.5, -0.25]) | 0 | 32 |
| Pulsos Rizados | np.array([0, 0.0571, 0.1142, 0.1709, 0.2266, 0.2824, 0.3366, 0.3896, 0.4425, 0.4922, 0.5411, 0.5897, 0.6332, 0.6768, 0.7184, 0.7556, 0.7928, 0.8259, 0.8559, 0.8858, 0.9091, 0.9311, 0.9527, 0.9705, 0.9884, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1]) | -500m | 64 (por resolución) |

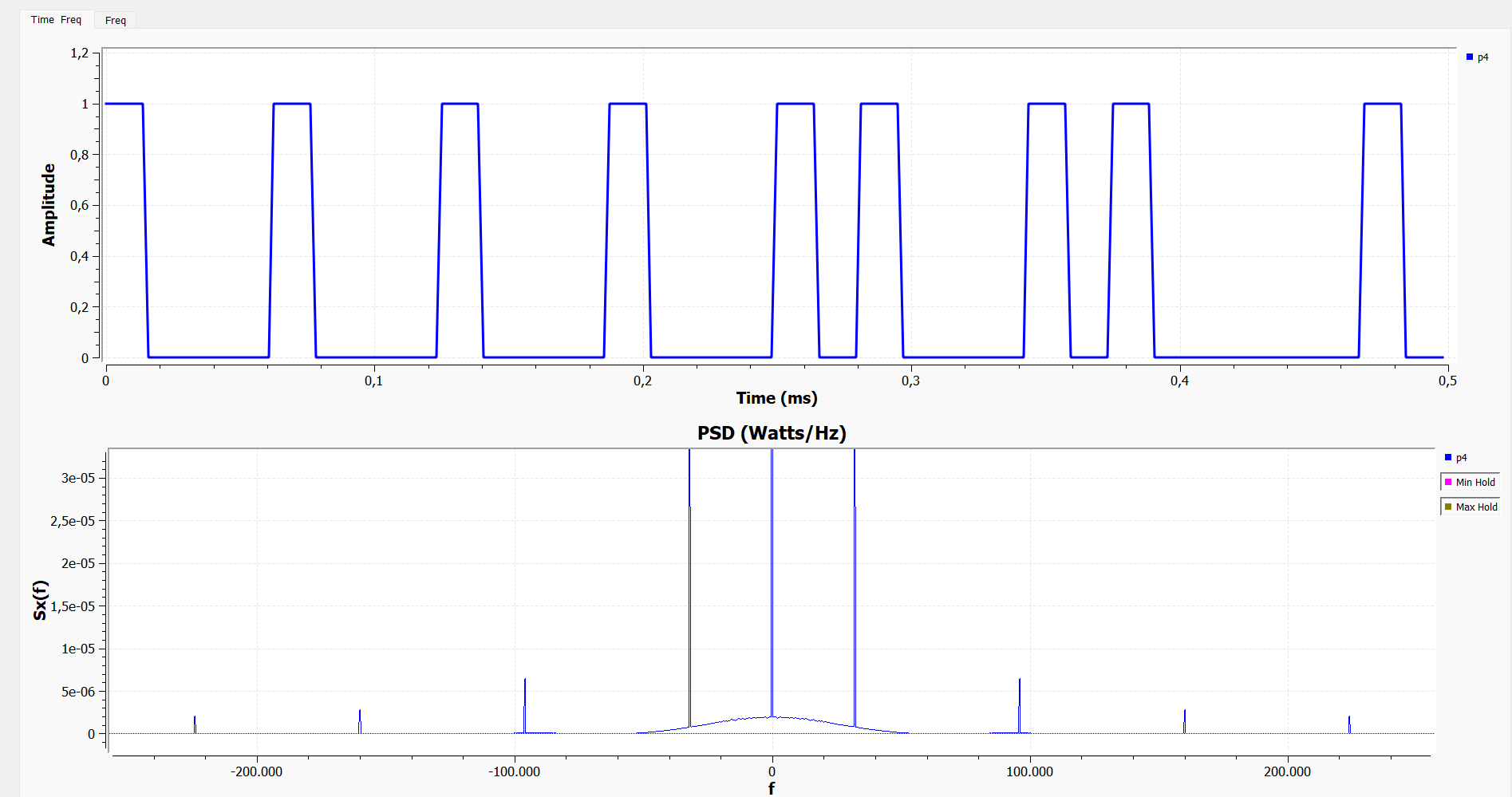
*p. ¿Qué cambios mínimos haría al flujograma, manipulando principalmente h, si desea que los bits en la señal binaria aleatoria tomen la forma de dientes de sierra?*

*np.array ([-1, -0.866, -0.733, -0.6, -0.466, -0.333, -0.2, -0.066, 0.0666, 0.2, 0.3333, 0.4666, 0.6, 0.7333, 0.8666, 1])*

**

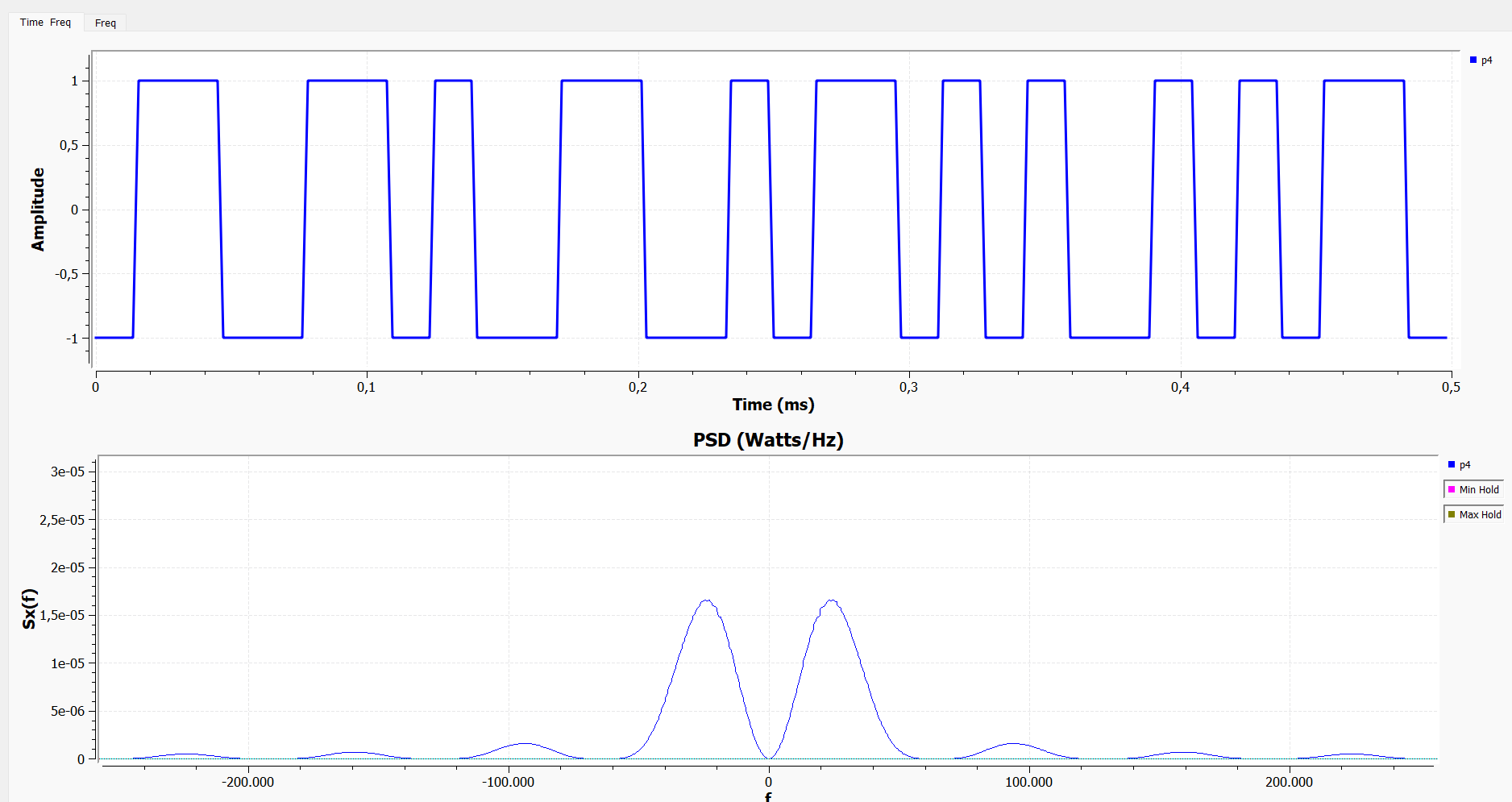
*q. ¿Qué cambios mínimos haría al flujograma, manipulando principalmente h, si desea que la señal binaria aleatoria tenga codificación de línea Unipolar RZ, es decir como se muestra en la Fig. 3?*

*np.array([0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0,0,0,0,0,0,0,0]) y la fuente en 0*

**

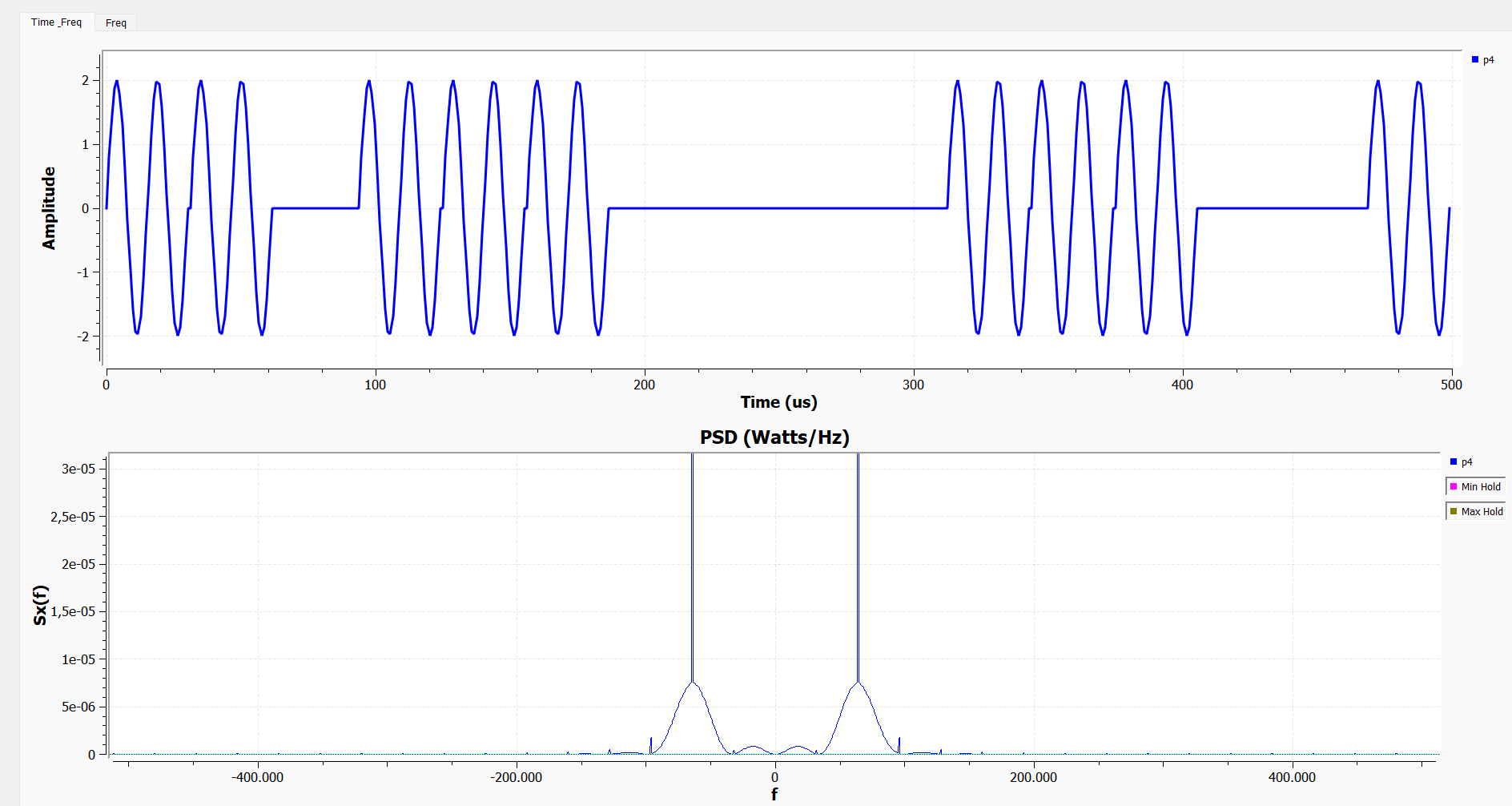
*r. ¿Qué cambios mínimos haría al flujograma, manipulando principalmente h, si desea que la señal binaria aleatoria tenga codificación de línea Manchester NRZ, es decir como se muestra en la Fig. 3?*

np.array([1,1,1,1,1,1,1,1,-1,-1,-1,-1,-1,-1,-1,-1])

**

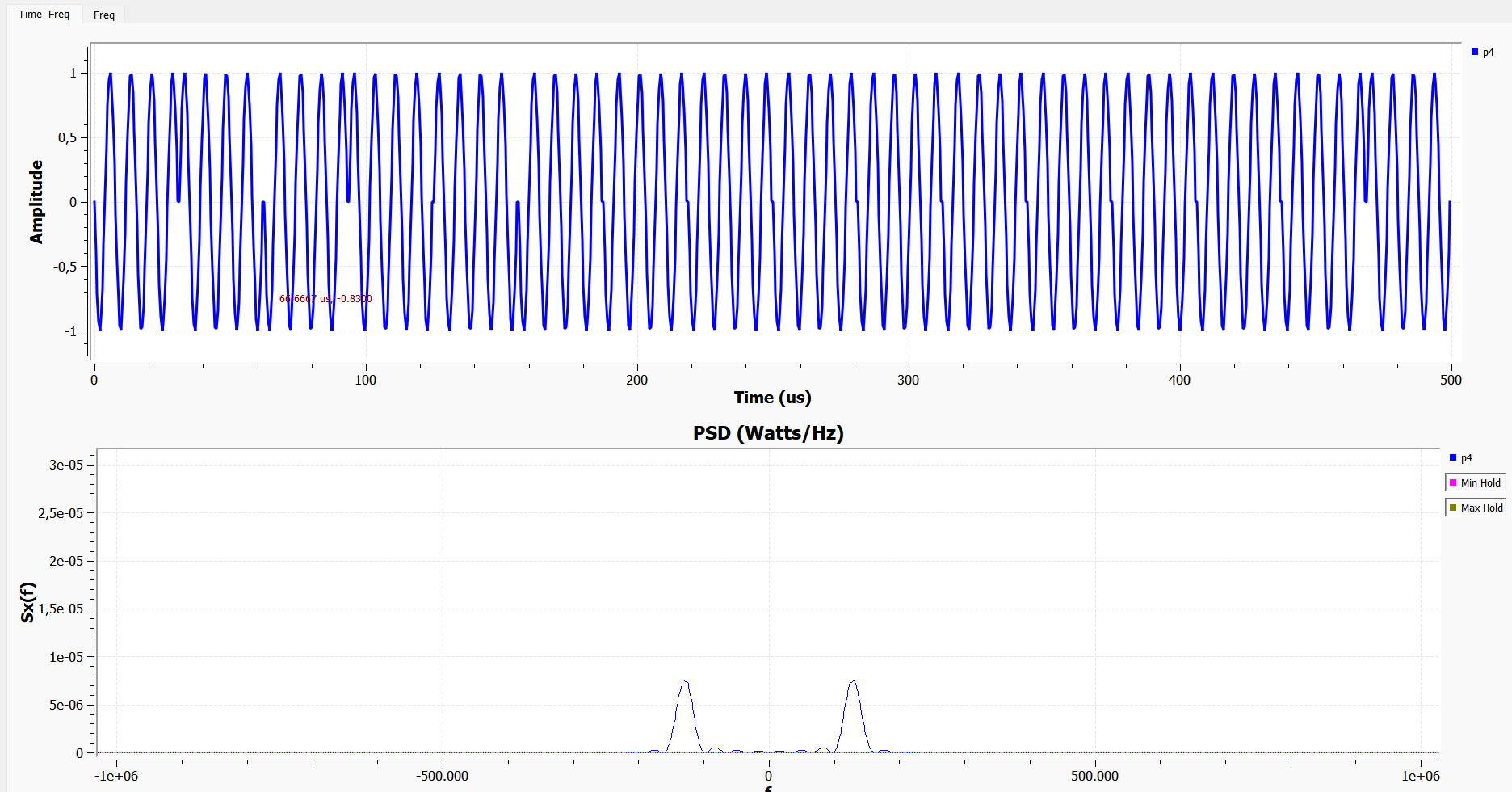
*s. ¿Qué cambios mínimos haría en el flujograma, aprovechando h y el FIR Interpolating Filter para que la señal binaria tenga la forma de señal OOK como se muestra en la Figura 4?*

np.sin(np.linspace(0, 4\*np.pi, 32))



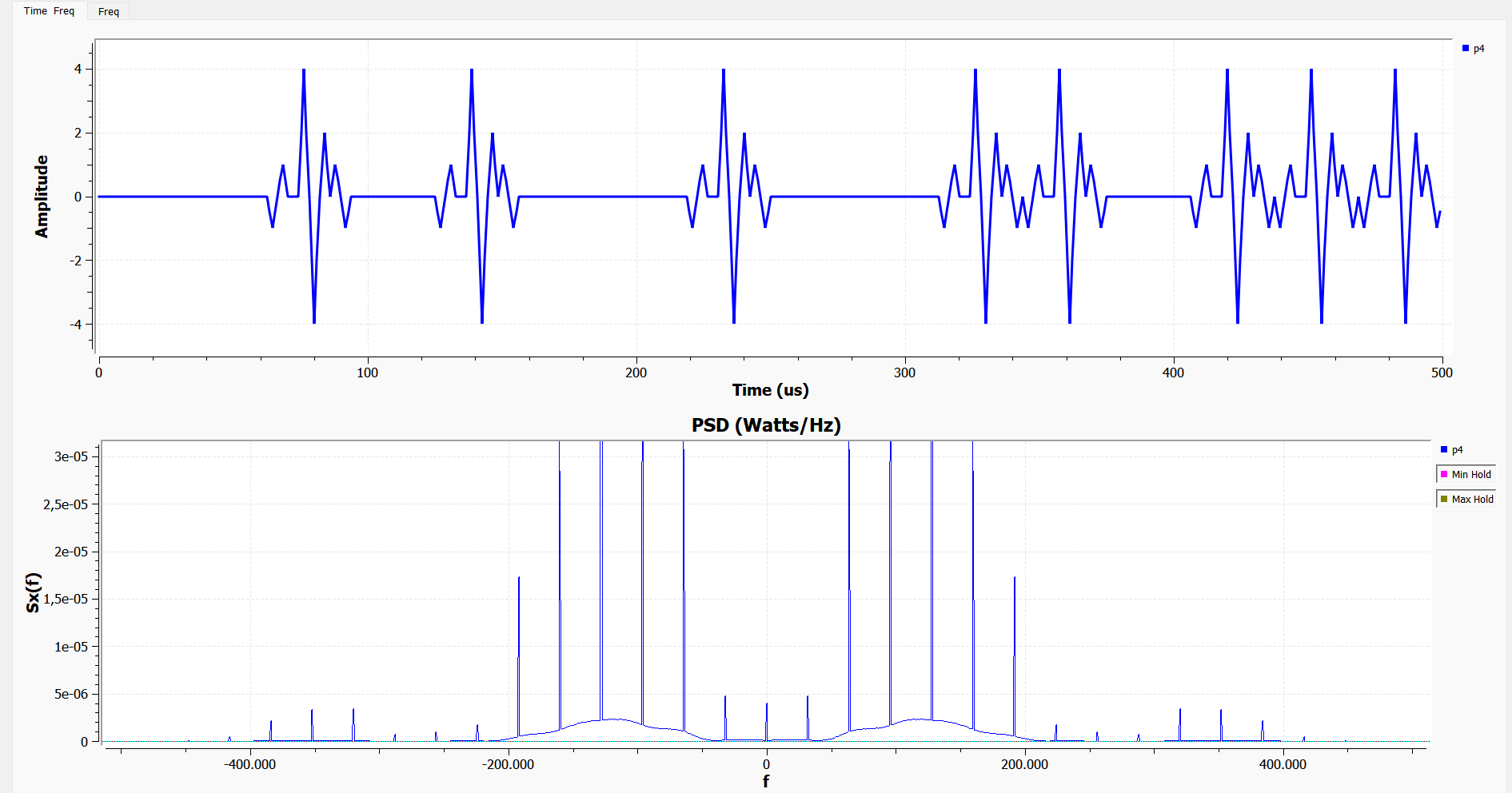
*t. ¿Qué cambios mínimos haría en el flujograma, aprovechando h y el FIR Interpolating Filter para que la señal binaria tenga la forma de señal BPSK como se muestra en la Figura 4?*

np.sin(np.linspace(0, 8\*np.pi, 64))



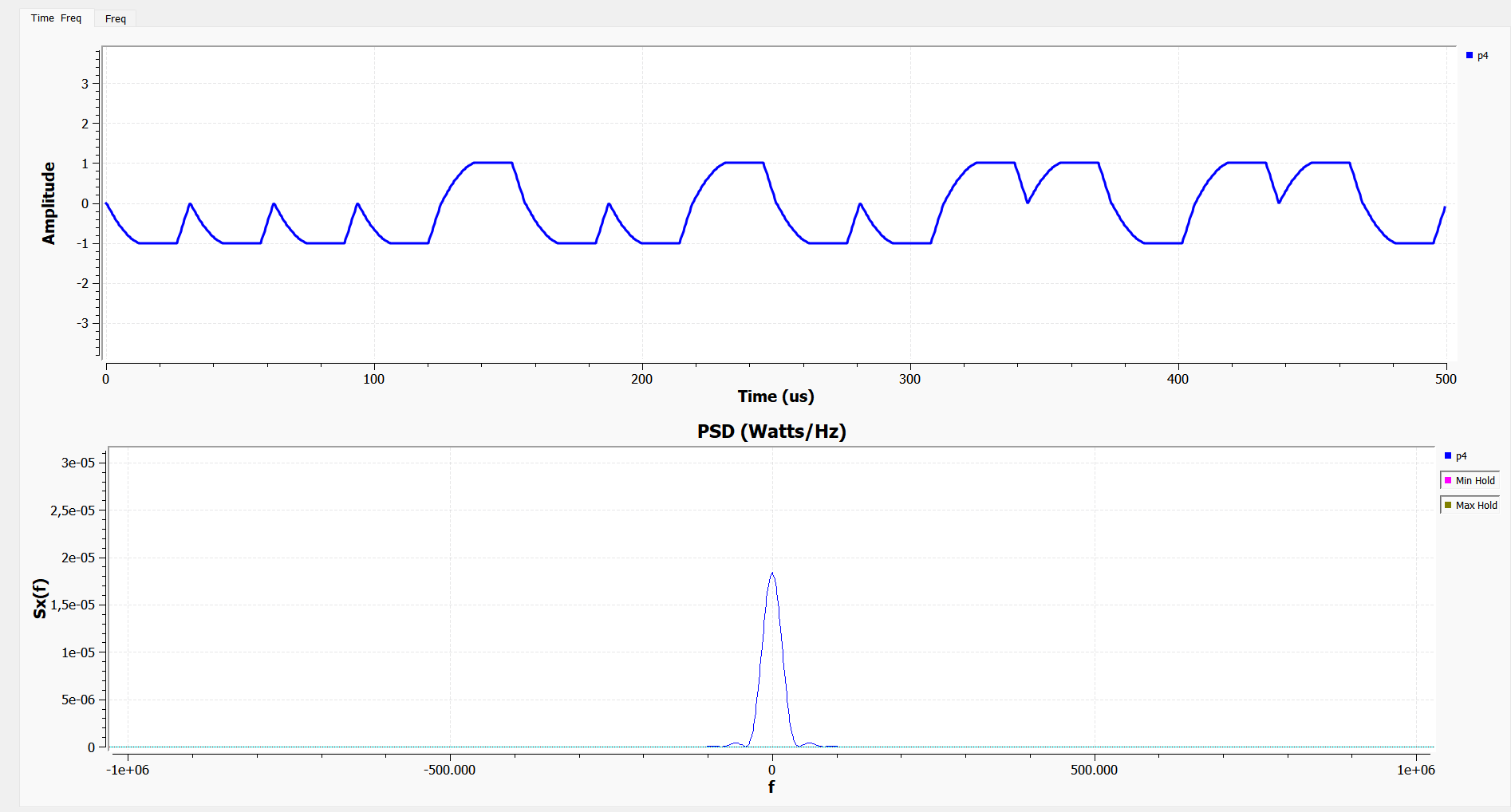
*u. ¿Qué cambios mínimos haría en el flujograma, aprovechando h y el FIR Interpolating Filter para que la señal binaria tenga la forma de los latidos del corazón como se muestra en la Figura 5?*

np.array([0, -0.25, -0.5, -0.25, 0, 0.25, 0.5, 0.25, 0, 0, 0, 0, 0, 1, 2, 1, 0, -1, -2, -1, 0, 0.5, 1.0, 0.5, 0, 0.25, 0.5, 0.25, 0, -0.25, -0.5, -0.25])

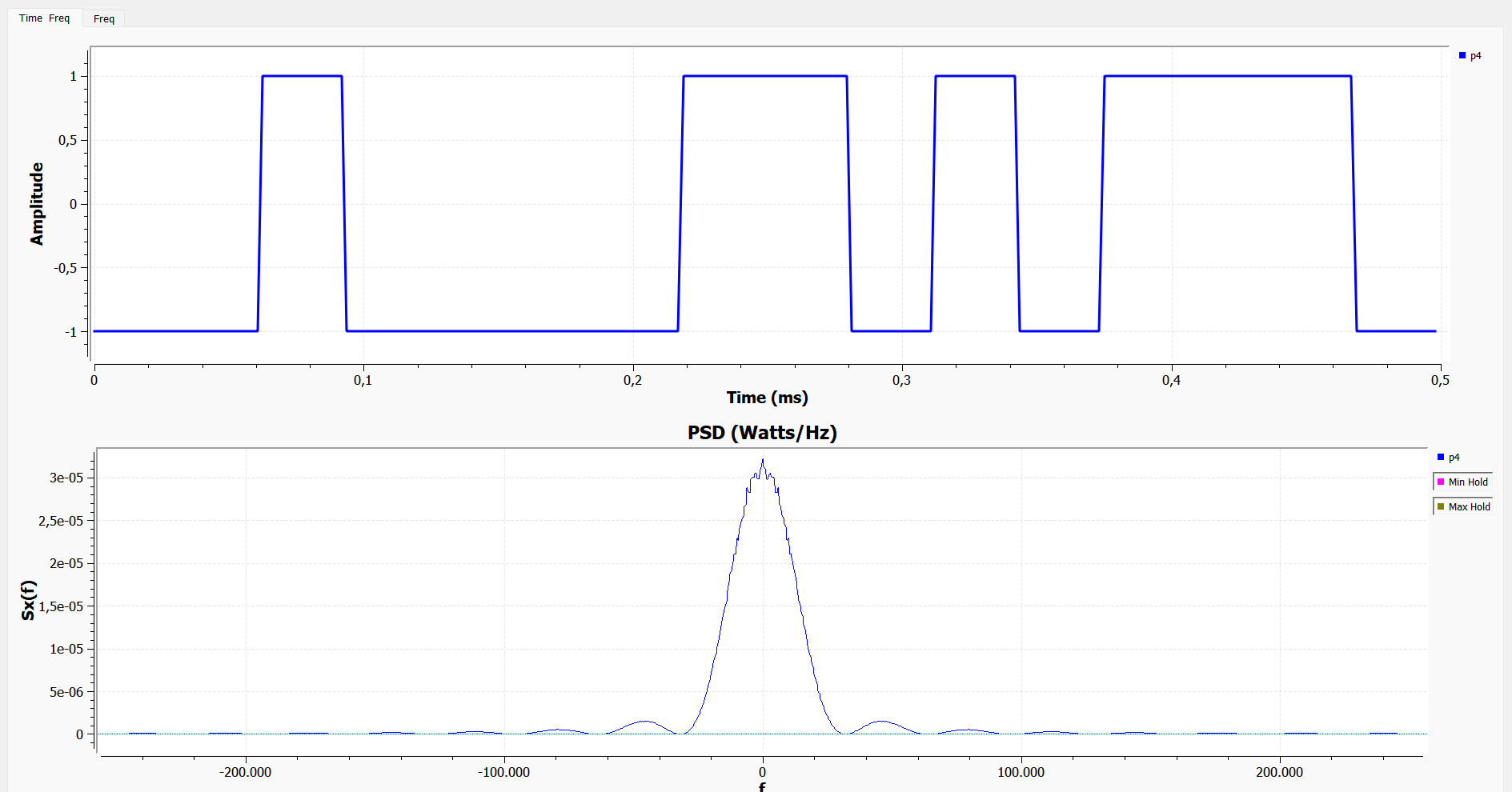
**

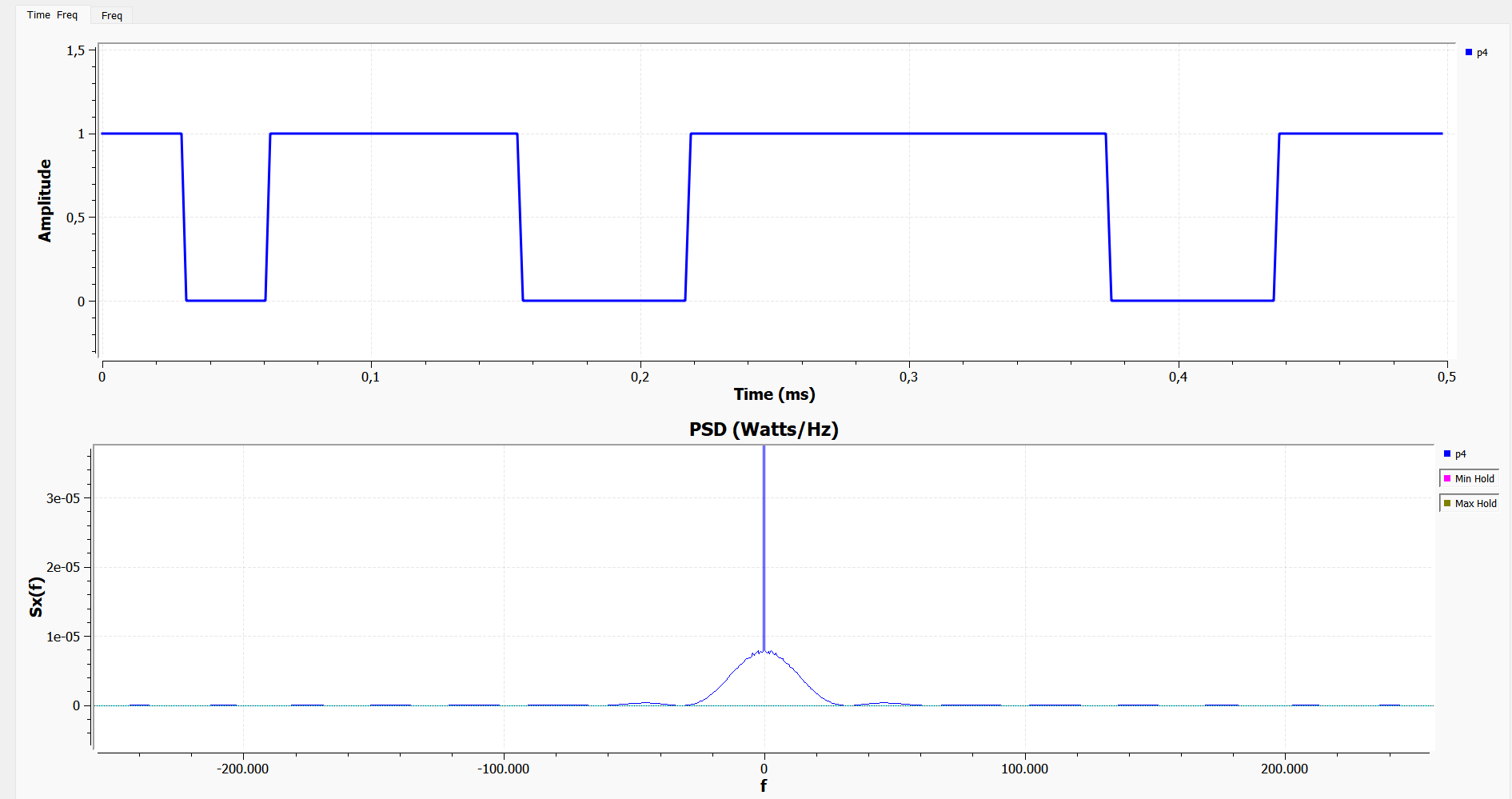
*v. ¿Qué cambios mínimos haría en el flujograma, aprovechando h y el FIR Interpolating Filter para que la señal binaria tenga la forma que se muestra en la Figura 6?*

np.array([0.0, 0.0571, 0.1142, 0.1709, 0.2266, 0.2824, 0.3366, 0.3896, 0.4425, 0.4922, 0.5411, 0.5897, 0.6332, 0.6768, 0.7184, 0.7556, 0.7928, 0.8259, 0.8559, 0.8858, 0.9091, 0.9311, 0.9527, 0.9705, 0.9884, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1])

**

*w. Explique usando gráficas de PSD la diferencia que existe entre la PSD de una señal binaria bipolar y una unipolar.*

**

**