

Laboratorio 3: De Radiofrecuencia a la Envolvente Compleja (GNURADIO)

Carlos Stiven Roa Martinez
Universidad Industrial de Santander
 Bucaramanga, Colombia
 Email: carlos2201948@correo.uis.edu.co

Dana Cotes Cala
Universidad Industrial de Santander
 Bucaramanga, Colombia
 Email: dana2212256@correo.uis.edu.co

Abstract—This laboratory focuses on the analysis and conversion of Radio Frequency (RF) signals into their Complex Envelope (CE) representation using GNU Radio. Digital modulations such as OOK, BPSK, and FSK were implemented in both RF and CE formats, and their characteristics were analyzed in the time, frequency, and constellation domains. The study provided insight into the advantages of baseband processing and the role of specialized blocks like VCOs for different modulation schemes.

Index Terms—Software-Defined Radio (SDR), GNU Radio, complex envelope, digital modulation, On-Off Keying (OOK), Binary Phase-Shift Keying (BPSK), Frequency-Shift Keying (FSK), time-domain analysis, frequency-domain analysis, constellation diagram.

Repositorio: https://github.com/DamaGit/com2_A1_G-9/tree/Pr%C3%A1ctica_3/Pr%C3%A1ctica_3

I. INTRODUCCIÓN

La Envolvente Compleja (EC) representa una herramienta fundamental en el procesamiento de señales de comunicación, permitiendo trabajar en banda base con representaciones más simples y eficientes computacionalmente. Este laboratorio explora la conversión de señales de RF a EC para modulaciones digitales básicas, analizando las diferencias en implementación y comportamiento entre ambos dominios.

II. OBJETIVOS

- Producir combinaciones de bloques de GNURadio para la aplicación específica de la envolvente compleja.
- Comprender los bloques, diagramas y su funcionalidad para la producción de otros sistemas.
- Representar las señales de salida tanto de RF como en EC en GNURadio.

III. METODOLOGÍA

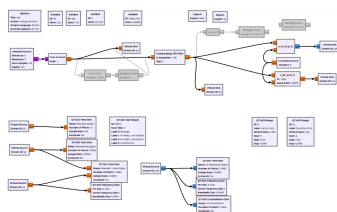


Fig. 1: Flujograma GNU Radio utilizado

Se utilizaron los flujogramas base RF_EC_ook.grc, RF_EC_bpsk.grc y RF_EC_fsk.grc para el análisis de las modulaciones, estos realmente modificaciones y ajustes minimos pra implementar cada una de las respectivas modulaciones, ante una misma entrada binaria, se realiza el analisis de cada modulacion:

A. Modulación On-Off Keying (OOK)

La (OOK) es una técnica de modulación digital basada en la variación de la **amplitud** de una señal portadora para representar información binaria. Es una forma simplificada de modulación por amplitud (ASK), donde la portadora se enciende o apaga según el bit transmitido tal como se aprecia en la figura 2.

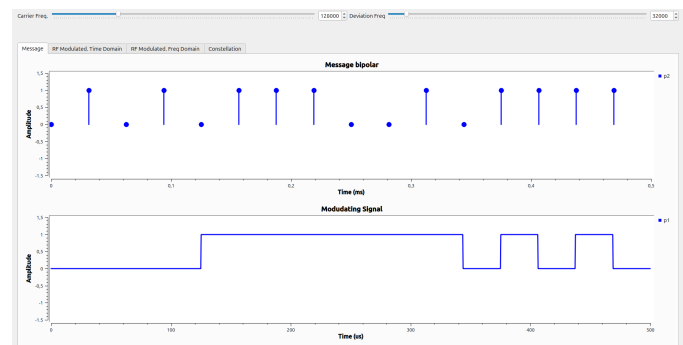


Fig. 2: Mensaje

En esta modulación, cada bit se representa de la siguiente manera:

- El bit **1** se transmite mediante una señal portadora de amplitud A .
- El bit **0** se representa con la ausencia de señal (amplitud nula).

Matemáticamente, puede expresarse como:

$$s(t) = \begin{cases} A \cos(2\pi f_c t), & \text{si el bit es 1,} \\ 0, & \text{si el bit es 0.} \end{cases}$$

La modulación OOK es simple de implementar y eficiente en términos de potencia, ya que no se transmite energía durante los bits '0'. Sin embargo, es más sensible al ruido en comparación con modulaciones que codifican información

en fase o frecuencia, sin embargo la robustez del sistema varía dependiendo de la frecuencia del carrier que se implemente.

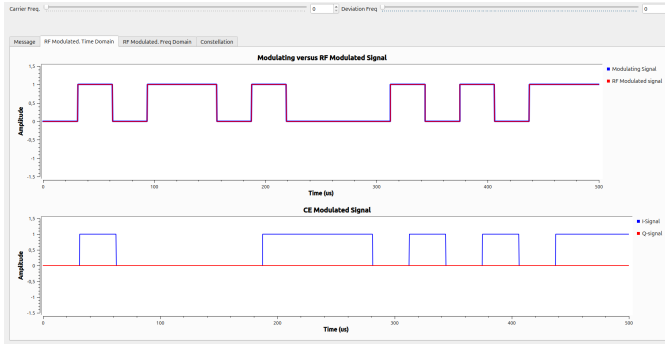


Fig. 3: OOK RF Modulated/Time Domain $F_c=0$

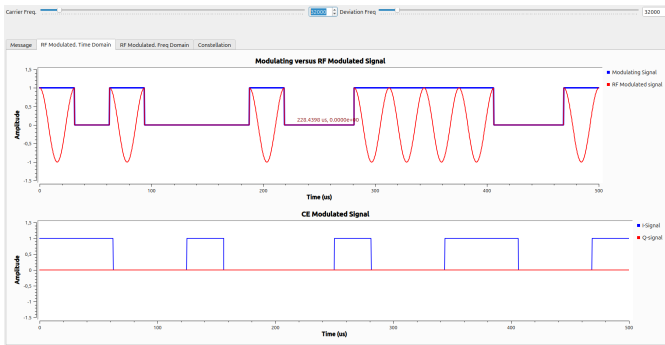


Fig. 4: OOK RF Modulated/Time Domain $F_c=32k$

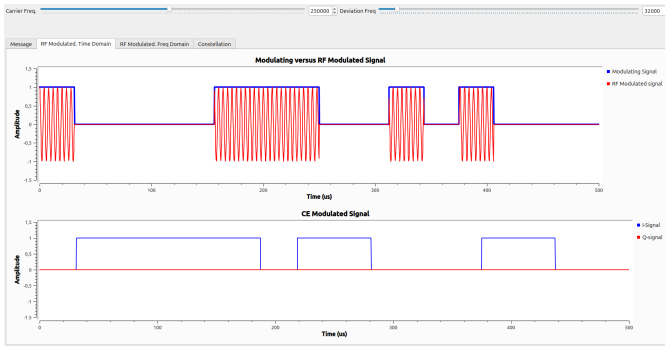


Fig. 5: OOK RF Modulated/Time Domain $F_c=250k$

Este tipo de modulación se utiliza en aplicaciones de bajo consumo y corto alcance, como controles remotos por infrarrojo, sistemas de comunicación óptica y algunos transmisores de radiofrecuencia digitales.

Por otra parte, al analizar el comportamiento de la modulación en el dominio de la frecuencia.

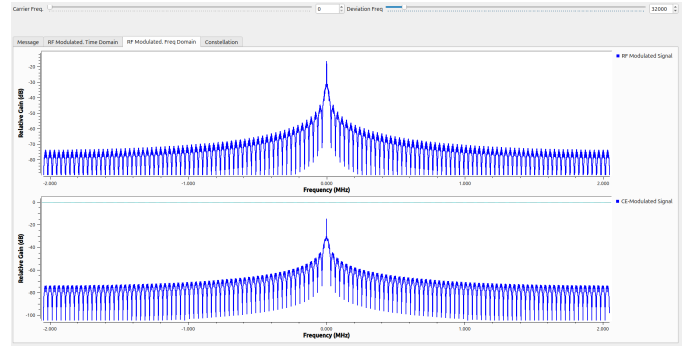


Fig. 6: OOK RF Modulated/Freq Domain $F_c=0$

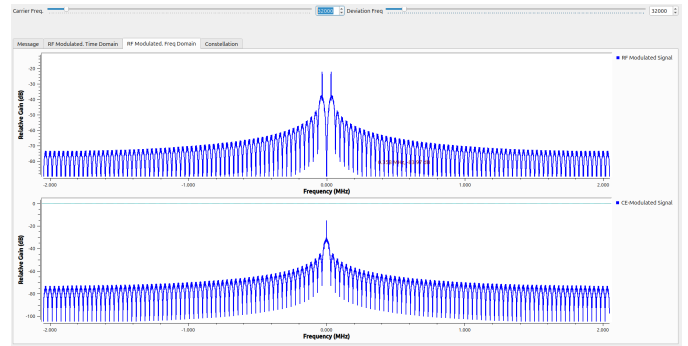


Fig. 7: OOK RF Modulated/Freq Domain $F_c=32k$

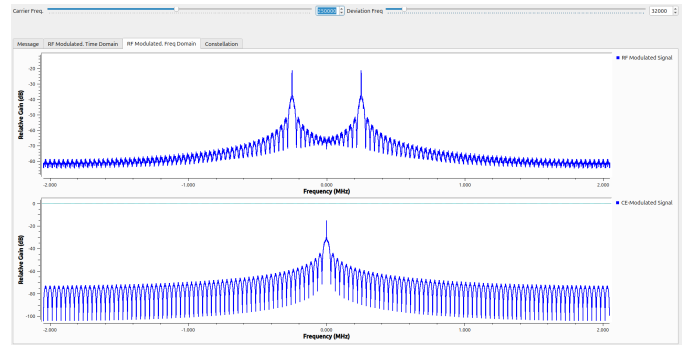


Fig. 8: OOK RF Modulated/Freq Domain $F_c=250k$

- **Lóbulo principal figura 6:** Centrado en la frecuencia portadora f_c con máxima amplitud.
- **Ancho de banda 7:** Aproximadamente $BW \approx \frac{R_b}{SPS} = \frac{32k}{4} = 8 \text{ kHz}$ por lóbulo.
- **Lóbulos laterales figura 9:** Distribuidos simétricamente alrededor de f_c , decayendo progresivamente.

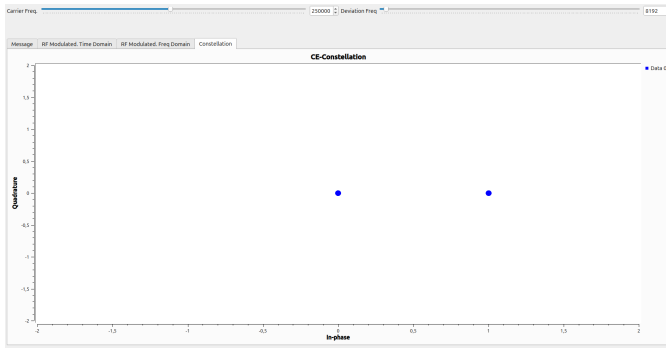


Fig. 9: OOK $F_c = 0/32/250$ k

La constelación de una señal OOK (On-Off Keying) presenta las siguientes características:

- **Bit '0'**: Representado por el punto $(0,0)$ en el plano complejo
- **Bit '1'**: Representado por el punto $(A,0)$ en el plano complejo
- **Dimensión**: Unidimensional (solo componente en fase I)

La constelación OOK confirma la naturaleza de modulación en amplitud pura, sin componente en cuadratura.

B. Modulación Binary Phase-Shift Keying (BPSK)

La modulación (BPSK) es una técnica de modulación digital en la que la información binaria se transmite modificando la **fase** de una señal portadora. En este caso, cada bit se representa mediante una de dos posibles fases, separadas por 180° .

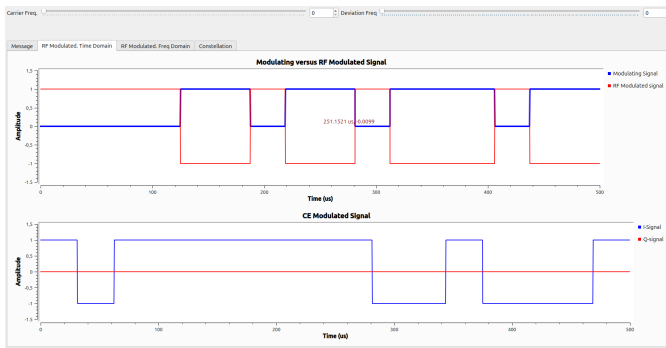


Fig. 10: BPSK $F_c = 0$ k

En esta modulación:

- El bit **1** se transmite con una fase de 0° .
- El bit **0** se transmite con una fase de 180° .

Matemáticamente, la señal puede expresarse como:

$$s(t) = A \cos(2\pi f_c t + \pi b(t)),$$

donde A es la amplitud, f_c la frecuencia de la portadora y $b(t)$ representa el bit transmitido (0 o 1).

La principal ventaja de BPSK es su **alta inmunidad al ruido** en comparación con otras modulaciones de amplitud. Debido a que sólo se utilizan dos fases bien definidas figura 11, la detección de los bits es más robusta ante interferencias y distorsiones del canal.

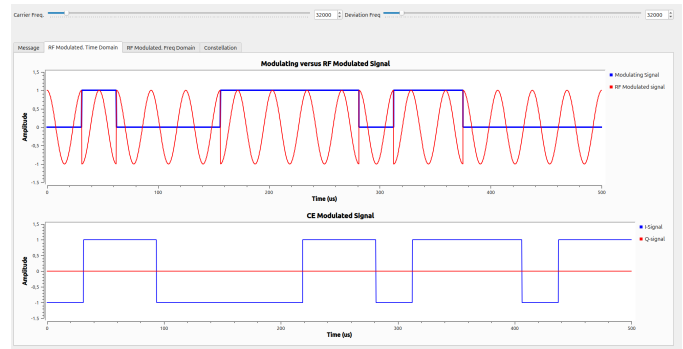


Fig. 11: BPSK $F_c = 32$ k

Sin embargo, transmite un solo bit por símbolo figura 12, por lo que su eficiencia espectral es menor que en modulaciones con más niveles de fase, como QPSK o 8-PSK.

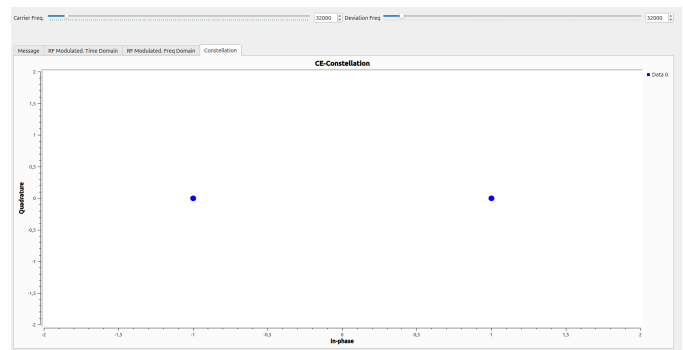


Fig. 12: BPSK constelación $F_c = 0/32$ k

La modulación BPSK presenta las siguientes características en el dominio de la frecuencia:

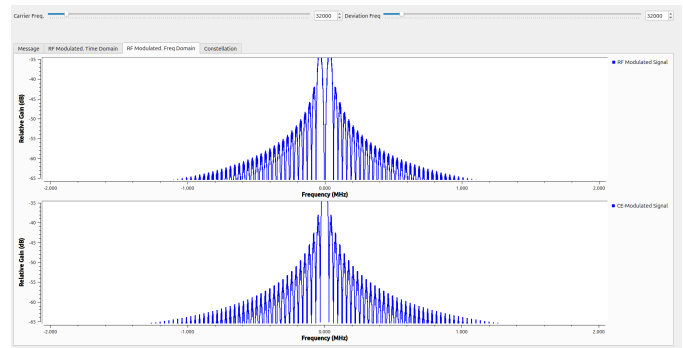


Fig. 13: BPSK RF Modulated/Freq Domain $f_c = 32$ k

adicionalmente en frecuencia para esta modulación se tiene:

- **Centro espectral**: 32 kHz (frecuencia de portadora)
- **Simetría perfecta**: Lóbulos idénticos alrededor de f_c
- **Forma sinc²**: Resultado de los pulsos rectangulares BPSK

La modulación BPSK es ampliamente utilizada en sistemas satelitales, enlaces de microondas, comunicaciones inalámbricas y sistemas de navegación, donde la fiabilidad de la transmisión es prioritaria sobre la velocidad de datos.

C. Modulación Frequency-Shift Keying (FSK)

La (FSK) es una técnica de modulación digital en la que la información binaria se transmite mediante variaciones en la **frecuencia** de una señal portadora.

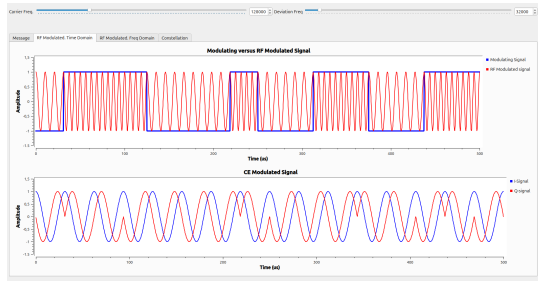


Fig. 14: BFSK time domain $f_c=128k$ Desviación $f_{eq}=32k$

En esta modulación, cada bit se asocia con una frecuencia diferente:

- El bit **1** se transmite con una frecuencia f_1 .
- El bit **0** se transmite con una frecuencia f_0 .

Matemáticamente, puede expresarse como:

$$s(t) = \begin{cases} A \cos(2\pi f_1 t), & \text{si el bit es 1,} \\ A \cos(2\pi f_0 t), & \text{si el bit es 0.} \end{cases}$$

Esta técnica tiene la ventaja de ser resistente al ruido y a la interferencia, ya que la información está codificada en la frecuencia, no en la amplitud ni en la fase.

Entre sus variantes se encuentran:

- **BFSK** (Binary FSK): usa dos frecuencias distintas, una para cada bit.
- **M-FSK**: emplea múltiples frecuencias para representar varios bits por símbolo, aumentando la eficiencia espectral.

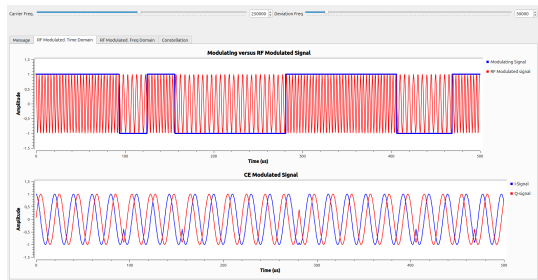


Fig. 15: BFSK time domain $f_c=250k$ Desviación $f_{eq}=50k$

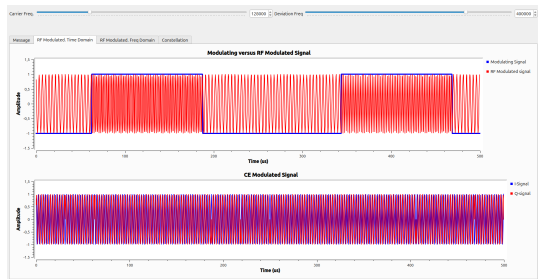


Fig. 16: BFSK time domain $f_c=128k$ Desviación $f_{eq}=400k$

La modulación BFSK presenta características espectrales únicas debido a la conmutación entre dos frecuencias portadoras.

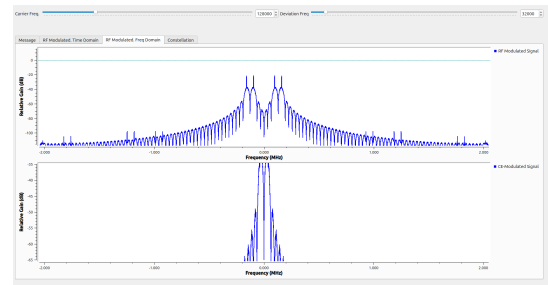


Fig. 17: BFSK time domain $f_c=128k$ Desviación $f_{eq}=32k$

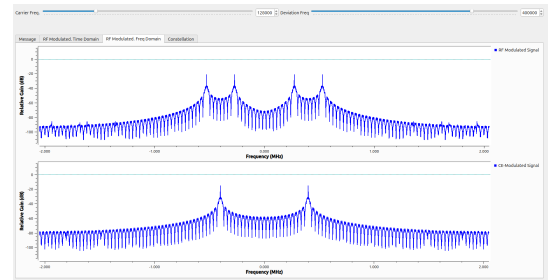


Fig. 18: BFSK time domain $f_c=128k$ Desviación $f_{eq}=400k$

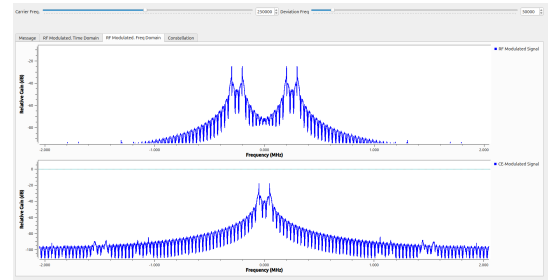


Fig. 19: BFSK time domain $f_c=250k$ Desviación $f_{eq}=50k$

- **Fig. 17:** $f_c = 128 \text{ kHz}$, $\Delta f = 32 \text{ kHz}$
Baja desviación en frecuencia → Espectro más compacto, superposición espectral significativa.
- **Fig. 18:** $f_c = 128 \text{ kHz}$, $\Delta f = 400 \text{ kHz}$
Alta desviación → Dos lóbulos espectrales separados, menor interferencia entre símbolos.
- **Fig. 19:** $f_c = 250 \text{ kHz}$, $\Delta f = 50 \text{ kHz}$
Frecuencia central mayor con desviación moderada → Espectro desplazado a mayor frecuencia, con ancho de banda intermedio.

aunque el análisis de forma de la modulación en el espectro se ve bastante bueno, al analizar el comportamiento de estas señales en sus respectivas constelaciones se evidencia una notable superposición de símbolos.

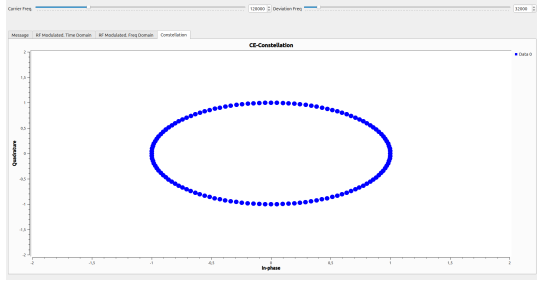


Fig. 20: BFSK Constellation $f_c=128k$ Desviation freq= 32k

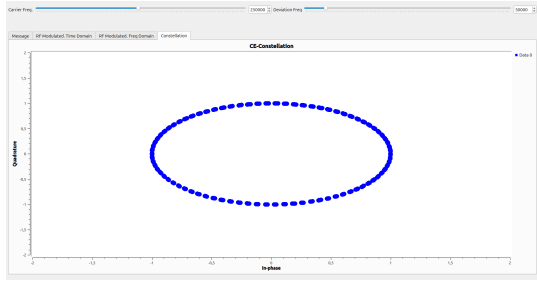


Fig. 21: BFSK Constellation $f_c=128k$ Desviation freq= 50k

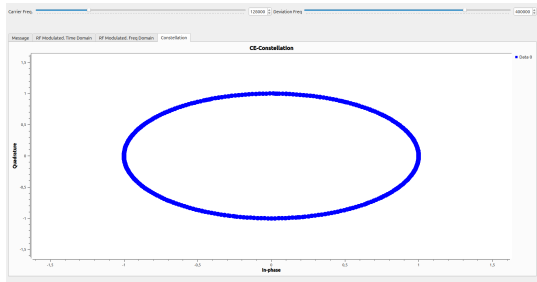


Fig. 22: BFSK Constellation $f_c=250k$ Desviation freq= 50k

La dispersión observada en las constelaciones BFSK se debe principalmente al **incumplimiento de la condición de ortogonalidad** entre las portadoras.

Para obtener puntos definidos en la constelación, debe satisfacerse:

$$\int_0^{T_b} \cos(2\pi f_1 t) \cos(2\pi f_2 t) dt = 0 \quad (1)$$

Esto se cumple cuando:

$$\Delta f = |f_2 - f_1| = \frac{k}{2T_b}, \quad k \in \mathbb{Z}^+ \quad (2)$$

- f_1, f_2 : Frecuencias de las señales cosenoidales.
- T_b : Duración del símbolo o bit.
- $\Delta f = |f_2 - f_1|$: Diferencia de frecuencias.
- $k \in \mathbb{Z}^+$: Un número entero positivo (1, 2, 3, ...).

En sistemas modernos, la FSK se utiliza ampliamente en comunicaciones por radio, módem y sistemas de identificación inalámbrica (RFID), donde la robustez frente a ruido es fundamental.

IV. EXPLICACIÓN DEL CÓDIGO

A. CE VCO or baseband VCO

```
1 from gnuradio import gr
2 import numpy as np
3
4 class blk(gr.sync_block):
5
6
7     def __init__(self,):
8         # Firma del bloque: 2 entradas float32, 1 salida
9         # complex64
10        super().__init__(
11            name='e_CE_VCO_fc',
12            in_sig=[np.float32, np.float32],
13            out_sig=[np.complex64]
14        )
15
16    def work(self, input_items, output_items):
17        # Entradas:
18        # A: amplitud por muestra (float32)
19        # Q: fase por muestra (float32, radianes, ya
20        # ACUMULADA)
21        A = input_items[0]
22        Q = input_items[1]
23
24        # Salida compleja (complex64)
25        y = output_items[0]
26
27        # N = nmero de muestras a procesar; A y Q deben
28        # tener la misma N
29        N = len(A)
30
31        # Ncleo: exponencial compleja con envolvente A[n]
32        # y[n] = A[n] * e^{j * Q[n]}
33        # Nota: np.exp(1j*Q) vectoriza y es estable para
34        # cualquier Q.
35        y[:] = A * np.exp(1j * Q)
36
37        # Devolver cuantas muestras se produjeron
38        return len(y)
```

Listing 1: CE VCO en banda base: implementación y comentarios

```
1 Name: CE VCO (e_CE_VCO_fc)
2 Type: sync (float, float complex)
3
4 Overview
5 -----
6 Generates a complex baseband carrier from per-sample
7 amplitude and phase:
8   y[n] = A[n] * exp(jQ[n]) = A[n] (cos Q[n] + j sin Q[n])
9
10 where:
11   - A[n]: instantaneous amplitude (linear scale),
12     higher values are possible but watch clipping.
13   - Q[n]: accumulated phase in radians,
14   - y[n]: complex output (complex64) with |y|=|A| and y=Q
15     (for A>=0).
16
17 Ports
18 -----
19 - In[0] A (float32, top): Per-sample amplitude. Use 0..1 as
20   normalized range;
21   higher values are possible but watch clipping.
22 - In[1] Q (float32, radians, bottom): Per-sample phase in
23   radians.
24   Must be an accumulated, continuous phase.
25 - Out[0] y (complex64): Complex baseband output (I + jQ).
26
27 Parameters
```

```

23 -----
24 (None.) The sample rate is defined by the flowgraph.
25 Frequency behavior depends
26 entirely on how you construct Q[n].
27
28 Usage Notes
29 -----
30 - No internal phase integration: the block does not
31   accumulate phase.
32   Phase continuity fully depends on Q[n].
33 - Pure tone at f0 (sample rate fs):
34    $Q[n] = Q[0] + 2 f_0 n / fs$ 
35 - PM/FM: integrate instantaneous frequency deviation
36   externally and feed the
37   accumulated phase into Q[n]:
38    $Q[n] = Q[n-1] + 2 f[n] / fs$ 
39 - Negative A[n] is equivalent to adding to the effective
40   phase.
41 - Optionally wrap Q to (-, ] for visualization (not
42   required).
43
44 Best Practices
45 -----
46 - Keep A[n] within the downstream dynamic range (e.g., 0..1)
47   to avoid clipping.
48 - If unit power is assumed, set A[n]=1 or scale for target
49   SNR.
50 - Ensure A and Q have identical vector lengths per scheduler
   call.

```

Listing 2: Block Help: CE VCO (Documentation for GNU Radio)

B. RF VCO (real-valued carrier)

```

1 from gnuradio import gr
2 import numpy as np
3 import math
4
5 class blk(gr.sync_block):
6
7     def __init__(self, fc=128000, samp_rate=320000):
8         # Firma del bloque: 2 entradas float32 (A, Q), 1
9         # salida float32 (y)
10        super().__init__(
11            name='e_RF_VCO_ff',
12            in_sig=[np.float32, np.float32],
13            out_sig=[np.float32]
14        )
15
16        # Parmetros del bloque
17        self.fc = float(fc) # Frecuencia de
18        portadora (Hz)
19        self.samp_rate = float(samp_rate) # Frecuencia de
20        muestreo fs (S/s)
21
22        # Estado: ndice de muestra acumulado (preserva
23        # continuidad de fase)
24        self.n_m = 0
25
26    def work(self, input_items, output_items):
27        # Entradas:
28        # A: amplitud por muestra (float32, escala lineal)
29        # Q: fase por muestra (float32, radianes, ya
30        # ACUMULADA/continua)
31        A = input_items[0]
32        Q = input_items[1]
33
34        # Salida real (float32)
35        y = output_items[0]
36
37        # N = nmero de muestras a procesar; A y Q deben
38        # tener la misma N

```

```

33 N = len(A)
34
35 # ndices absolutos de muestra: [n_m, n_m+1, ...,
36 # n_m+N-1]
37 # Usamos arange (paso entero exacto) para
38 # estabilidad numrica
39 n = np.arange(self.n_m, self.n_m + N,
40 dtype=np.float64)
41
42 # Actualizar ndice acumulado para preservar
43 # continuidad de fase
44 self.n_m += N
45
46 # ngulo (radianes): 2*fc*n/fs + Q[n]
47 theta = (2.0 * math.pi * self.fc / self.samp_rate) *
48 n + Q
49
50 # Ncleo: coseno real escalado por A[n]
51 # y[n] = A[n] * cos(2*fc*n/fs + Q[n])
52 y[:] = A * np.cos(theta)
53
54 # Devolver cuantas muestras se produjeron
55 return len(y)

```

Listing 3: RF VCO en banda pasante (float, float → float): implementación y comentarios

```

1 Name: RF VCO (e_RF_VCO_ff)
2 Type: sync (float, float float)
3
4 Overview
5 -----
6 Generates a real-valued RF carrier from per-sample amplitude
7 and phase:
8   y[n] = A[n] * cos(2fcn/fs + Q[n])
9
10 where:
11 - A[n]: instantaneous amplitude (linear scale),
12 - Q[n]: per-sample phase (radians, accumulated/continuous),
13 - fc: carrier frequency (Hz),
14 - fs: sampling rate (samples/s),
15 - y[n]: real output (float32).
16
17 Ports
18 -----
19 - In[0] A (float32, top): Per-sample amplitude. Use 0..1 as
20   normalized range;
21   higher values are possible but watch downstream clipping.
22 - In[1] Q (float32, radians, bottom): Per-sample phase in
23   radians (ACCUMULATED).
24   Use Q=0 for a plain cosine; vary Q to realize PM/FM.
25 - Out[0] y (float32): Real-valued output waveform.
26
27 Parameters
28 -----
29 - fc (Hz): Carrier (center) frequency. For alias-free
30   operation, keep fc < fs/2.
31 - samp_rate (S/s): Processing sample rate fs. Should match
32   the flowgraph rate.
33
34 Behavior and Notes
35 -----
36 - Phase continuity across scheduler calls is preserved via
37   an internal running
38   sample index; this avoids discontinuities between buffers.
39 - A and Q must have identical vector lengths per call.
40 - If you change fs at runtime, reset the internal phase
41   state (or re-create the
42   block) to avoid phase jumps.
43
44 Usage Tips
45 -----

```



```

39 - Keep |A[n]| 1 unless the downstream chain has sufficient
    headroom.
40 - Pure tone at f0 (sample rate fs): Q[n] = Q[0] + 2 f0 n /
    fs.
41 - For PM/FM: integrate instantaneous frequency deviation
    externally and feed the
42 accumulated phase into Q[n].

```

Listing 4: Block Help: RF VCO (Documentation for GNU Radio)

V. PREGUNTAS DE CONTROL

A. 1. Sobre el flujograma RF_EC_xxx.grc

- I. Elección de *SPS* suficientemente alto para versión paso-banda (RF). Sea f_c la portadora, R_s la tasa de símbolos y $f_s = \text{SPS} \cdot R_s$ la frecuencia de muestreo. Para evitar aliasing en RF se requiere

$$f_c + \frac{B}{2} < \frac{f_s}{2} \implies \text{SPS} > \frac{2f_c + B}{R_s}.$$

Cotas útiles:

- BPSK (coseno alzado, roll-off α): $B \approx (1 + \alpha)R_s$.
- 2-FSK (regla de Carson): $B \approx 2(\Delta f + R_b)$, con $R_b = R_s$ para 1 bit/símbolo.

Verificación práctica: (i) FFT sin pliegues cerca de $f_s/2$, (ii) oscilación de portadora claramente visible dentro de cada símbolo, (iii) EVM/BER estables al reducir *SPS* hacia el límite.

- II. BPSK: si “Multiply Const” = 1, equivale a quitarlo. Con ganancia 1 el bloque es identidad (no cambia amplitud ni signo). Es equivalente a cable directo si no se usa para habilitar/deshabilitar en tiempo real ni gestiona *tags*.
- III. FSK: deducción de la constante de “Multiply Const”. En 2-FSK, el símbolo $m[n] \in \{\pm 1\}$ modula la desviación Δf . Si $Q[n]$ es la fase acumulada que entra al VCO complejo,

$$Q[n] = Q[n-1] + 2\pi \frac{(f_c + m[n]\Delta f)}{f_s}.$$

Si f_c lo fija el VCO RF y solo se genera la fase de desviación,

$$Q[n] = Q[n-1] + 2\pi \frac{m[n]\Delta f}{f_s}.$$

Por tanto, el Multiply Const previo al integrador debe ser

$$2\pi \frac{\Delta f}{f_s}$$

de modo que $m[n] \times \frac{2\pi\Delta f}{f_s} \rightarrow \text{integrador} \rightarrow Q[n]$.

- IV. OOK: razón para configurar “Constant Source” = 0, y no así en BPSK/FSK. En OOK se conmuta amplitud: $A[n] \in \{0, A_0\}$, mientras la fase permanece constante ($Q[n] = 0$). En BPSK/FSK la información viaja en fase/frecuencia, con $A[n] \approx 1$.
- V. Entradas del VCO: por qué OOK usa la primera (A) y BPSK/FSK la segunda (Q). La entrada superior es amplitud *A* (OOK enciende/apaga portadora); la inferior es fase *Q* (BPSK rota fase, FSK integra frecuencia).

VI. BPSK: reubicar “Interpolating FIR Filter” inmediatamente antes del VCO. Es válido si el conformado (p. ej., RRC) y la interpolación se aplican al tren de símbolos antes de construir $Q[n]$. No es equivalente filtrar después sobre $Q[n]$ ya integrado o sobre la portadora.

VII. FSK: reubicar “Interpolating FIR Filter” inmediatamente antes del VCO. Es válido si el FIR actúa sobre la señal de control de frecuencia (o símbolos) antes de la multiplicación por $2\pi/f_s$ y del integrador que forma $Q[n]$. No se recomienda filtrar $Q[n]$ ya integrado.

B. 2. Otras preguntas de control

- a) VCO RF con entradas de amplitud y frecuencia (propuesta en GNU Radio). Diagrama funcional (RF real):

```

1 # Rama de amplitud
2 [A[n] (float)] --> [Limiter/Clip opcional] -----+
3                                                     |
4 # Rama de frecuencia                               v
5 [f_inst[n] (Hz)] -> [Add Const fc] -> [Multiply Const
6 (2/fs)]
7
8 Q[n] --+
9
10 |
11 (A,Q) -> [VCO RF (float,float->float)] -> y_RF[n] =
12 A[n]cos(Q[n])

```

Listing 5: Propuesta de VCO RF con entradas *A* y f_{inst}

Alternativa con EC y mezcla:

```

1 m[n]/control -> [...] -> Q[n]
2 A[n], Q[n] -> [VCO EC (float,float->complex)] -> y_EC[n]
3 y_EC[n] x cos(2 fc n/fs) -> RF real (o usar Frequency
4 Xlating Mixer)

```

- b) FSK: f_c máximo en RF sin aliasing con valores por defecto. Con $f_s = \text{SPS} R_s$ y ancho de banda por Carson $B \approx 2(\Delta f + R_b)$:

$$f_c^{\max} = \frac{f_s}{2} - \frac{B}{2} = \frac{\text{SPS} R_s}{2} - (\Delta f + R_b).$$

Conviene usar margen práctico del 10–20 % respecto al límite.

- c) FSK: desviación de frecuencia máxima para generar solo la EC. La EC ocupa aproximadamente $\pm(\Delta f + R_b)$. Para evitar aliasing en banda base:

$$\Delta f_{\max} \approx \frac{f_s}{2} - R_b = \frac{\text{SPS} R_s}{2} - R_b.$$

Se recomienda margen del 20–30 % por conformado y no idealidades.

- d) BPSK: *SPS* mínimo para versión RF. Con BPSK y roll-off α :

$$\text{SPS}_{\min} > \frac{2f_c}{R_s} + (1 + \alpha).$$

Criterio visual práctico: garantizar 8–10 ciclos de la portadora por símbolo,

$$\text{SPS} \gtrsim \frac{8f_c}{R_s} \text{ a } \frac{10f_c}{R_s}.$$

C. Apartado de preguntas (síntesis integradora)

El parámetro *SPS* es crucial para representar correctamente la RF y la EC; un *SPS* adecuado asegura ancho de banda suficiente, sincronía y SNR favorables. El bloque Multiply Const controla magnitudes como $2\pi\Delta f/f_s$ en FSK; en BPSK con valor 1 puede ser prescindible. La ubicación de bloques afecta el desempeño: en OOK la modulante entra por *A* (entrada superior); en BPSK/FSK por *Q* (entrada inferior). Reubicar el *Interpolating FIR* es viable si actúa antes de construir $Q[n]$ (BPSK: sobre símbolos; FSK: sobre control de frecuencia antes de integrar). Los límites de f_c y Δf se determinan por Nyquist/Carson y deben considerar márgenes de diseño.

VI. CONCLUSIONES

- 1) Se logró comprender la relación entre las señales de Radiofrecuencia (RF) y su representación en Envolvente Compleja (EC), evidenciando cómo esta conversión facilita el procesamiento digital al trabajar directamente en banda base, reduciendo la carga computacional sin pérdida de información relevante.
- 2) La implementación práctica de las modulaciones digitales OOK, BPSK y FSK permitió observar diferencias fundamentales entre variaciones en amplitud, fase y frecuencia, destacando la robustez de BPSK frente al ruido y la mayor eficiencia espectral de FSK en comparación con OOK.
- 3) Se concluye que el **cumplimiento estricto de la ecuación de ortogonalidad** es esencial para obtener constelaciones BFSK óptimas y garantizar un desempeño confiable del sistema de comunicación. Este descuido en el cálculo inicial resultó en:
 - Constelaciones con dispersión significativa
 - Solapamiento espectral entre símbolos
 - Mayor probabilidad de error en la demodulación

REFERENCES

- [1] H. Ortega, B. Bodá, O. Mauricio, and R. Torres, "Comunicaciones digitales basadas en radio definida por software," [Online]. Available: <https://sites.google.com/saber.uis.edu.co/comdig>.