

# INFORME DE LABORATORIO: COMUNICACIÓN INALÁMBRICA CON MÓDULOS NRF24L01

Ramirez Velásquez, Tomas Alejandro.

@tomasramirez20 (GitHub)

<https://github.com/tomasramirez20/Practica-8-Com.Digitales>

[Est.tomas.ramirez@unimilitar.edu.co](mailto:Est.tomas.ramirez@unimilitar.edu.co)

UMNG

## 1. Introducción

Se llevó a cabo la implementación de un enlace inalámbrico punto a punto utilizando módulos transceptores **nRF24L01+** y microcontroladores **Raspberry Pi Pico W**. El objetivo principal fue establecer una comunicación confiable para el control remoto de un servomotor, visualizando en tiempo real el ángulo enviado tanto en el transmisor como en el receptor mediante pantallas LCD. Adicionalmente, se realizaron análisis del espectro radioeléctrico en la banda de 2.4 GHz, se estudió el protocolo SPI mediante un analizador lógico, y se evaluó el alcance del enlace en exteriores bajo diferentes configuraciones de potencia y tasa de datos. Finalmente, se contrastaron los resultados experimentales con un modelo teórico de presupuesto de enlace.

## 2. Marco Teórico

### Módulo nRF24L01+

Es un transceptor de bajo consumo que opera en la banda ISM de 2.4 GHz. Soporta tasas de datos de 250 kbps, 1 Mbps y 2 Mbps, y permite configurar la potencia de salida entre -18 dBm y 0 dBm. La comunicación con el microcontrolador se realiza mediante el bus SPI.

### Protocolo SPI

Interfaz síncrona full-duplex utilizada para la configuración y control del módulo nRF24L01. Las señales principales son:

- **SCK**: Reloj
- **MOSI**: Datos del maestro al esclavo
- **MISO**: Datos del esclavo al maestro

- **CSN**: Selección de esclavo
- **IRQ**: Interrupción

### 1) 2.3. Presupuesto de Enlace

Modelo utilizado para estimar la potencia recibida en función de la distancia, ganancias de antena y pérdidas en el medio. La ecuación básica es:

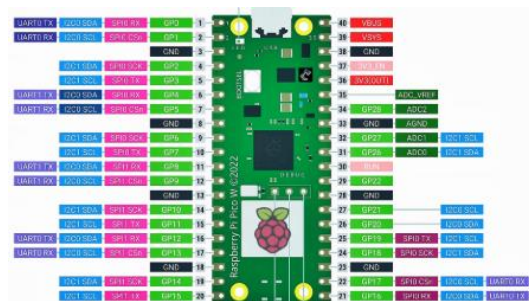
$$L(dB) = 32,44 + 20 \log_{10}(fMHz) + 20 \log_{10}(dKm)$$

## 3. Procedimiento y Montaje

### 3.1. Diagrama Esquemático de Conexiones

Aquí tienes el contenido reescrito y formateado para Word, con las tablas estructuradas y el texto organizado:

#### Raspberry Pi Pico 2



- **Power**: Pines de alimentación, como VBUS (5 V) y 3V3(OUT), utilizados para suministrar energía a periféricos y sensores.
- **Ground (GND)**: Pines de referencia eléctrica comunes para todos los circuitos.
- **GPIO (General Purpose Input/Output)**: Pines configurables para entrada o salida digital, así como para control PWM.
- **UART**: Pines dedicados a comunicación serial (TX y RX).
- **I2C**: Pines para comunicación mediante el protocolo I<sup>2</sup>C, indicados como SDA (datos) y SCL (reloj).
- **SPI**: Pines usados para el protocolo SPI (SCK, MOSI, MISO y CSn).
- **ADC**: Entradas analógicas (GP26, GP27, GP28) que permiten leer valores de sensores analógicos.
- **Debug / System control**: Pines especiales para control o depuración.

#### nRF24L01

#### Conexiones:

- VCC → 3V3(OUT) pin 36
- GND → GND pin 38
- CE → GP14
- CSN → GP5 (el GP15 se usará para el servo)
- SCK → GP6
- MOSI → GP7
- MISO → GP4

#### Descripción de pines SPI:

- **SCK (Serial Clock):** La Pico genera las pulsaciones de reloj que sincronizan la transferencia de bits.
- **MOSI (Master Out Slave In):** Línea por la que el Pico envía datos al nRF24L01.
- **MISO (Master In Slave Out):** Línea por la que el nRF24L01 devuelve datos al Pico.
- **CSN (Chip Select Not):** Activa el módulo nRF24L01 cuando el Pico desea comunicarse.
- **CE (Chip Enable):** Controla si el módulo está en modo transmisión o recepción.

#### OLED SSD1306

Tabla 1: Conexiones del OLED SSD1306

Pin OLED SSD1306	Pico pin físico	Pico GPIO	Descripción
GND	38	GND	Tierra
VCC	36	3V3 (OUT)	Alimentación
SCL	15	GP11	I2C1 SCL
SDA	14	GP10	I2C1 SDA

Dirección I2C: 0x3C (ya configurada en el código)

#### Joystick TX (emisora)

Tabla 2: Conexiones del Joystick

Pin del joystick	Función	Conexión en el Pico
VCC	Alimentación (5 V)	→ VBUS del Pico (pin físico 40)

GND	Tierra	→ GND del Pico (por ejemplo, pin 38)
VRx	Eje X (señal analógica)	→ GP26 (pin físico 31)
VRy	Eje Y (opcional por ahora)	→ GP27 (pin físico 32)
SW	Botón (opcional)	→ GP22 (pin físico 29) con Pin.IN, Pin.PULL_UP

#### Servo - RX (receptora)

##### Conexiones:

- Señal servo → GP15 - amarillo
- +5V servo → VBUS (5V del USB) pin 40 - rojo
- GND servo → GND del Pico - negro

**Importante:** NO alimentar el servo desde 3V3. Usar el VBUS (5V del USB) y masa común con el Pico.

#### MPU6050 (acelerómetro + giroscopio)

Tabla 3: Conexiones del MPU6050

Pin MPU 6050	Pico pin físico	Pico GPIO	Descripción
VCC	36	3V3 (OUT)	Alimentación 3.3 V (⚠ NO 5 V)
GND	38	GND	Tierra común
SCL	22	GP17	Mismo bus I2C1 que la OLED
SDA	21	GP16	Mismo bus I2C1 que la OLED
AD0	—	—	Conéctala a GND → dirección 0x68 (si lo pones a 3V3, será 0x69)
INT	—	—	Dejar sin conectar (no se usa)

XDA / XCL	—	—	Dejar sin conectar (solo sirven para sensores externos)
-----------------	---	---	---

## Implementación del Código

Una vez realizadas las conexiones procedemos a implementar el código para realizar la comunicación y funcionamiento de los elementos. En el repositorio se encontrarán, nombrados como Rx final y TxT, los dos códigos resultantes con mejor funcionalidad, después de probar con muchas variables de potencia, canal y velocidad. A continuación explicaré algunos apartes del código realizado:

### 1. Configuración Radio NRF24L01

Configuración de potencia, velocidad y canal de transmisión. Parámetros modificables (algunas posibilidades):

#### Potencia (0-3):

- 0 = -18dBm (0.016mW)
- 1 = -12dBm (0.063mW)
- 2 = -6dBm (0.25mW)
- 3 = 0dBm (1mW) ← Máxima potencia

*Para más potencia se necesitaría Módulos con amplificador (ejemplo): nRF24L01+PA+LNA = hasta 20dBm (100mW)*

#### Velocidad (0-2):

- 0 = 250kbps ← Máximo alcance
- 1 = 1Mbps ← Balance
- 2 = 2Mbps ← Máxima velocidad

Podemos modificar estos valores de acuerdo a nuestros requerimientos:

- Más potencia y menos velocidad = Mayor alcance (hasta 100m)
- Menos potencia y más velocidad = Menor alcance (10-20m) pero mayor respuesta.

### 2. Configuración Servo Motor - Control PWM del Servo

Conversión de ángulos a señales PWM para control servo.

- Límites angulares: max(0, min(180, angulo))
- Pulso mínimo: 500  $\mu$ s (0°)
- Pulso máximo: 2500  $\mu$ s (180°)
- Rango total: 2000  $\mu$ s

### 3. Configuración Acelerómetro MPU6050 - Lectura y Calibración

Lectura de aceleración en 3 ejes y conversión a m/s<sup>2</sup>.

- Escala: 16384.0 LSB/g para  $\pm 2g$
- Calibración: Offsets automáticos en 2 segundos
- Frecuencia muestreo: utime.sleep\_ms(10) en calibración

#### Efectos de variación:

- Cambiar escala: Ajustar sensibilidad ( $\pm 2g$ ,  $\pm 4g$ ,  $\pm 8g$ ,  $\pm 16g$ )
- Tiempo calibración: Aumentar para mayor precisión
- Filtrado: Suavizar datos con promedio móvil

### 4. Protocolo de Comunicación

Protocolo personalizado con sincronización y checksum.

- SYNC\_BYTE: 0xA5 (byte de sincronización)
- Checksum: Verificación de integridad
- Tamaño payload: 4 bytes fijo

#### Efectos de variación:

- Cambiar SYNC\_BYTE: Evitar interferencias con otros sistemas
- Agregar más datos: Aumentar tamaño de payload
- Implementar ACK: Mayor confiabilidad, pero menor velocidad

### 5. Control de Frecuencia

#### Parámetros modificables:

- TX: 10ms = 100 FPS
- RX: 500 $\mu$ s = respuesta ultra rápida

#### Efectos de variación:

- Aumentar sleep: Menor consumo, mayor latencia
- Reducir sleep: Mayor consumo, menor latencia
- Optimizar: Balance entre respuesta y estabilidad

### Resumen de Configuraciones

En general podemos modificar varios parámetros de acuerdo a nuestros requerimientos, resumiendo:

- Para máximo alcance:** nrf.set\_power\_speed(3, 0)
- Para máxima velocidad:** nrf.set\_power\_speed(0, 2) (configuración actual)
- Para bajo consumo:** Aumentar utime.sleep\_ms()
- Para mayor precisión:** Incrementar tiempo de calibración MPU6050

- **Para diferentes servos:** Ajustar rangos PWM en `mover_servo_instantaneo()`

Se adjunta el link del repositorio en GitHub, en donde se puede encontrar los códigos y fotos de evidencia sobre el desarrollo de la practica.

### 3.4. Análisis de Espectro

- Se identificaron los canales WiFi 1, 6 y 11 con alta ocupación.
- Se eligió el canal 100 por estar libre de interferencias significativas.

### 3.5. Presupuesto de Enlace Teórico vs. Experimental

- Para:

$$L(dB) = 32,44 + 20 \log_{10}(2500MHz) + 20 \log_{10}(0,065Km)$$

$$L(dB) = 76,66$$

#### Parámetros utilizados:

- **Potencia de transmisión (Pt):** 0 dBm
- **Ganancia de antena transmisora (Gt):** 0 dBi (antena integrada)
- **Ganancia de antena receptora (Gr):** 0 dBi
- **Frecuencia (f):** 2500 MHz
- **Distancia (d):** 0.065 Km (65 m)

#### Cálculo para 250 kbps:

- Sensibilidad del receptor: -94 dBm
- Potencia recibida teórica:

$$Pr = Pt + Gt + Gr - Lfs$$

$$Pr = 0 + 0 + 0 - 76,66 = -76,66 \text{ dBm}$$

- **Margen del enlace:** -76,66 dBm - (-94 dBm) = 17,34 dB

#### Cálculo para 2 Mbps:

- Sensibilidad del receptor: -82 dBm
- Potencia recibida teórica: -76,66 dBm
- **Margen del enlace:** -76,66 dBm - (-82 dBm) = 5,34 dB

**Interpretación:** El cálculo corregido sin pérdidas adicionales muestra que:

- A 250 kbps existe un amplio margen positivo de 17,34 dB, confirmando la robustez del enlace
- A 2 Mbps el margen positivo de 5,34 dB indica que teóricamente debería funcionar, pero en la práctica factores no ideales como:

El modelo teórico de espacio libre predice correctamente la tendencia observada: mayor tasa de datos reduce el alcance máximo. La diferencia de 12 dB en sensibilidad entre 250 kbps y 2 Mbps es el factor principal que limita el alcance a mayores tasas.

## 4. Conclusiones

- Se logró un enlace inalámbrico confiable entre dos módulos nRF24L01+, con control remoto de un servomotor y visualización en LCD.
- El análisis del espectro permitió seleccionar un canal libre, minimizando interferencias.
- Las pruebas de alcance confirmaron que a menor tasa de datos y mayor potencia, se obtiene un mayor alcance.
- El análisis del bus SPI permitió comprender en detalle el protocolo de comunicación con el módulo.
- Los resultados experimentales de alcance fueron consistentes con el modelo teórico de espacio libre.
- Los resultados experimentales fueron consistentes con la teoría, mostrando aproximadamente 35 m de alcance máximo a 250 kbps vs 20 m a 2 Mbps. Las discrepancias entre el modelo teórico y la práctica se atribuyen a efectos del canal real no considerados en el modelo ideal de espacio libre.