Comunicación por láser en espacio libre con modulación On-Off Keying y protocolo serie universal asíncrono

Julio Cesar Hernández Mitjans[1], Luis Alberto Torres Muñoz[1], Natalia de Jesús Gaeta Verdin[1], Geliztle Alejandra Parra Escamilla[2], David Ignacio Serrano García[1], Francisco Joel Cervantes Lozano[1]
[1]Universidad de Guadalajara, Centro Universitario de Ciencias Exactas e Ingenierías, Departamento de Ingeniería Electro-Fotónica
[2]Universidad de Guadalajara, Centro Universitario de Ciencias Exactas e Ingenierías, Departamento de Ciencias Computacionales
julio.hernandez7319@alumnos.udg.mx

Resumen

En este trabajo se construyó y caracterizó un sistema de comunicación por láser en espacio libre (Free Space Optics, FSO) con modulación On-Off Keying (OOK). Para enviar los caracteres en código ASCII (American Standard Code for Information Interchange) se usó el formato de trama del protocolo UART (Universal Asynchronous Receiver-Transmitter), generado con la librería SoftwareSerial de Arduino. El sistema emplea componentes económicos: dos Arduino Uno, un láser rojo de 650 nm como transmisor y un fotorresistor (LDR) como detector. Generar la señal OOK con la trama UART evita programar a mano los retardos entre bits y mantiene el código simple. Con el sistema a 50 cm de distancia y con iluminación constante, se transmitieron números, letras y palabras; los 650 caracteres enviados (10 repeticiones por carácter, 5200 bits) llegaron sin errores. Por separado, se midió la respuesta del LDR con un generador de señales y un osciloscopio. Su salida solo sigue con claridad la conmutación del láser hasta unos 18 Hz: el tiempo de subida es de ≈19 ms, la constante de tiempo de ≈8.6 ms y el ancho de banda efectivo de ≈18 Hz. A frecuencias mayores (232 Hz, 330 Hz y 6.8 kHz) la señal se atenúa mucho, hasta 87% a 6.8 kHz, o se distorsiona. Los dos puntos débiles del sistema son el LDR, por su respuesta lenta, y el Arduino Uno (16 MHz, 8 bits), que deja poco margen para funciones adicionales. Los resultados, todavía preliminares, indican que es posible armar un enlace FSO de bajo costo con componentes comerciales.

Palabras clave: Free Space Optics (FSO); On-Off Keying (OOK); Universal Asynchronous Receiver-Transmitter (UART); comunicación óptica.

Introducción

La fibra óptica es el estándar de las redes de alta velocidad y ha cambiado la forma de transmitir datos en las últimas décadas. Sin embargo, hay lugares donde tender fibra no es viable, ya sea por costo, por tiempo o por la geografía. Para esos casos existe la tecnología Free Space Optics (FSO), que envía datos con haces de luz a través del aire y combina las velocidades de la fibra con la flexibilidad de los enlaces inalámbricos (Khalighi y Uysal, 2014). Entre los esquemas de modulación que se usan en FSO, el On-Off Keying (OOK) es de los más sencillos: es barato y fácil de implementar (Ghassemlooy *et al*., 2019), por lo que resulta adecuado para proyectos de laboratorio (Leccese y Cagnetti, 2019). Para generar el OOK en la práctica, este trabajo usa el formato de trama del protocolo UART (Universal Asynchronous Receiver-Transmitter) como capa de enlace. Un UART agrupa cada carácter en una trama de bits (uno de inicio, 8 de datos y uno de parada) a una velocidad que define el usuario (Microchip Technology, 2018). En muchos microcontroladores esta función la hace un periférico de hardware; en el Arduino Uno también puede emularse por software con la librería SoftwareSerial, que es la opción que se usó aquí. Con print() de esa librería, el microcontrolador manda los datos al láser en una trama estándar, sin que el usuario tenga que escribir retardos con delay(). Este trabajo implementa así un sistema de comunicación por láser con FSO y OOK usando componentes baratos (Arduino Uno, láser de 650 nm y LDR), con el que se transmitieron números, letras y palabras. Como parte de la validación se caracterizó el LDR para conocer sus limitaciones dinámicas (tiempos de subida y de caída, constante de tiempo y ancho de banda) y saber hasta qué frecuencia puede seguir una señal modulada, lo que permite identificarlo como el principal factor que limita la velocidad. Los resultados hacen de este desarrollo un buen punto de partida para pasar a arquitecturas más rápidas (ESP32) y a detectores de mayor ancho de banda (fototransistor o fotodiodo PIN), con la meta de superar 1 Mbps e incorporar mecanismos de corrección de errores (checksums, CRC, FEC) (Banzi y Shiloh, 2022; Maier *et al*., 2017).

Para dar una idea del desempeño alcanzado conviene compararlo con otros trabajos de bajo costo. (Eldallal *et al*., 2019) montaron un sistema FSO con Arduino Uno y un láser de 650 nm, parecido al nuestro, pero con un fotodiodo TSL250R como detector. Con él lograron 125 kbps para texto y 506.3 kbps para audio, muy por encima de los 9600 baudios (≈7.7 kbps de datos útiles) que se usaron con el LDR. La diferencia sugiere que el LDR es el principal factor limitante en nuestro sistema y respalda la idea de sustituirlo por un fotodiodo PIN o un fototransistor. Por otro lado, algunos proyectos estudiantiles han probado arreglos de fotodiodos con realineamiento mecánico para reducir el efecto del clima en enlaces FSO de bajo costo (Xiao, 2020). En conjunto, Arduino resulta una base accesible para experimentar con comunicaciones ópticas en espacio libre, pero alcanzar tasas de transmisión altas exige detectores rápidos.

El objetivo de este trabajo es implementar y caracterizar un enlace FSO de bajo costo con modulación OOK, construido con Arduino Uno, un láser rojo de 650 nm y un LDR como detector, e identificar las limitaciones que impone este hardware. La contribución es doble: por un lado, la señal OOK se genera con el formato de trama del protocolo UART, producido con la librería SoftwareSerial, en lugar de programar retardos explícitos; por otro, se caracteriza experimentalmente la respuesta dinámica del LDR (tiempos de subida y de caída, constante de tiempo y ancho de banda), lo que sirve de base para pasar a un microcontrolador y a un detector más rápidos.

1.1 Free Space Optics (FSO)

Free Space Optics (FSO) es una tecnología de comunicación inalámbrica que transmite datos con haces de luz a través del aire, sin cables (Khalighi y Uysal, 2014). Los sistemas FSO pueden alcanzar distancias de hasta 70 km y velocidades de hasta 20 Gbps en condiciones óptimas. Entre sus ventajas están que no requieren licencias de espectro, se instalan rápido, no les afectan las interferencias electromagnéticas y ofrecen buena seguridad gracias a la direccionalidad del haz. Sus desventajas principales son la sensibilidad al clima (niebla, lluvia, nieve), la necesidad de una línea de visión directa y la alineación precisa entre transmisor y receptor (Ghassemlooy *et al*., 2019) (ver Figura 1).

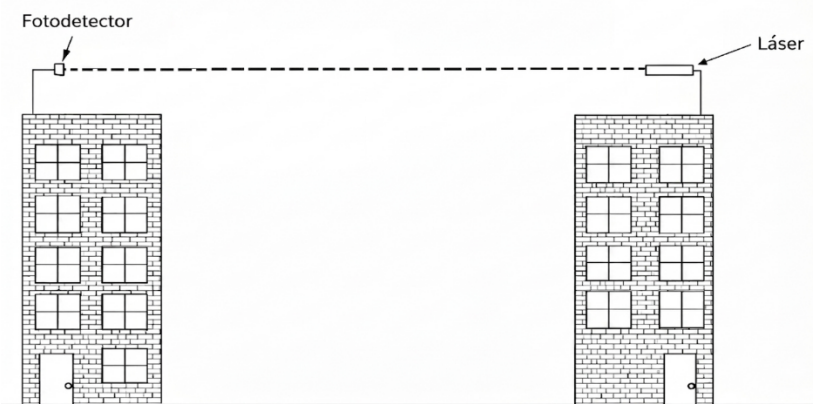


Figura 1. Comunicación FSO en un sistema implementado entre dos edificios.
Fuente: Autoría propia.

1.2 Modulación On-Off Keying (OOK)

La modulación On-Off Keying (OOK) es la forma más simple de modular luz digitalmente: el láser se enciende para representar un "1" y se apaga para representar un "0" (Ghassemlooy *et al*., 2019). Sus ventajas son la sencillez de implementación, el bajo costo, el bajo consumo de energía y que funciona con detectores simples, como un LDR o un fotodiodo básico. Entre sus desventajas están la sensibilidad a la luz ambiental, a las vibraciones y a la desalineación, su baja eficiencia espectral (1 bit por símbolo) y la necesidad de un umbral de detección que se adapte cuando las condiciones cambian (Leccese y Cagnetti, 2019) (ver Figura 2).

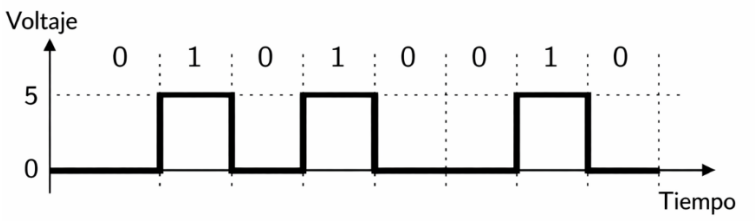


Figura 2. Modulación de comunicación OOK.
Fuente: Autoría propia

Metodología Propuesta

2.1 Implementación práctica de OOK con UART

En la práctica, la señal OOK puede generarse de dos maneras. La primera consiste en controlar un pin digital directamente con digitalWrite() y fijar los tiempos a mano con delay() o delayMicroseconds(). Es una buena opción didáctica, pero pierde precisión temporal cuando la velocidad pasa de unos pocos cientos de baudios, porque otras tareas del microcontrolador alteran los tiempos. La segunda, que es la que se usó aquí, aprovecha el formato de trama del protocolo UART (Universal Asynchronous Receiver-Transmitter). Al escribir un carácter con print() de la librería SoftwareSerial, esta genera sola la trama completa (bit de inicio, 8 bits de datos y bit de parada) (Microchip Technology, 2018), con tiempos de bit calibrados respecto al reloj del microcontrolador y sin que el usuario programe retardos. La señal digital resultante se conecta directamente al láser y produce una modulación OOK con una trama estándar. Los UART suelen trabajar a velocidades como 9600, 19200 o 57600 baudios; en este trabajo se usaron 9600 baudios.

2.2 Implementación con Arduino Uno

En la primera etapa del proyecto se armó un sistema de comunicación por láser con dos placas Arduino Uno.

Transmisor: un Arduino Uno conectado directamente a un módulo láser rojo de 650 nm. Se eligió esa longitud de onda porque es visible, barata y fácil de conseguir, lo que facilita alinear el haz a simple vista entre transmisor y receptor, algo muy útil en esta etapa experimental. El Arduino genera la señal OOK con el formato de trama UART (Universal Asynchronous Receiver-Transmitter): se configura un puerto serie emulado por software con la librería SoftwareSerial, que permite usar cualquier pin digital como TX. Aquí se usó el pin digital 9 como TX, conectado directamente al módulo láser, que se alimenta a través de ese pin: recibe 5 V cuando el nivel lógico es '1' (láser encendido) y 0 V cuando es '0' (láser apagado). La función LaserSerial.print() genera la trama UART (bit de inicio, 8 bits de datos y bit de parada) a una velocidad configurable, y esa trama enciende y apaga el láser según el nivel de cada bit, lo que produce la modulación OOK. Así no hace falta manejar los retardos con delay(): la librería se encarga de la temporización y el código queda más simple. El módulo láser ya incluye su propia resistencia limitadora, por lo que no se necesitó una resistencia externa.

Receptor: un segundo Arduino Uno conectado a un LDR (fotorresistor) mediante un divisor de voltaje con dos resistencias fijas de 100 kΩ. El divisor convierte la variación de resistencia del LDR en un voltaje, que el Arduino lee en su entrada analógica (pin A0). El ADC se configuró con resolución de 10 bits, referencia interna de 5 V y un tiempo de conversión de unos 100 µs. El receptor muestrea el pin analógico de forma continua, sin retardos adicionales, con lo que se alcanza una tasa de muestreo efectiva de aproximadamente 9.6 kHz.

La Figura 3 muestra el esquema eléctrico completo del sistema.

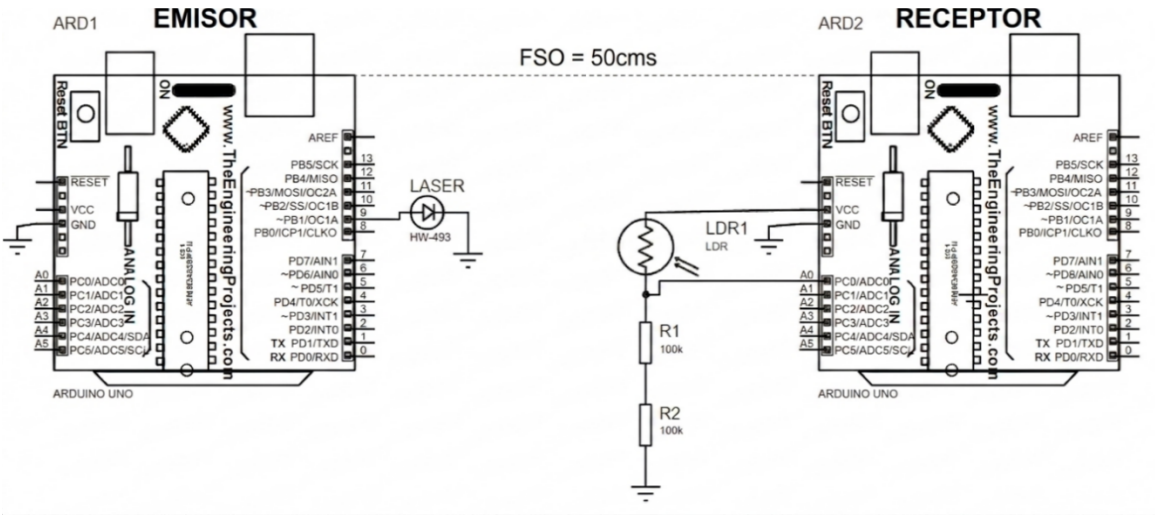


Figura 3. Esquema eléctrico del sistema. Se muestran las conexiones del láser al pin digital del Arduino transmisor y del divisor de voltaje del LDR (dos resistencias de 100 kΩ) al pin analógico A0 del Arduino receptor.
Fuente: Autoría propia

2.2.1 Código de implementación con UART

A continuación se muestra el código del transmisor basado en UART; el pin digital 9 alimenta directamente el módulo láser:

```
#include <SoftwareSerial.h>

// RX = pin 8 (no usado en el transmisor)
// TX = pin 9 (conectado al ánodo del láser)
SoftwareSerial LaserSerial(8, 9);

void setup() {
  LaserSerial.begin(9600); // Velocidad de transmisión (bits por segundo)
}

void loop() {
  LaserSerial.print("Hola:"); // La librería genera la trama UART (señal OOK)
  delay(2000); // Pausa entre envíos
}
```

Cuando LaserSerial.print() envía un '1' binario, el pin 9 pasa a 5 V y enciende el láser; cuando envía un '0', baja a 0 V y lo apaga. El receptor, por su parte, lee la señal del fotodetector en la entrada analógica A0 (ver la descripción del receptor) y reconstruye por software los caracteres transmitidos.

2.3 Sistema emisor-receptor

La distancia entre transmisor y receptor fue de 50 cm, en condiciones controladas de laboratorio: iluminación constante, sin vibraciones y con el montaje fijo sobre una mesa óptica (ver Figura 4).

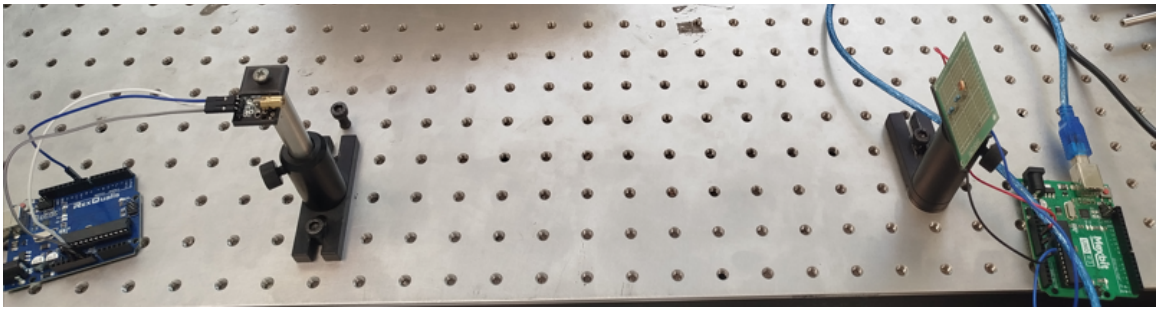


Figura 4. Sistema Emisor – Receptor, montaje en la mesa óptica.
Fuente: Autoría propia

Resultados

3.1 Transmisión de caracteres y BER

Con la configuración descrita se lograron transmitir números, letras y palabras completas. Se enviaron los diez dígitos decimales (del 0 al 9), solos y en secuencias cortas, y las 26 letras del alfabeto en mayúsculas y minúsculas (52 caracteres en total), y en todos los casos se verificó que la recepción fuera correcta. También se enviaron palabras completas, como "Hola" (ver Figura 5) y otras de longitud parecida, lo que muestra que el sistema puede comunicar información alfanumérica básica.

Cada carácter y cada palabra se envió 10 veces, con una tasa de éxito del 100% en condiciones controladas de laboratorio: distancia fija de 50 cm, iluminación ambiental constante, montaje fijo sobre una mesa óptica y sin fuentes de vibración importantes. Estos resultados indican que el enlace óptico es viable con tecnología de bajo costo y que se repite en las condiciones de prueba.

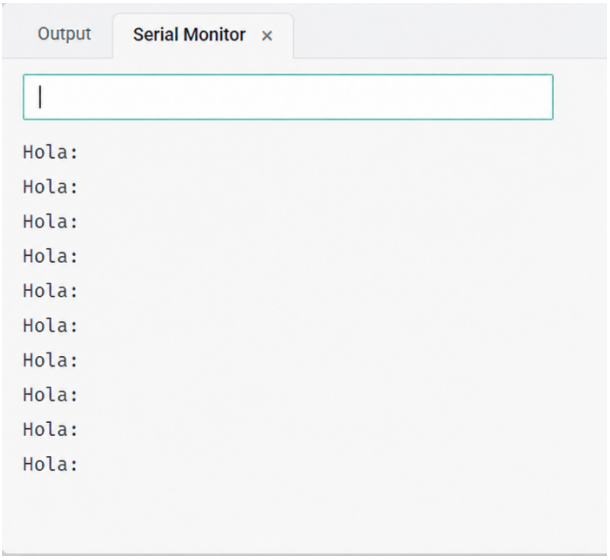


Figura 5. Palabra "Hola:" recibida en el Arduino conectado al LDR.
Fuente: Autoría propia

Para cuantificar la fiabilidad del enlace se calculó la tasa de error de bit (BER). En total se transmitieron 650 caracteres (10 dígitos, 52 letras y 3 palabras, cada uno repetido 10 veces), que equivalen a 5,200 bits con una codificación ASCII de 8 bits por carácter. No se detectó ningún error en las transmisiones, por lo que el BER es 0 (ecuación 1).

$$BER = \frac{\text{Número de bits erróneos}}{\text{Número de bits transmitidos}} = \frac{0}{5200} = 0 \quad (1)$$

Este valor no implica que no pueda haber errores en condiciones más exigentes, pero indica que, dentro de las condiciones de prueba, la comunicación fue fiable.

3.2 Caracterización experimental del LDR

Para conocer las capacidades del LDR con el divisor de 100k/100k, se caracterizó su respuesta en frecuencia. El montaje consistió en un generador de señales que modulaba un láser rojo de 650 nm, con el LDR conectado a un osciloscopio. Esta caracterización se hizo con el generador y no con la trama UART del Arduino. Además, el láser se colocó literalmente pegado al LDR (distancia nula), que es el mejor caso posible de iluminación; en condiciones reales de operación, con emisor y receptor separados, la iluminación sobre el LDR será distinta.

Caso 1 – Frecuencia de aproximadamente 18 Hz

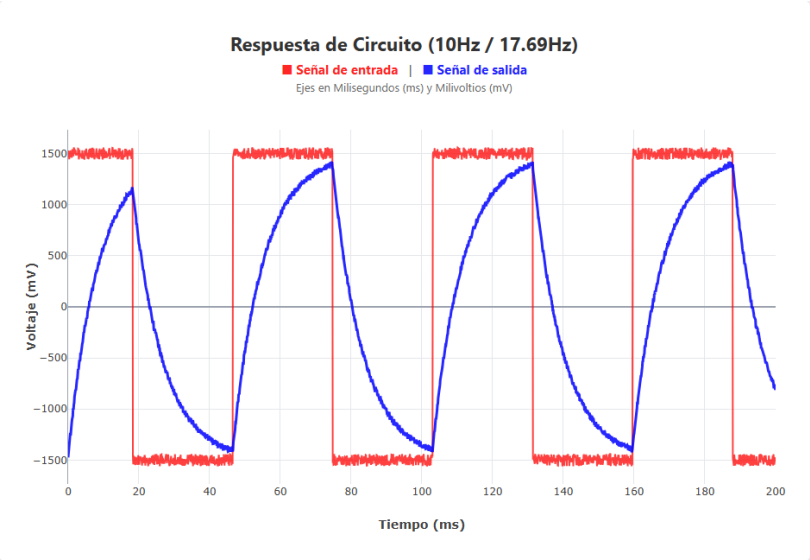


Figura 6. Respuesta del sistema a ~18 Hz. CH2 (rojo): entrada al láser (500 mV/div). CH1 (azul): salida del LDR (20 mV/div). Fuente: Autoría propia

En la Figura 6, la salida del LDR sigue la conmutación del láser con una respuesta exponencial marcada: los bordes están muy redondeados y la señal tarda casi todo el semiperiodo (≈28 ms) en llegar a su valor final. Es la única medición en la que la salida sigue claramente la conmutación, y por eso permite estimar con confianza los parámetros dinámicos del detector.

Caso 2 – Frecuencia de aproximadamente 6.8 kHz

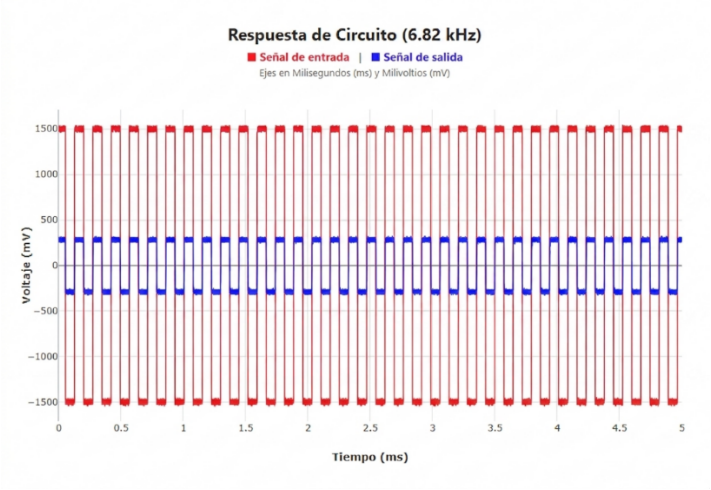


Figura 7. Respuesta del sistema a ~6.8 kHz. CH2 (rojo): entrada al láser (500 mV/div). CH1 (azul): salida del LDR (50 mV/div). Fuente: Autoría propia

En la Figura 7, la señal de salida conserva forma cuadrada, con flancos abruptos y una amplitud muy baja; es demasiado débil para que la detecte el ADC de un microcontrolador.

Con el generador se registró señal a la salida del LDR hasta 6.8 kHz, aunque con una amplitud muy reducida (Caso 2). A partir del oscilograma del Caso 1 (~18 Hz) se midieron los parámetros dinámicos del detector: tiempo de subida (t_r) ≈ 19 ms y tiempo de caída (t_f) ≈ 19 ms. Con estos valores se estimó la constante de tiempo mediante la ecuación (2):

$$t_r = 2.2 \cdot \tau \tag{2}$$

De aquí se obtiene $\tau \approx 8.6$ ms, y con la ecuación (3) se calculó el ancho de banda efectivo:

$$BW = \frac{0.35}{t_r} \tag{3}$$

Se obtiene $BW \approx 18\ Hz$. Este ancho de banda concuerda con que 18 Hz sea la frecuencia más alta a la que la salida siguió con claridad la conmutación del láser.

3.2.1 Curva de respuesta en frecuencia del LDR

Para complementar la caracterización temporal del detector se construyó la curva de respuesta en frecuencia del LDR (Figura 8), que relaciona la amplitud de salida (mV p-p) con la frecuencia de modulación (Hz). El eje horizontal está en escala logarítmica para poder ver todo el rango de frecuencias (de ~10 Hz a 10 kHz).

Los puntos experimentales corresponden a las mediciones a 18 Hz, 232 Hz, 330 Hz y 6.8 kHz. La amplitud de salida es máxima a 18 Hz, donde casi no hay atenuación. A partir de 232 Hz la amplitud cae de forma notable, y la caída se acentúa a 330 Hz. A 6.8 kHz la amplitud es muy baja, lo que indica que el LDR ya no sirve para una transmisión modulada rápida.

La curva muestra el comportamiento de filtro pasa bajas típico de un dispositivo de respuesta lenta. El punto de corte (-3 dB) está en torno a 18 Hz, lo que concuerda con el ancho de banda estimado a partir del tiempo de subida (ecuación 3). Por encima de esa frecuencia, las amplitudes registradas (reducciones de 50% a 232 Hz, 62% a 330 Hz y 87% a 6.8 kHz respecto de 18 Hz) son mayores que las que daría una respuesta de primer orden con ese ancho de banda, que dejaría del orden de 10% de la amplitud a 232 Hz y menos de 1% a 6.8 kHz (ver Figura 8).

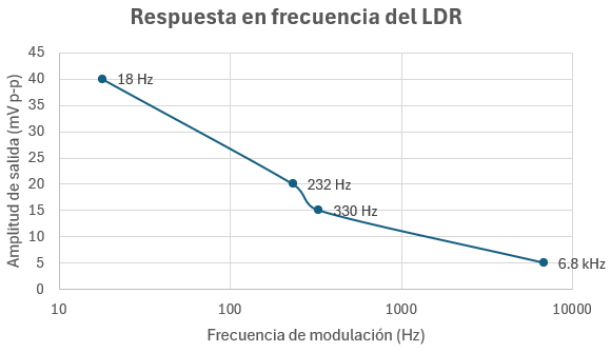


Figura 8. Curva amplitud-frecuencia del detector LDR (escala logarítmica en el eje horizontal). Se indican los puntos experimentales (18 Hz, 232 Hz, 330 Hz y 6.8 kHz) y la tendencia general de atenuación con la frecuencia.
Fuente: Autoría propia

3.3 Limitaciones

Aunque la transmisión de texto funcionó, se identificaron dos limitaciones principales en el sistema de la Fase 1.

Limitación 1: Microcontrolador. El Arduino Uno es lento y tiene poca capacidad de procesamiento, por lo que no puede ejecutar tareas adicionales sin interrumpir la comunicación. La Tabla 1 resume sus características y las compara con las del ESP32 (Maier *et al*., 2017).

Tabla 1*. Comparativa cuantitativa entre Arduino Uno y ESP32*

Característica	Arduino Uno	ESP32
Frecuencia de reloj	16 MHz	240 MHz (15× mayor)
Arquitectura	8 bits	32 bits
Núcleos de procesamiento	1 (single-core)	2 (dual-core)
Velocidad OOK práctica	~115 kbps (teórica)	>1 Mbps (estimada)
Corrección de errores en paralelo	No posible	Sí (núcleo dedicado)

Limitación 2: Detector (LDR). La caracterización del LDR (sección 3.2) mostró que la salida del detector solo sigue claramente la conmutación del láser hasta unos 18 Hz, en línea con su ancho de banda estimado (~18 Hz). Por encima de esa frecuencia se registró señal hasta 6.8 kHz, pero distorsionada o con amplitud reducida: aproximadamente 50% menos a 232 Hz, 62% menos a 330 Hz y 87% menos a 6.8 kHz respecto del valor a 18 Hz.

Conclusiones

En la primera fase del proyecto se implementó un sistema de comunicación por láser con Arduino Uno y modulación OOK, con resultados preliminares sobre la viabilidad de un enlace óptico con componentes de bajo costo en condiciones de laboratorio controladas. La caracterización del LDR mostró que su salida solo sigue claramente la conmutación del láser hasta unos 18 Hz, con un ancho de banda efectivo estimado en ≈18 Hz y una fuerte atenuación a frecuencias mayores. Se logró transmitir números, letras y palabras con una tasa de éxito del 100% en condiciones controladas (50 cm de distancia, iluminación constante). Como trabajo futuro se propone caracterizar la salida del LDR durante la transmisión de la trama UART.

Se identificaron dos limitaciones principales. La primera es el LDR, el principal cuello de botella para escalar la velocidad: su respuesta lenta (tiempo de subida de ≈19 ms) limita su ancho de banda a ≈18 Hz. La segunda es el Arduino Uno (16 MHz, 8 bits, un solo núcleo), cuya velocidad máxima teórica ronda los 115 kbps y que, al tener un solo núcleo, no puede ejecutar corrección de errores sin interrumpir la comunicación.

Migrar al ESP32 (240 MHz, 32 bits, dos núcleos) permitiría superar 1 Mbps y ejecutar tareas de corrección de errores (checksums, CRC, FEC) en un núcleo dedicado. Aun así, ese cambio no basta por sí solo: también hay que reemplazar el LDR por un fototransistor o un fotodiodo PIN, que ofrecen anchos de banda mucho mayores (>100 kHz). Con esa combinación se espera alcanzar tasas superiores a 1 Mbps. Los resultados obtenidos hacen de este desarrollo un buen punto de partida para evolucionar hacia arquitecturas más rápidas y detectores de mayor ancho de banda.

Una contribución metodológica de este trabajo fue usar el formato de trama UART (Universal Asynchronous Receiver-Transmitter), generado con la librería SoftwareSerial, para producir la señal OOK en lugar de programar retardos explícitos con delay(). Con ello el código se simplificó y la trama quedó estandarizada (bits de inicio, datos y parada), aunque, como la librería emula el UART por software, el procesador queda ocupado durante la transmisión de cada carácter. La migración propuesta al ESP32 conservará este esquema UART sobre OOK, pero con el periférico UART por hardware y aprovechando sus tres puertos (Maier *et al*., 2017) para transmitir, recibir y depurar al mismo tiempo sin interferencias, y para liberar al procesador durante la transmisión.

Agradecimientos

Quiero agradecer a la Secretaría de Ciencia, Humanidades, Tecnología e Innovación (SECIHTI) el apoyo financiero recibido durante mis estudios de maestría. Asimismo, agradezco a la Universidad de Guadalajara por las facilidades brindadas para el desarrollo de este proyecto.

Referencias

Banzi, M., y Shiloh, M. (2022). *Getting started with Arduino* (4th ed.). Maker Media.

Eldallal, T. A., Fayed, H. A., y Aly, M. H. (2019). Data transmission prototypes through wireless optical communication link using Arduino microcontroller. *Journal of Optoelectronics and Advanced Materials, 21*(7–8), 459–469. https://joam.inoe.ro/articles/data-transmission-prototypes-through-wireless-optical-communication-link-using-arduino-microcontroller/fulltext

Ghassemlooy, Z., Popoola, W., y Rajbhandari, S. (2019). *Optical wireless communications: System and channel modelling with MATLAB®* (2nd ed.). CRC Press.

Khalighi, M. A., y Uysal, M. (2014). Survey on free space optical communication: A communication theory perspective. *IEEE Communications Surveys y Tutorials, 16*(4), 2231–2258. 10.1109/COMST.2014.2329501

Leccese, F., y Cagnetti, M. (2019). Low-cost laser communication system for educational purposes. *International Journal of Electrical Engineering Education, 56*(2), 158–173. DOI=10.5120/ijca2025924957

Maier, A., Sharp, A., y Vagapov, Y. (2017). Comparative analysis and practical implementation of the ESP32 microcontroller module for the Internet of Things. In *2017 Internet Technologies and Applications (ITA)* (pp. 143–148). 10.1109/ITECHA.2017.8101926

Microchip Technology. (2018). *ATmega48A/PA/88A/PA/168A/PA/328/P data sheet* (DS40002061A). https://www.microchip.com/en-us/product/atmega328

Xiao, C. (2020). *A low-cost approach to handheld laser diode communication* (Project EBED029). Society for Science.