

Software Radio

2024-2025

Alberto Marquillas

Diego André Gonçalves

Índice

Introducción.....	1
Qpsk generator	2
Zero padding	4
LPF	5
NCO	6
Modulador QPSK	9
Conclusiones	13

Introducción

La presente práctica tiene como objetivo la implementación y simulación de un sistema de modulación digital en cuadratura (QPSK) utilizando el lenguaje de descripción hardware VHDL. Esta actividad forma parte del laboratorio de la asignatura *Software Radio* y pretende consolidar los conocimientos adquiridos sobre diseño digital, procesamiento de señales y técnicas de comunicación modernas aplicadas a sistemas reconfigurables.

El sistema diseñado simula un transmisor experimental que opera en la banda VHF, utilizando una portadora de 40,7 MHz y una tasa de transmisión de 1 Mbit/s. Para ello, se desarrolla un modulador QPSK compuesto por varios bloques funcionales interconectados: un generador de símbolos pseudoaleatorios, un interpolador basado en inserción de ceros y filtrado pasa-bajo (LPF), un oscilador digital (NCO), y un modulador en fase y cuadratura.

A través de distintas simulaciones parciales y una simulación final del sistema completo, se verifica el correcto funcionamiento de cada uno de los bloques, así como la coherencia de la señal final transmitida (Data2Ant). La práctica también incluye un análisis en MATLAB del espectro generado, que permite evaluar la calidad del proceso de modulación y la conformidad con los requisitos del sistema.

Qpsk generator

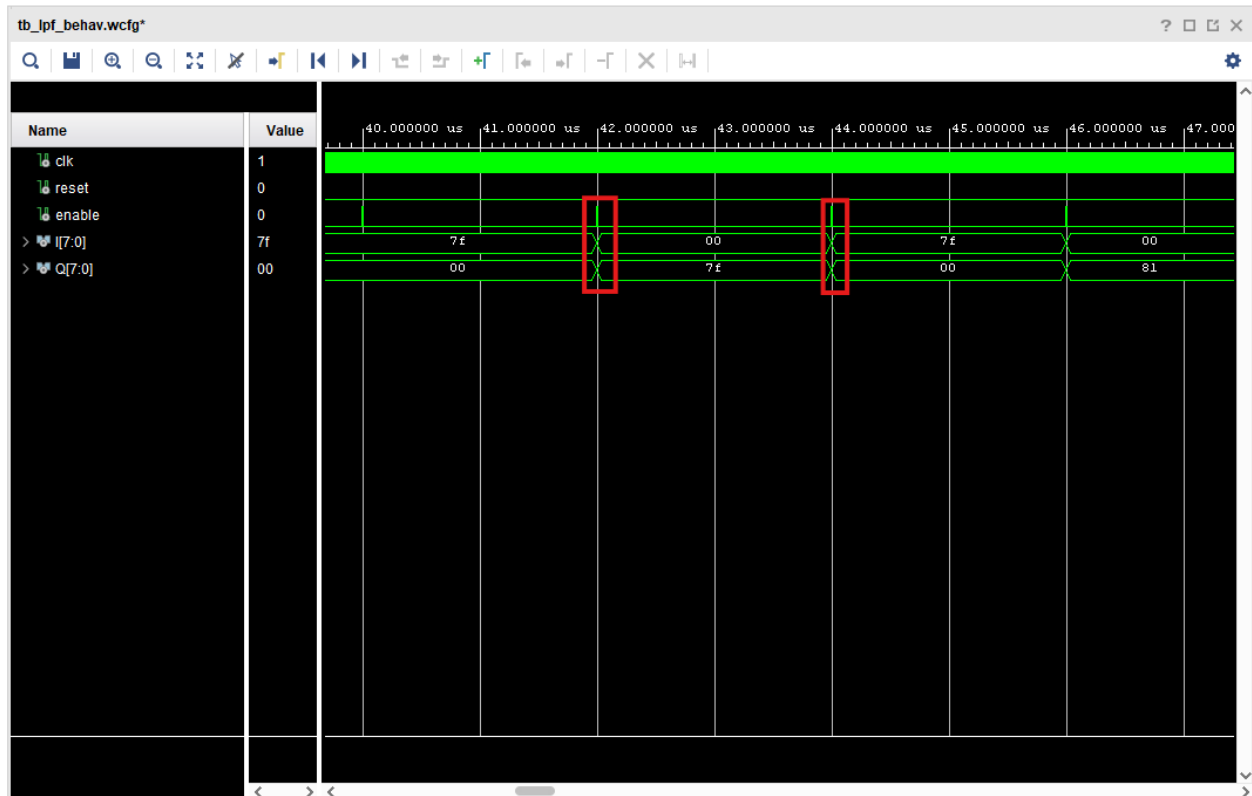
Este bloque implementa un generador de símbolos QPSK a partir de dos registros LFSR (Linear Feedback Shift Register) de 11 bits cada uno, los cuales generan secuencias pseudoaleatorias. Cada registro se actualiza desplazando los bits y aplicando una operación XOR entre el último y el antepenúltimo bit del registro, lo que permite obtener un nuevo bit pseudoaleatorio en cada ciclo habilitado.

Los dos bits pseudoaleatorios generados (random_bit1 y random_bit0) se combinan para formar una de las cuatro posibles combinaciones binarias (00, 01, 10, 11). Cada combinación se mapea directamente a un símbolo QPSK, representado por las componentes en fase (I) y en cuadratura (Q), de acuerdo con la tabla de mapeo especificada. Estos valores se cuantifican a 8 bits en complemento a dos (CA2).

La actualización de los registros y la generación de un nuevo símbolo I/Q solo se produce en los flancos activos del reloj cuando la señal enable está activa (a 1), lo cual ocurre a una frecuencia de 500 kHz. De este modo, se garantiza una tasa de generación de símbolos de 500 kSym/s, adecuada para el posterior procesamiento por el interpolador y el modulador I/Q.

Combinació (random_bit1 i random_bit 0)	Número de Símbol	I (Figura 7)	Q (Figura 7)	I (quantitzat amb 8 bits)	Q (quantitzat amb 8 bits)
0 (00)	0	1	0	01111111 (127 en CA2)	00000000 (0 en CA2)
1 (01)	1	0	1	00000000 (0 en CA2)	01111111 (127 en CA2)
2 (10)	2	-1	0	10000001 (-1 en CA2)	00000000 (0 en CA2)
3 (11)	3	-1	-1	00000000 (0 en CA2)	10000001 (-127 en CA2)

Generador de símbolos en función de los bits random.



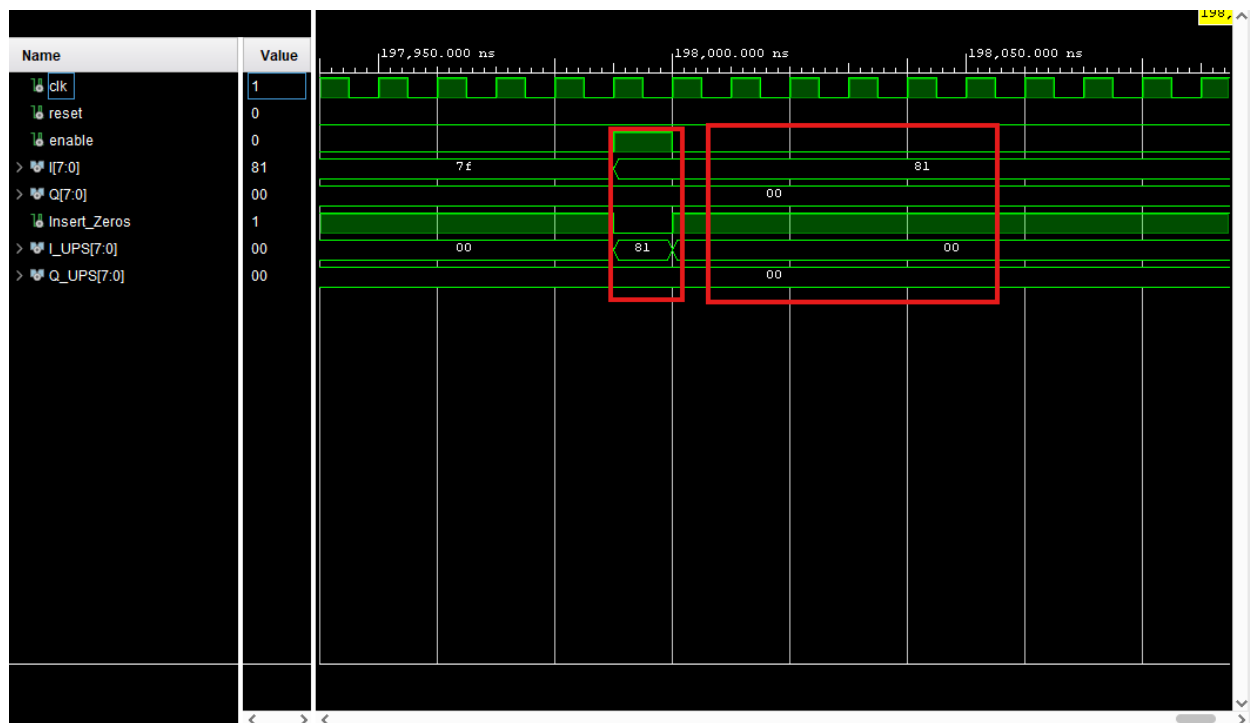
Generador QPSK justo en el momento que se activa el Enable.

Zero padding

Este bloque se encarga de realizar la interpolación necesaria para adaptar la frecuencia de muestreo de los símbolos QPSK desde 500 kHz hasta 100 MHz, tal como requiere el sistema. La interpolación se realiza mediante la inserción de ceros entre las muestras originales.

Cuando la señal de control Insert_Zeros está activa (Insert_Zeros = '1'), el bloque fuerza la salida a cero, insertando muestras nulas. En cambio, cuando Insert_Zeros = '0', el bloque propaga directamente los valores de entrada provenientes del generador QPSK hacia las salidas I_UPS y Q_UPS.

De este modo, se consigue una secuencia de datos donde cada símbolo válido de entrada es seguido por 199 ceros, lo cual multiplica la frecuencia de muestreo de 500 kHz a 100 MHz. Esta técnica es esencial para preparar la señal antes de aplicarle el filtro paso bajo (LPF), que eliminará las réplicas espectrales generadas por la interpolación.



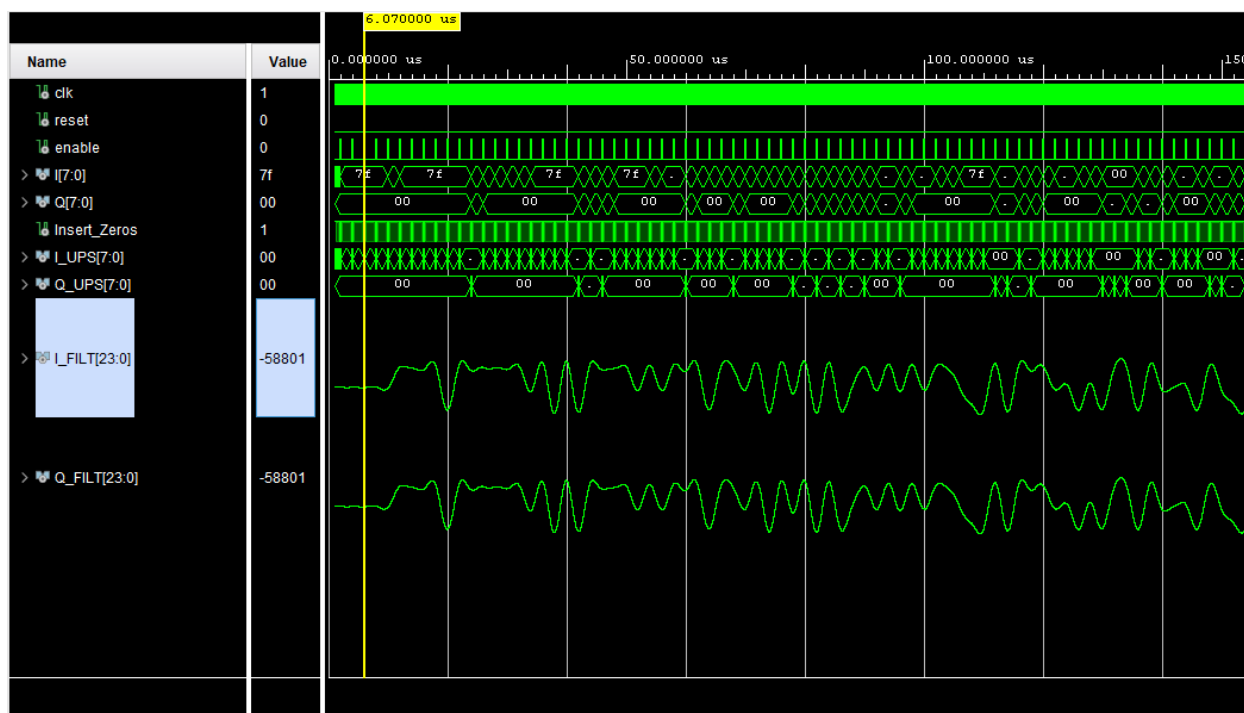
Zero_Padding cuando enable es 1 e insert_zero es 0.

LPF

Este bloque aplica un filtro FIR con respuesta en coseno realzado (Root Raised Cosine) a las señales I y Q ya interpoladas. Su función es eliminar las réplicas espectrales introducidas por el proceso de **zero padding**, suavizando así la transición entre símbolos y limitando el ancho de banda del canal.

En la simulación se observa que las señales I_UPS y Q_UPS, tras ser interpoladas, contienen secuencias de valores nulos (ceros) intercaladas con los valores reales de los símbolos. El filtro LPF opera sobre estas señales y produce en sus salidas I_FILT y Q_FILT unas formas de onda continuas y suavizadas, resultado de la convolución con la respuesta impulsional del filtro.

Estas salidas presentan la forma característica de una señal interpolada y filtrada, preparada para ser modulada sobre una portadora. Es importante destacar que el filtro no solo elimina componentes de alta frecuencia no deseadas, sino que también reconstruye adecuadamente el contorno temporal de los símbolos QPSK originales.



Respuesta del filtro LPF: se observa cómo las señales I y Q se suavizan tras la interpolación, eliminando las discontinuidades introducidas por el zero padding.

Para enerar una portadora sinusoidal en fase y cuadratura (coseno y seno) centrada a 40.7 MHz, con una frecuencia de muestreo de 100 MHz, para la modulación I/Q del señal QPSK.

Para un DDS (Direct Digital Synthesizer), el incremento de fase que determina la frecuencia de salida se calcula mediante la fórmula:

$$incremento\ de\ fase = \frac{F_{out} \cdot 2^{[32]}}{F_{clk}}$$

$$incremento\ de\ fase = \frac{40.7MHz \cdot 2^{[32]}}{100MHz} = 1748051689$$

incremento de fase (binario) = 01101000001100010010011011101001

DDS Compiler (6.0)

DocumentationIP Location

IP SymbolInformation

Show disabled ports

M_AXIS_DATA
M_AXIS_PHASE
event_pinc_invalid
event_poff_invalid
event_phase_in_invalid
event_s_phase_tlast_missing
event_s_phase_tlast_unexpected
event_s_phase_chanid Incorrect
event_s_config_tlast_missing
event_s_config_tlast_unexpected

S_AXIS_PHASE
S_AXIS_CONFIG
ack
acklen
aresetn

Component Namedds_compiler_0

ConfigurationImplementationDetailed ImplementationPhase Angle Increment ValuesSummaryAdditional Summary

Configuration OptionsPhase Generator and SIN COS LUT

System Requirements

System Clock (MHz)100[0.01 - 1000.0]

Number of Channels1

Mode Of OperationStandard

Frequency per Channel (Fs)100.0 MHz

SIN/COS Output TypeInteger

Parameter SelectionHardware Parameters

Noise ShapingNone

Hardware Parameters

Phase Width32[3 - 48]

Output Width16[3 - 16]

Configuración NCO 1

OK Cancel

7



IP Symbol

Information

☒ Show disabled ports

+

S_AXIS_PHASE

+

S_AXIS_CONFIG

+

ack

+

ack_en

+

aresetn

+

M_AXIS_DATA

+

M_AXIS_PHASE

+

event_pinc_invalid

+

event_poff_invalid

+

event_phase_in_invalid

+

event_s_phase_tlast_missing

+

event_s_phase_tlast_unexpected

+

event_s_phase_chanid_incorrect

+

event_s_config_tlast_missing

+

event_s_config_tlast_unexpected

Component Name

dds_compiler_0

Configuration

Implementation

Detailed Implementation

Phase Angle Increment Values

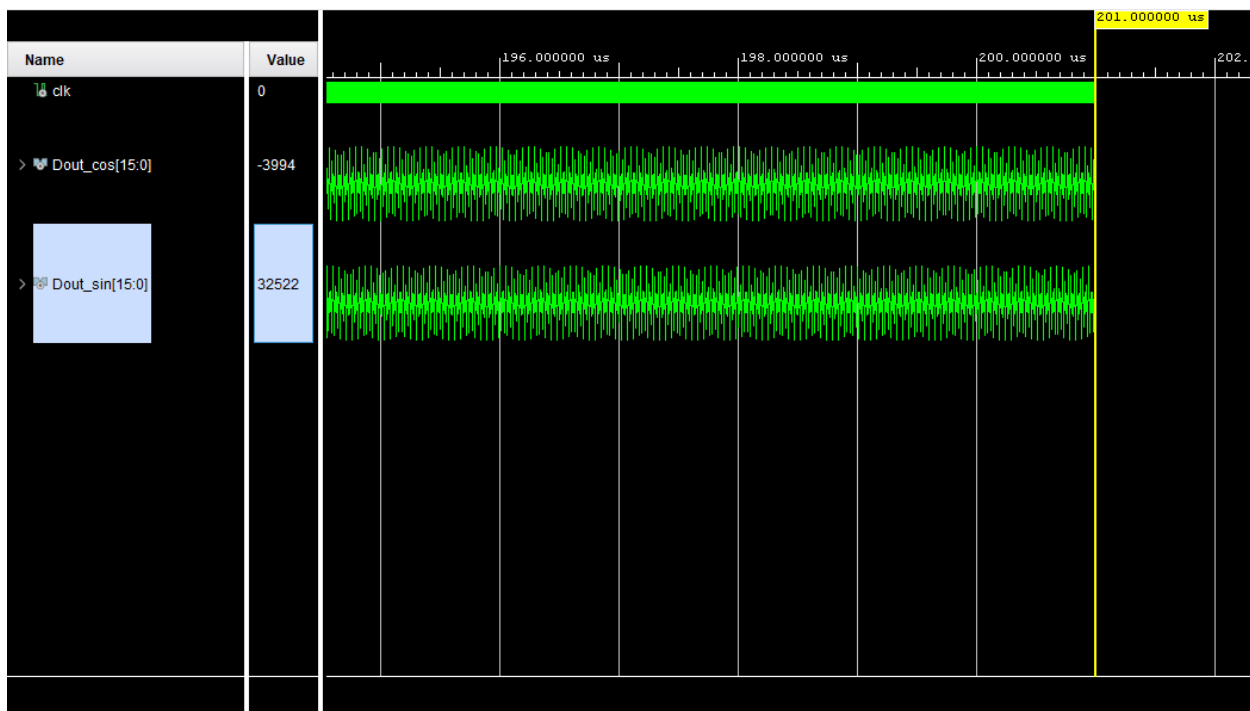
Summary

Additional Summary

OK

Cancel

$$F_{Out} = 40.7\text{MHz}$$

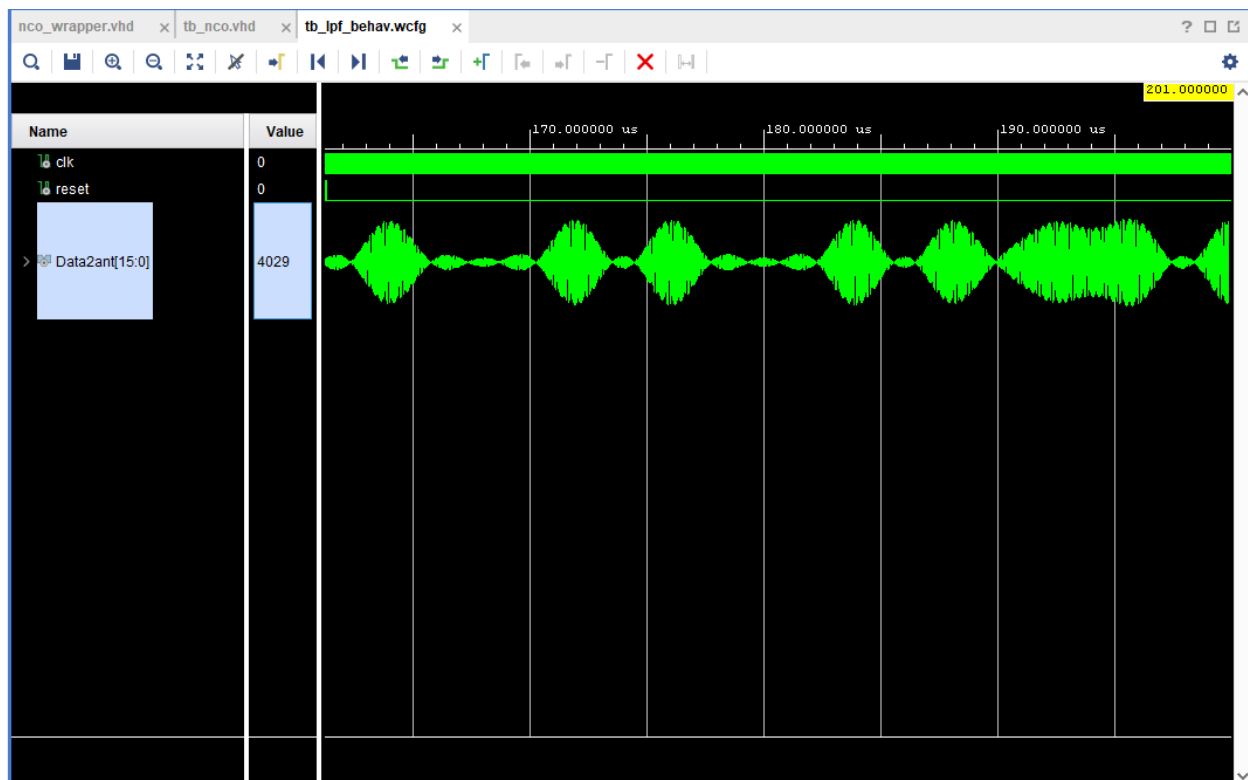


Simulación del bloque NCO

Modulador QPSK

Este bloque corresponde a la ejecución del `tb_modulador_qpsk`, que integra todos los bloques del modulador QPSK descritos a la práctica: el generador de símbolos, el bloque de interpolación (cero padding + filtro pasa-bajo), y el modulador Y/Q con NCO.

El objetivo de la simulación es verificar que la señal `Data2Ant`, generado en la salida del sistema, presenta una modulación QPSK coherente, con una portadora centrada a 40,7 MHz y con un ancho de banda ajustado según la interpolación y el filtraje aplicados.



Simulación del test bench del modulador

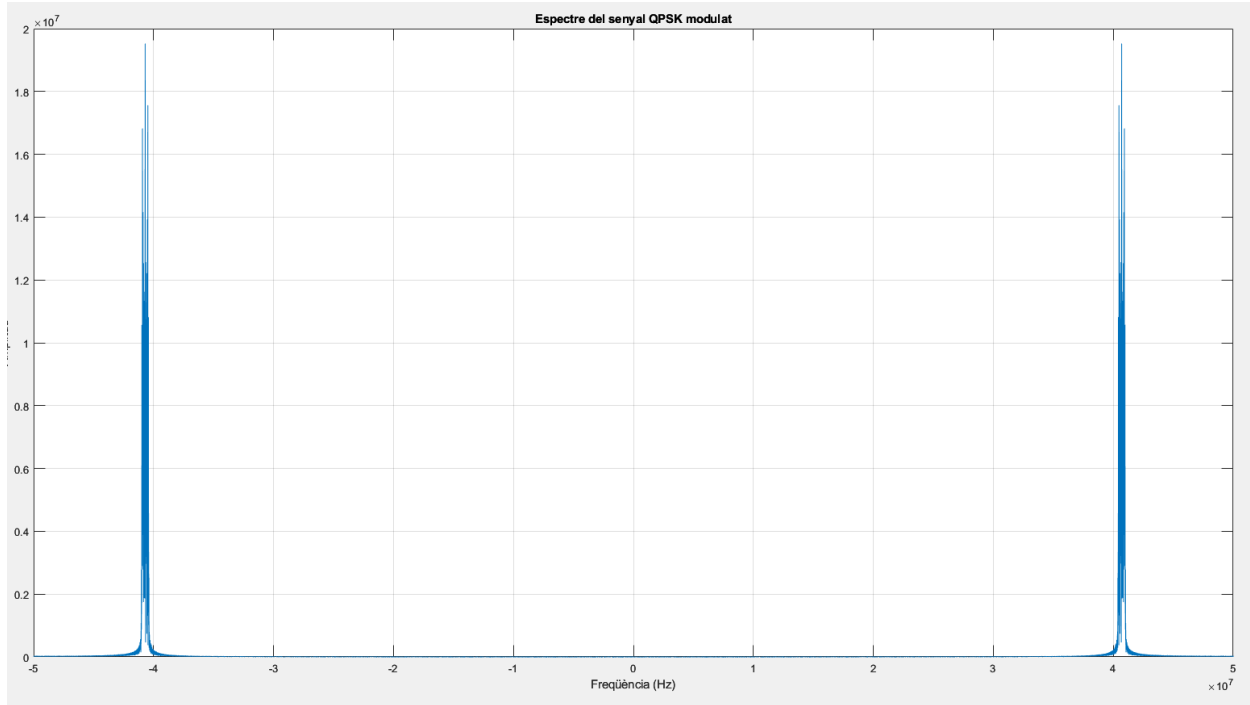
Tal como se muestra en la gráfica de simulación:

La señal `Data2Ant` muestra una forma de ola característica de una señal QPSK interpolado y filtrado, con variaciones de amplitud debidas a la combinación de símbolos Y/Q y a la modulación con la portadora.

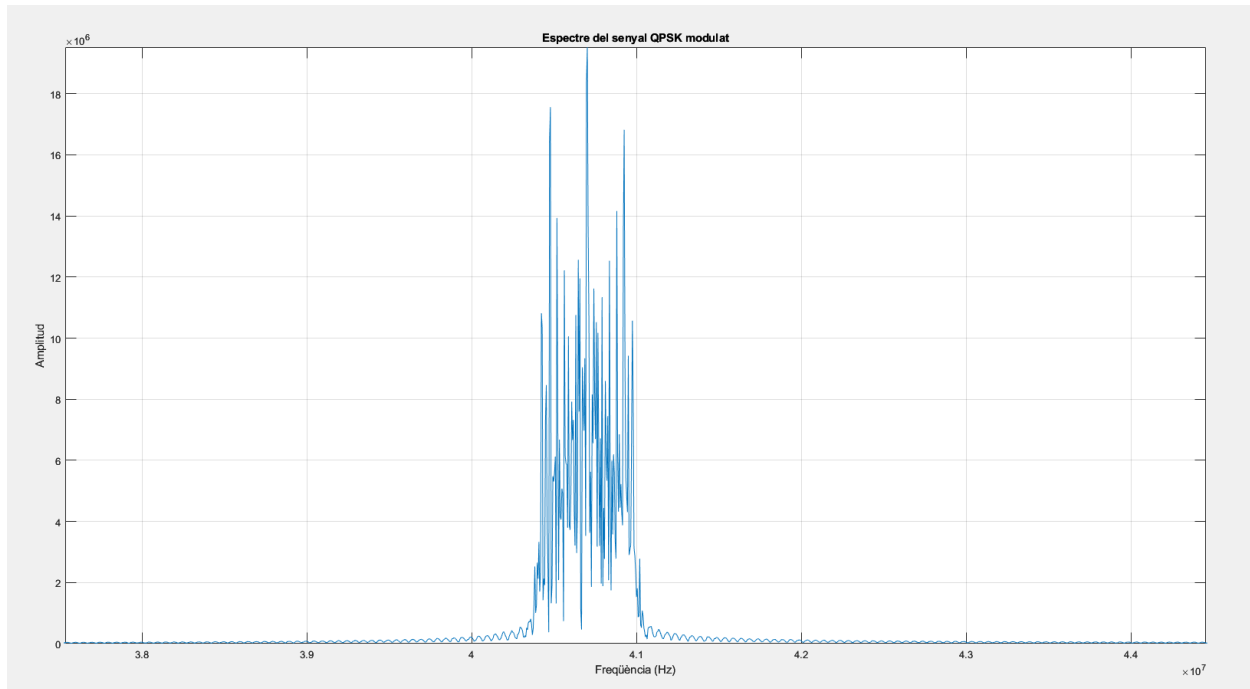
La señal es aleatoria pero simétrica, evidenciando que el generador QPSK funciona correctamente, creando símbolos pseudoaleatorios y mapejant-los con las fases

esperadas.

Las transiciones suaves de la señal indican que el filtro pasa-bajo ha sido aplicado correctamente, eliminando los espectros duplicados introducidos por el cero padding.



Espectre del senyal transmès amb una portadora de 40.7 MHz.



Espectre ampliat del senyal transmès amb una portadora de 40.7 MHz.

La senyal presenta un ritme clar de símbols cada 2 μ s (500 kSym/s), interpolados correctament a 100 MSps.

Se perciben agrupaciones periódicas de símbolos, lo cual es indicativo de un cero padding correcto (199 ceros entre símbolos).

Si configurásemos el NCO para transmitir a la banda de 6 m (50 MHz), ¿funcionaría correctamente el modulador QPSK que hemos implementado? ¿por qué?

Sí, el modulador QPSK seguiría funcionando correctamente si se configura el NCO para transmitir a una frecuencia de 50 MHz (dentro de la banda de 6 metros), siempre que se mantenga la frecuencia de muestreo del sistema en 100 MHz.

La clave está en que el DDS (Direct Digital Synthesizer) utilizado en el NCO permite generar señales de portadora a diferentes frecuencias dentro del rango permitido por el sistema, simplemente ajustando la palabra de control de fase (Phase Increment Word). En este caso, generar una portadora a 50 MHz sigue siendo compatible con una frecuencia de muestreo de 100 MHz, ya que cumple el criterio de Nyquist:

La frecuencia máxima que puede generarse sin aliasing es $f_s/2 = 50$ MHz.

Por tanto, 50 MHz es precisamente el límite superior de generación sin aliasing, lo cual significa que la señal QPSK podría ser generada justo en ese borde. Aunque funcionaría, es importante tener en cuenta que al operar tan cerca del límite de Nyquist, pueden producirse algunos efectos no deseados como:

Mayor sensibilidad al ruido y distorsión espectral, debido a la proximidad con los reflejos espectrales.

Requisitos más estrictos de filtrado, para evitar aliasing o productos de imagen en un entorno real con DAC y RF.

Conclusiones

A lo largo de esta práctica se ha diseñado, implementado y verificado un sistema completo de modulación QPSK utilizando VHDL. Esta experiencia ha permitido comprender en profundidad cómo se estructura un transmisor digital, desde la generación de símbolos hasta la obtención de una señal modulada lista para ser enviada a un conversor digital-analógico (DAC).

Cada uno de los bloques del sistema —generador de símbolos, interpolador mediante zero padding, filtro paso bajo y modulador I/Q— ha sido implementado siguiendo especificaciones precisas y verificado mediante test benches individuales. La correcta integración de todos los bloques se ha confirmado a través de la simulación final, donde la señal de salida (Data2Ant) presenta las características esperadas de una modulación QPSK interpolada y filtrada, con una portadora centrada a 40,7 MHz.

Además, el análisis espectral realizado con MATLAB ha corroborado la validez del diseño, evidenciando un espectro centrado y limitado en banda, como corresponde a un sistema bien interpolado y filtrado. También se ha razonado sobre la capacidad del sistema para operar en otras frecuencias dentro de la banda VHF, como 50 MHz, demostrando la flexibilidad del diseño cuando se respetan las condiciones del teorema de Nyquist.

En definitiva, esta práctica ha reforzado tanto los conocimientos teóricos como las habilidades prácticas en diseño digital aplicado a sistemas de comunicaciones, mostrando el potencial del lenguaje VHDL para implementar soluciones reales y eficientes en el ámbito del *software radio*.