

G-DSP Engine

Fase 0: Definición del Sistema y Análisis de Punto Fijo

Trabajo de Fin de Grado — Módem 16-QAM en FPGA

G-DSP Team

8 de febrero de 2026

Índice

1. Introducción	2
2. Plataforma Hardware	2
2.1. Sipeed Tang Nano 9K	2
2.2. Arquitectura DSP	2
3. Parámetros del Sistema	2
3.1. Cálculo de la Tasa de Símbolo	3
3.2. Ancho de Banda Ocupado	3
4. Formato de Punto Fijo	3
4.1. Notación $Qm.n$	3
4.2. Selección: Q1.11 (12 bits)	3
4.3. Propiedades del Formato Q1.11	4
5. Análisis de Precisión	4
5.1. Rango Dinámico	4
5.2. Relación Señal a Ruido de Cuantización (SQNR)	4
5.3. Margen sobre el Requisito de 16-QAM	4
6. Justificación de 12 Bits	5
7. Niveles de Constelación 16-QAM	5
8. Presupuesto de Errores de Cuantización	6
9. Presupuesto de Recursos DSP	6
10. Modelo de Referencia (Golden Model)	6
10.1. Funciones del Golden Model	6
10.2. Normalización de Coeficientes	7
11. Conclusiones de Fase 0	7
12. Referencias	7

1. Introducción

Este documento establece los fundamentos del proyecto G-DSP Engine, un procesador de banda base 16-QAM implementado íntegramente en lógica digital sobre la FPGA Gowin GW1NR-LV9. Se definen los parámetros del sistema, el formato aritmético de punto fijo y las restricciones de recursos que guían todas las decisiones de diseño en fases posteriores.

La **Fase 0** comprende:

- Selección de la plataforma hardware y sus limitaciones
- Definición de parámetros del sistema (frecuencias, tasas de símbolo)
- Análisis y selección del formato de punto fijo
- Presupuesto de recursos DSP
- Establecimiento del modelo de referencia en Python (Golden Model)

2. Plataforma Hardware

2.1. Sipeed Tang Nano 9K

El proyecto se desarrolla sobre la tarjeta de desarrollo **Sipeed Tang Nano 9K**, que incorpora la FPGA Gowin GW1NR-LV9QN88PC6/I5. Esta plataforma fue seleccionada por su equilibrio entre prestaciones, costo y disponibilidad de interfaces multimedia (HDMI nativo).

Cuadro 1: Recursos de la FPGA GW1NR-LV9

Recurso	Cantidad	Notas
LUTs (4-input)	8,640	Lógica combinacional
Flip-Flops	6,480	Registros
DSP Slices (MULT9)	20	Multiplicadores 9×9
BSRAM (18 Kbit)	26	Block RAM
PSRAM	64 Mbit	HyperRAM externa (no utilizada)
PLL	2	Síntesis de reloj

2.2. Arquitectura DSP

Los bloques DSP del GW1NR-LV9 son de tipo **MULT9/pREG**, que proporcionan:

- Un multiplicador **9×9 firmado** nativo
- Cascada para formar **18×18** (2 slices) o **36×36** (4 slices)
- Registro de pipeline integrado (pREG)

Para datos de 12 bits, utilizamos la configuración 18×18, consumiendo 2 slices MULT9 por multiplicador completo.

3. Parámetros del Sistema

Los parámetros fundamentales del sistema están centralizados en el paquete `gdsp_pkg.sv`, garantizando consistencia entre todos los módulos RTL.

Cuadro 2: Parámetros del sistema definidos en `gdsp_pkg.sv`

Parámetro	Símbolo	Valor	Unidad
Frecuencia de reloj	f_{clk}	27	MHz
Reloj de píxel (HDMI)	f_{pix}	74.25	MHz
Muestras por símbolo	SPS	4	—
Tasa de muestreo	$f_s = f_{clk}$	27	Msps
Tasa de símbolo	$f_{sym} = f_s/SPS$	6.75	MBaud
Bits por símbolo	—	4	bits
Tasa de bits	R_b	27	Mbps
Factor de roll-off	α	0.25	—
Taps del filtro RRC	N_{taps}	33	—

3.1. Cálculo de la Tasa de Símbolo

Con un reloj de sistema de 27 MHz y 4 muestras por símbolo:

$$f_{sym} = \frac{f_{clk}}{SPS} = \frac{27 \times 10^6}{4} = 6,75 \text{ MBaud} \quad (1)$$

Para modulación 16-QAM ($\log_2(16) = 4$ bits/símbolo):

$$R_b = f_{sym} \times \log_2(M) = 6,75 \times 4 = 27 \text{ Mbps} \quad (2)$$

3.2. Ancho de Banda Ocupado

El ancho de banda del filtro RRC con factor de roll-off $\alpha = 0,25$:

$$BW = (1 + \alpha) \times f_{sym} = 1,25 \times 6,75 = 8,4375 \text{ MHz} \quad (3)$$

Este ancho de banda cumple holgadamente con la frecuencia de Nyquist ($f_s/2 = 13,5$ MHz).

4. Formato de Punto Fijo

4.1. Notación $Qm.n$

Adoptamos la notación $Qm.n$ donde:

- m = número de bits de parte entera (excluyendo signo)
- n = número de bits de parte fraccionaria
- Bit de signo implícito en complemento a dos
- Ancho total: $W = 1 + m + n$ bits

4.2. Selección: Q1.11 (12 bits)

Tras evaluar múltiples opciones, se selecciona el formato **Q1.11**:

Cuadro 3: Comparativa de formatos de punto fijo candidatos

Formato	Bits	Rango	Resolución (LSB)	SQNR
Q1.7	8	$[-1, +0,992]$	$7,81 \times 10^{-3}$	~44 dB
green!15 Q1.11	12	$[-1, +0,9995]$	$4,88 \times 10^{-4}$	~68 dB
Q1.15	16	$[-1, +0,99997]$	$3,05 \times 10^{-5}$	~92 dB
Q2.10	12	$[-2, +1,999]$	$9,77 \times 10^{-4}$	~62 dB

4.3. Propiedades del Formato Q1.11

$$\text{Rango: } V \in [-1,0, +1,0 - 2^{-11}] \approx [-1,0, +0,99951] \quad (4)$$

$$\text{Resolución (1 LSB): } \Delta = 2^{-11} \approx 4,88 \times 10^{-4} \quad (5)$$

$$\text{Conversión a entero: } x_{int} = \text{round}(x_{float} \times 2^{11}) \quad (6)$$

$$\text{Conversión a flotante: } x_{float} = \frac{x_{int}}{2^{11}} \quad (7)$$

5. Análisis de Precisión

5.1. Rango Dinámico

El rango dinámico de un formato de punto fijo de N bits se calcula como:

$$DR = 20 \log_{10} \left(\frac{V_{max}}{V_{min}} \right) = 20 \log_{10} \left(\frac{2^{N-1} - 1}{1} \right) \quad (8)$$

Para Q1.11 ($N = 12$):

$$DR = 20 \log_{10}(2047) \approx 66,2 \text{ dB} \quad (9)$$

5.2. Relación Señal a Ruido de Cuantización (SQNR)

Para una señal sinusoidal de amplitud completa, la SQNR teórica es:

$$SQNR = 6,02N + 1,76 \text{ dB} \quad (10)$$

Donde N es el número de bits. Para Q1.11:

$$SQNR = 6,02 \times 12 + 1,76 = 72,24 + 1,76 = 74,0 \text{ dB (teórico)} \quad (11)$$

En la práctica, para señales QAM cuya distribución no es sinusoidal pura, se obtiene aproximadamente:

$$SQNR_{QAM} \approx 68 \text{ dB} \quad (12)$$

5.3. Margen sobre el Requisito de 16-QAM

Para 16-QAM con $BER = 10^{-5}$, el SNR mínimo requerido es aproximadamente **20 dB**. Con 68 dB de SQNR:

$$\text{Margen} = SQNR - SNR_{req} = 68 - 20 = 48 \text{ dB} \quad (13)$$

Este margen garantiza que el ruido de cuantización será completamente despreciable frente al ruido térmico del canal.

6. Justificación de 12 Bits

La selección de 12 bits (Q1.11) responde a un equilibrio óptimo:

1. **Compatibilidad DSP:** Los multiplicadores 18×18 de Gowin aceptan nativamente operandos de hasta 18 bits. Con 12 bits:

$$\text{Producto} = 12 + 12 = 24 \text{ bits} \quad (14)$$

El multiplicador produce 36 bits, dejando 12 bits de margen para acumulación directa dentro del DSP.

2. **Ancho del acumulador FIR:** Para 33 taps:

$$W_{\text{accum}} = W_{\text{prod}} + \lceil \log_2(N_{\text{taps}}) \rceil = 24 + 6 = 30 \text{ bits} \quad (15)$$

3. **Eficiencia de recursos:** 12 bits vs 16 bits reduce el consumo de registros y rutas de interconexión en $\sim 25\%$, crítico en una FPGA de 9K LUTs.

4. **Margen SQNR:** 48 dB sobre el requisito asegura que la cuantización no degradará el rendimiento del módem.

7. Niveles de Constelación 16-QAM

La constelación 16-QAM normalizada utiliza niveles $\{\pm 1, \pm 3\}/\sqrt{10}$ en cada eje, garantizando potencia media unitaria.

Cuadro 4: Niveles 16-QAM en Q1.11

Nivel	Valor Flotante	Cálculo	Q1.11 (decimal)
$-3/\sqrt{10}$	-0,948683	round($-0,948683 \times 2048$)	-1943
$-1/\sqrt{10}$	-0,316228	round($-0,316228 \times 2048$)	-648
$+1/\sqrt{10}$	+0,316228	round($+0,316228 \times 2048$)	+648
$+3/\sqrt{10}$	+0,948683	round($+0,948683 \times 2048$)	+1943

Estos valores están parametrizados en `gdsp_pkg.sv`:

```

1 parameter signed [DATA_WIDTH-1:0] QAM_NEG3 = -12'sd1943;
2 parameter signed [DATA_WIDTH-1:0] QAM_NEG1 = -12'sd648;
3 parameter signed [DATA_WIDTH-1:0] QAM_POS1 = 12'sd648;
4 parameter signed [DATA_WIDTH-1:0] QAM_POS3 = 12'sd1943;
```

Listing 1: Niveles QAM en gdsp_pkg.sv

8. Presupuesto de Errores de Cuantización

Cuadro 5: Error de truncamiento por etapa del módem

Etapa	Entrada	Salida	Error (LSB)
QAM Mapper	LUT exacta	12 bits	0 (exacto)
RRC Tx Filter	12 bits	12 bits	~0.5 (redondeo)
Suma AWGN	12 + 12 bits	12 bits	~0.5
RRC Rx Filter	12 bits	12 bits	~0.5
Gardner Interp.	12 bits	12 bits	~0.5
Costas Derotator	12 bits	12 bits	~0.5
Total acum.	—	—	~3 LSB peor caso

El error acumulado de 3 LSB en Q1.11 representa:

$$\varepsilon_{total} = 3 \times 2^{-11} \approx 1,46 \times 10^{-3} \quad (16)$$

Este valor es despreciable comparado con el piso de ruido del canal AWGN en cualquier escenario práctico de interés.

9. Presupuesto de Recursos DSP

Los 20 slices MULT9 disponibles se distribuyen según:

Cuadro 6: Asignación de bloques DSP

Bloque	DSP Slices	Notas
RRC Tx (semiplegado)	2–4	Multiplexado temporal
RRC Rx (semiplegado)	2–4	Misma arquitectura
Gardner interpolador	2	Interpolación lineal
Costas Loop	4	Mult. complejo (3 reales)
AWGN / misc	2–4	Escalado de ruido
Reserva	2–4	Cierre temporal / debug
TOTAL	≤ 20	✓ Cabe

10. Modelo de Referencia (Golden Model)

Se desarrolló un modelo de referencia bit-exacto en Python (`scripts/golden_model.py`) que implementa toda la cadena de señal con aritmética idéntica al RTL.

10.1. Funciones del Golden Model

- Generación de coeficientes RRC normalizados para Q1.11
- Mapeo 16-QAM con codificación Gray
- Filtrado RRC (convolución) con truncamiento idéntico al hardware
- Generación de vectores de test en formato `.hex` y `.mem`
- Plots de constelación, diagrama de ojo y espectro

10.2. Normalización de Coeficientes

Los coeficientes del filtro RRC se normalizan para evitar desbordamiento:

$$h_{norm}[n] = \frac{h[n]}{\max |h[n]|} \times (2^{FRAC_BITS-1} - 1) \quad (17)$$

Esto garantiza que el tap de mayor magnitud sea ligeramente menor que 0.5 en representación Q1.11, previniendo overflow durante la acumulación del FIR.

11. Conclusiones de Fase 0

Esta fase estableció los cimientos del proyecto:

1. **Plataforma validada:** Tang Nano 9K con 20 DSP slices y 8.6K LUTs es suficiente para un módem 16-QAM completo.
2. **Formato Q1.11:** Proporciona 68 dB de SQNR, 48 dB de margen sobre el requisito de 16-QAM, con eficiencia de recursos óptima.
3. **Parámetros del sistema:** 27 MHz de reloj, 6.75 MBaud, 27 Mbps, $\alpha = 0,25$, 33 taps RRC.
4. **Golden Model:** Referencia bit-exacta en Python para verificación rigurosa del RTL en todas las fases subsiguientes.

12. Referencias

1. Gowin Semiconductor, *GW1NR Series FPGA Products Datasheet*, DS861, 2023.
2. R. G. Lyons, *Understanding Digital Signal Processing*, 3rd ed., Prentice Hall, 2011, Ch. 12.
3. J. G. Proakis and M. Salehi, *Digital Communications*, 5th ed., McGraw-Hill, 2008, Ch. 5.
4. Xilinx, *Efficient Shift Registers, LFSR Counters, and Long Pseudo-Random Sequence Generators*, XAPP 052, 1996.