



**GRUPO TECNOLÓGICO UNIVERSITARIO  
CAMPUS TOLUCA**

**PROYECTO INTEGRADOR:**

**COMUNICACIÓN EN CÓDIGO MORSE: UN SISTEMA VISUAL Y  
AUDITIVO EN PROGRAMACIÓN**

**INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**INTEGRANTES:**

Bruno Carbajal Bladimir Santiago

Enriquez González Luis Antonio

Gómez De La Cruz Israel

Jiménez Dado Efraín

Martínez Hernández Jonathan Eduardo

Morales Diaz Geraldine Michelle

Pontón Hernández Alejandro

Santamaria Juárez Kevin

Torres Flores José María

**PROFESORES:**

Dra. Yanet Romero Arriaga (Álgebra Lineal).

Dr. Mariano Cruz Albarrán (Cálculo Diferencial e Integral).

Mtro. Ángel Gabriel Lojero Rosales (Electrónica).

Mtro. José Alberto González Avendaño (Habilidades de Expresión Oral y Escrita).

Ing. Serena Mejía Pichardo (Programación).

**Fecha de Presentación: 08/04/2025.**

**ÍNDICE**

<b>FASE I: IDEOLÓGICA .....</b>	<b>6</b>
1.1 DESCRIPCIÓN DE LA IDEA .....	6
1.2 IDENTIFICACIÓN DE LA NECESIDAD .....	6
1.3 NOMBRE DEL PROYECTO .....	7
1.4 ANTECEDENTES .....	7
<b>FASE II: CONTEXTUAL .....</b>	<b>8</b>
2.1 PROBLEMÁTICA .....	8
2.2 JUSTIFICACIÓN .....	8
2.3 OBJETIVOS .....	9
2.3.1 OBJETIVO GENERAL .....	9
2.3.2 OBJETIVOS ESPECÍFICOS .....	9
<b>FASE III: MARCO TEÓRICO (CONCEPTUAL).....</b>	<b>10</b>
3.1. ¿QUÉ ES EL CÓDIGO MORSE? .....	10
3.2. FUNCIONALIDAD .....	10
3.3. CÓDIGO MORSE .....	12
<b>FASE IV: DESARROLLO .....</b>	<b>14</b>
4.1. ¿PARA QUIÉN ESTÁ DIRIGIDO? .....	14
4.2. DESCRIPCIÓN .....	14
4.3. DISEÑO DEL DISPOSITIVO .....	15
4.4. RECURSOS Y MATERIALES NECESARIOS .....	15
4.3.1 COSTOS Y GANANCIAS .....	16
4.5. SIMULACIÓN .....	17
<b>FASE V: CONCLUSIÓN .....</b>	<b>18</b>
5.1. CONCLUSIONES .....	18
5.2. RECOMENDACIONES .....	19
<b>ANEXO A: MATERIALES UTILIZADOS .....</b>	<b>20</b>
<b>ANEXO B: DIAGRAMAS DEL DISPOSITIVO .....</b>	<b>26</b>
<b>ANEXO C: CÓDIGOS DE LA PROGRAMACIÓN DEL DISPOSITIVO .....</b>	<b>27</b>
<b>ANEXO D: CÓDIGO QR DEL MANUAL DE USUARIO .....</b>	<b>58</b>
<b>ANEXO E: CRONOGRAMA DE ACTIVIDADES .....</b>	<b>58</b>
5.3. FUENTES DE CONSULTA .....	58

## INTRODUCCIÓN

El código Morse, desarrollado en la primera mitad del siglo XIX, ha sido una de las formas más eficaces y duraderas de comunicación a través de señales sonoras o visuales. Aunque inicialmente se utilizó en el ámbito de las telecomunicaciones y en situaciones de emergencia, su uso se ha extendido a lo largo del tiempo en diversas disciplinas, desde la aviación hasta la exploración espacial. Hoy en día, a pesar de los avances tecnológicos, el código Morse sigue siendo un recurso valioso, tanto en situaciones críticas como en la enseñanza de conceptos fundamentales de codificación y transmisión de información. Su simplicidad, basada en la combinación de puntos y rayas para representar caracteres alfabéticos y numéricos, lo convierte en una herramienta ideal para introducir conceptos de programación y de sistemas binarios en la educación.

Este proyecto tiene como objetivo la creación de un traductor de texto a código Morse que, a través de una interfaz interactiva, permita a los usuarios aprender y experimentar con este sistema de comunicación. Diseñado principalmente con fines educativos, el traductor no solo facilitará la conversión de texto a código Morse, sino que también ofrecerá la posibilidad de practicar la decodificación, lo que promoverá un aprendizaje completo del sistema. De esta manera, se busca proporcionar a los estudiantes y aprendices una comprensión profunda de los fundamentos del código Morse, estimulando su curiosidad y fortaleciendo sus habilidades en la codificación y la decodificación de mensajes.

La herramienta contará con funcionalidades que permitirán al usuario tanto ingresar texto para ser convertido a código Morse como escuchar la secuencia de señales resultante, con el objetivo de reforzar la asociación entre las representaciones gráficas y sonoras del código. Además, se incluirán ejercicios y pruebas que desafiarán al usuario a decodificar mensajes en

Comunicación en código Morse: Un sistema visual y auditivo en Programación.

Morse, lo que fomentará la práctica y el dominio del sistema. Este enfoque interactivo y práctico no solo busca enseñar el código Morse de manera efectiva, sino también ofrecer una experiencia educativa que sea atractiva, accesible y adecuada para diferentes niveles de conocimiento.

El diseño de este traductor se enfocará en ser intuitivo, accesible y fácil de usar, lo que permitirá a estudiantes de distintas edades y niveles educativos explorar el código Morse de forma autónoma o guiada. La plataforma será diseñada pensando en una experiencia de usuario fluida y clara, que incentive la experimentación y el aprendizaje autónomo. Asimismo, se utilizarán tecnologías actuales que garanticen la compatibilidad y facilidad de uso, permitiendo su integración en diversos entornos educativos, desde aulas tradicionales hasta entornos de aprendizaje en línea.

En resumen, este proyecto no solo tiene como objetivo la creación de una herramienta para la conversión entre texto y código Morse, sino que también aspira a ser una plataforma educativa que promueva el aprendizaje activo, la comprensión del código Morse y la aplicación de conceptos clave de la ingeniería de la información. Al integrar teoría con práctica en un entorno interactivo, se busca proporcionar una experiencia educativa completa que motive a los estudiantes a explorar y comprender un sistema de comunicación que, aunque antiguo, sigue siendo relevante en la formación de futuros profesionales en áreas relacionadas con las telecomunicaciones, la seguridad y la programación.

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

El documento de este proyecto se estructura en varias fases con el objetivo de organizar el proceso de creación del traductor de código Morse de una manera lógica y eficiente. Cada fase cumple una función específica dentro del desarrollo, permitiendo un enfoque ordenado y metodológico. De esta manera, se abordan desde la concepción de la idea hasta su implementación y evaluación final. La división en fases facilita la identificación de problemas, la asignación de recursos y la optimización del tiempo de trabajo. Además, permite asegurar que se cumplan los objetivos planteados en cada etapa. Gracias a esta organización, el proyecto puede avanzar de manera estructurada y con una base teórica bien definida.

## **FASE I: IDEOLÓGICA**

### **1.1 DESCRIPCIÓN DE LA IDEA**

El código Morse es un medio de comunicación el cual se caracteriza por estar basado en líneas y puntos. Este proyecto consiste en realizar un traductor de lenguaje natural, es decir, del español a código Morse y de código Morse a lenguaje natural; al ser un sistema audiovisual se podrá apreciar mediante un buzzer y una luz (foco, leds, etc.), que permita la comunicación principalmente a largas distancias o bien en zonas donde carecen de conexiones de red o alguna señal de intercomunicación.

Para el primer caso, la entrada de datos o el mensaje que se desee traducir, el usuario lo ingresará desde un equipo de cómputo, dando como resultado la traducción en código Morse apreciándolo tanto de manera visual como auditivo. En el segundo caso, que implica la traducción de código Morse al español, por medio de un push-button, el usuario ingresará el mensaje para que posteriormente este se vea en la pantalla del equipo de cómputo.

### **1.2 IDENTIFICACIÓN DE LA NECESIDAD**

Este proyecto no solo es una herramienta como medio de traductor, sino que responde a diversas necesidades, sin embargo, la principal de este es la educación ya que permitirá practicar tanto el código Morse en sus diferentes formas (texto, sonido y medio visual) como el funcionamiento del traductor.

Otra necesidad que se pudiera subsanar con el proyecto sería la comunicación en situaciones de emergencia principalmente en desastres naturales donde los medios de comunicación actuales se ven comprometidos.

Comunicación en código Morse: Un sistema visual y auditivo en Programación.

Por último, la necesidad que nos gustaría poder alcanzar en un futuro sería en el ámbito militar por su fiabilidad, simplicidad y capacidad de transmisión en diversas condiciones. Dado que es un sistema de comunicación que está basado en un código sencillo, ágil y preciso, permite que nuestro dispositivo tenga la posibilidad de ser utilizado por profesionales de esta área.

### **1.3 NOMBRE DEL PROYECTO**

Comunicación en código Morse: Un Sistema Visual y Auditivo en Programación.

### **1.4 ANTECEDENTES**

Es un proyecto meramente para fines académicos con el fin de innovar el sistema de comunicación por medio del código Morse, permitiendo que la mayoría de las personas logren entender lo que se desee transmitir, involucrando cada una de las áreas de la ingeniería en sistemas computacionales con módulos del segundo cuatrimestre.

## **FASE II: CONTEXTUAL**

### **2.1 PROBLEMÁTICA**

Dificultad en la enseñanza y el aprendizaje del código Morse debido a la necesidad de memorización manual y la falta de herramientas interactivas que faciliten su comprensión y aplicación práctica.

### **2.2 JUSTIFICACIÓN**

El código Morse, debido a su simplicidad y eficiencia, también puede ser útil en sistemas de seguridad para una comunicación discreta. Dada la accesibilidad de tecnologías y softwares de código abierto, se pretende crear un sistema de traducción automático entre el Código Morse y el lenguaje natural, en el que la traducción sea de acuerdo con la necesidad del usuario y en el caso que la necesidad sea traducir de código Morse a lenguaje natural esto se podrá realizar por medio de un push-button, por el cual será posible introducir el mensaje.

Al ser un proyecto meramente para fines académicos, también puede ser utilizado como medio de aprendizaje para toda aquella persona que tenga el interés en aprender el código Morse, pues al ser un traductor, no necesariamente tiene que saber lo básico pues este sistema se encargará de realizar la traducción tanto de letras, palabras y oraciones que resulten del interés propio.



## **2.3 OBJETIVOS**

### **2.3.1 OBJETIVO GENERAL**

Diseñar y desarrollar un programa que permita ingresar un texto en lenguaje natural desde la computadora y lo traduzca automáticamente a código Morse, mostrando la traducción tanto de manera visual (a través de un foco) como auditiva (mediante un zumbador). Además, el sistema debe permitir enviar un mensaje en código Morse utilizando un pulsador, y mostrar en pantalla el mensaje recibido ya traducido a lenguaje natural.

### **2.3.2 OBJETIVOS ESPECÍFICOS**

- Desarrollar una herramienta de traducción automática entre el lenguaje natural y código Morse, asegurando que la herramienta sea accesible para usuarios con distintos niveles de conocimiento en Morse.
- Integrar diversas formas de interacción con el usuario, como la entrada de información a través de texto, y la salida de datos mediante sonido y emisión de luz, fortaleciendo el aprendizaje del código Morse. Esta integración de estímulos visuales y auditivos facilitará la comprensión y memorización del código Morse, creando una experiencia más completa y enriquecedora para el usuario.

### **FASE III: MARCO TEÓRICO (CONCEPTUAL)**

#### **3.1. ¿QUÉ ES EL CÓDIGO MORSE?**

El código Morse es un medio de comunicación basado en la transmisión y recepción de mensajes empleando sonidos o rayos de luz y un alfabeto compuesto por puntos y rayas. Aunque este código surgió en el siglo XIX su empleo es perfectamente viable hoy en día cuando la existencia de condiciones adversas no permite el empleo de otros medios más desarrollados como, por ejemplo, la transmisión por voz (Preciado S.E. 2025).

El código Morse fue desarrollado para usar el telégrafo, un dispositivo ideado también por Samuel Morse en 1832. Era un sistema de comunicación que permite transmitir mensajes a larga distancia mediante la codificación y envío de señales, generalmente en forma de pulsos eléctricos a través de cables. Emplea un dispositivo llamado "transmisor" que convierte los mensajes en señales eléctricas, y un "receptor" que recupera las señales y las traduce nuevamente en mensajes legibles (National Geographic 2023).

#### **3.2. FUNCIONALIDAD**

El código Morse está formado por puntos, rayas y espacios que, una vez combinados, forman palabras, números y símbolos.

La forma de emplearse se realiza a través de vibraciones, donde cada letra viaja a través de un cable telegráfico el cual se transforma en impulsos eléctricos mediante un pulsador o sensor que activa el circuito.

Es importante conocer las siguientes reglas para comprender el funcionamiento del Código Morse (Antena 3 Noticias 2024):

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

- Cada letra, número y símbolo se representa con una secuencia única de puntos y rayas.
- Los puntos son señales cortas y las rayas son señales largas.
- El punto es considerado como una unidad o  $1/25$  segundos.
- La duración de un punto es la mínima posible, mientras que una raya dura aproximadamente tres veces más. La línea equivale a tres puntos.
- Entre cada par de símbolos de una misma letra hay una pausa de aproximadamente un punto.
- Entre las letras de una misma palabra hay una pausa de aproximadamente tres puntos.
- Para separar palabras, se utiliza una pausa de aproximadamente tres veces la duración de una raya.
- Entre un impulso y otro impulso existe una unidad de tiempo.

### 3.3. CÓDIGO MORSE

A continuación, se lista el código Morse de cada letra del alfabeto y algunos caracteres especiales:

A	· —	N	— ·	1	· — — — —	?	· · — — —
B	— · · ·	O	— — —	2	· · — — —	!	— · — · — —
C	— · — ·	P	· — — ·	3	· · · — —	.	· — · — · —
D	— · ·	Q	— — · —	4	· · · · —	,	— — — · — —
E	·	R	· — ·	5	· · · · ·	;	— · — · — ·
F	· · — ·	S	· · ·	6	— · · · ·	:	— — — · · ·
G	— — ·	T	—	7	— — — · ·	+	· — · — ·
H	· · · ·	U	· · —	8	— — — · ·	-	— · · · · —
I	· ·	V	· · · —	9	— — — — ·	/	— · — · —
J	· — — — —	W	· — — —	0	— — — — —	=	— · · · —
K	— · —	X	— · · —				
L	· — · ·	Y	— · — —				
M	— —	Z	— — · ·				

*Ilustración 1 Código Morse.  
Fuente: Buenos Aires Radio Club.*

En la elaboración de este proyecto se utilizó el diccionario del código Morse anteriormente mencionado (Ilustración 1 Código Morse).

Fuente: Buenos Aires Radio Club.

### 3.4. ÁREAS DONDE SE UTILIZA

Aunque en la actualidad el código Morse no es tan utilizado para una comunicación cotidiana y de la vida diaria, todavía sigue siendo empleado por diversas personas, grupos e instituciones en situaciones específicas y de emergencias. Entre algunas de estas áreas se encuentran:

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

- **Entrenamiento y educación:** Sirve como herramienta educativa en el aprendizaje de habilidades de concentración y memoria (Ambientum 2024).
- **Aeronáutica y navegación marítima:** En la aeronáutica las estaciones como VOR (Very High Frequency Omnidirectional Range) e ILS (Instrument Landing System) transmiten su identificador en código Morse, que los pilotos pueden utilizar para determinar su posición y ruta a seguir y en otros casos, guiando su trayectoria hasta la pista de aterrizaje (Aero Museo 2023).
- **Servicios de emergencia:** El código Morse demostró ser vital en situaciones de emergencia. Los navegantes y pilotos lo utilizan para transmitir señales de socorro cuando otros métodos de comunicación fallan. La señal de socorro más conocida, “SOS” (···---···), es reconocida internacionalmente y puede ser transmitida fácilmente con cualquier dispositivo que emita luz o sonido (La Nación 2024).
- **Radioafición:** El código Morse sigue siendo una habilidad esencial para los radioaficionados. Facilita la comunicación global, permitiendo a los aficionados conectarse con otros en todo el mundo. A pesar de la disponibilidad de los medios digitales modernos, muchos radioaficionados siguen utilizando el código Morse como un medio de comunicación fiable y eficiente (AlmondSolutions 2023).
- **Uso militar:** Su capacidad para enviar mensajes cifrados y su relativa simplicidad lo hace ideal para la transmisión de órdenes y reportes en el campo de batalla. Incluso, hoy en día, el código Morse sigue siendo parte del entrenamiento de las fuerzas armadas de algunos países.

## FASE IV: DESARROLLO

### 4.1. ¿PARA QUIÉN ESTÁ DIRIGIDO?

- **Educadores:** Integrar esta forma de comunicación en algunos niveles educativos.
- **Personas con discapacidades:** contribuir para que las personas con facultades en el habla logren comunicarse de una manera más efectiva.
- **Personal de los servicios de emergencias:** cuando las comunicaciones convencionales tienden a fallar.
- **Personal de seguridad (Ejercito, Secretaría de la Defensa Nacional, etc.):** para mantener en confidencialidad la información en casos de guerras.

### 4.2. DESCRIPCIÓN

Para el desarrollo de este proyecto se implementó mediante programación un traductor en el cual desde un equipo de cómputo se introduzca alguna palabra, frase o un mensaje más específico dando como resultado la traducción a código Morse mostrándolo en la pantalla (del equipo de cómputo), y en medio audiovisual, es decir, la traducción se escuchará en pitidos (sonidos) y se emitirá una luz la cual parpadeará mientras se reproduce el sonido.

Para la traducción de código Morse a lenguaje natural, se pretende ingresar el mensaje en código Morse mostrándose en pantalla las líneas del código Morse, para que posteriormente se traduzca y lograr apreciar la traducción en forma de texto.

Para el correcto funcionamiento del prototipo que incluye este proyecto, se trabajó en dos lenguajes de programación; Arduino y Python cada uno con sus propias funciones.

Comunicación en código Morse: Un sistema visual y auditivo en Programación.

La variante del lenguaje C++ Arduino, es el lenguaje que se encarga del funcionamiento de la placa Arduino UNO y que, por ende, es donde queda cargado o guardado el diccionario con cada letra del alfabeto, pero con sus respectivas representaciones en código Morse.

Por otro lado, en Python es donde se realiza el proceso de traducción tomando desde Arduino el diccionario en código Morse para que al ser introducidos los mensajes sea de forma automática su traducción. Así mismo, permite realizar las gráficas de los tiempos al momento de enviar y recibir un mensaje por el puerto serial mediante una interfaz, permitiendo al usuario hacer uso del pulsador para ingresar el mensaje y que este se pueda ver en pantalla.

### **4.3. DISEÑO DEL DISPOSITIVO**

Durante la etapa de del diseño del dispositivo se generaron dos diagramas, un diagrama de flujo que indica el funcionamiento del traductor desde la activación del puerto serial y la conexión de la interfaz programada en Python con la tarjeta Arduino UNO.

El segundo representa el funcionamiento del dispositivo mediante un diagrama de bloques.

### **4.4. RECURSOS Y MATERIALES NECESARIOS**

Para lograr el correcto funcionamiento del prototipo de este proyecto se utilizaron diversos materiales los cuales, se enlistan a continuación: (Para consultar sus características ver Anexo A).

- 1 equipo de cómputo.
- 1 foco Base E12.
- 1 socket Base E12.

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

- 1 Resistencia genérica.
- 1 Resistencia de cerámica.
- 1 TRIAC BTA08-600B.
- 1 MOC 3021.
- 1 push-button (pulsador).
- 1 buzzer (Zumbador).
- Arduino UNO.
- Cable Estañado.

### 4.3.1 COSTOS Y GANANCIAS

- 1 foco Base E12 **\$27**
- 1 socket Base E12. **\$52**
- 1 Resistencia genérica 1 k $\Omega$ . **\$2**
- 1 Resistencia de cerámica de 220  $\Omega$  a 10 watts. **\$19**
- 1 TRIAC BTA08-600B **\$35**
- 1 MOC 3021. **\$68**
- 1 push-button (pulsador). **\$26**
- 1 buzzer (Zumbador). **\$45**
- Cable Estañado. **\$22**

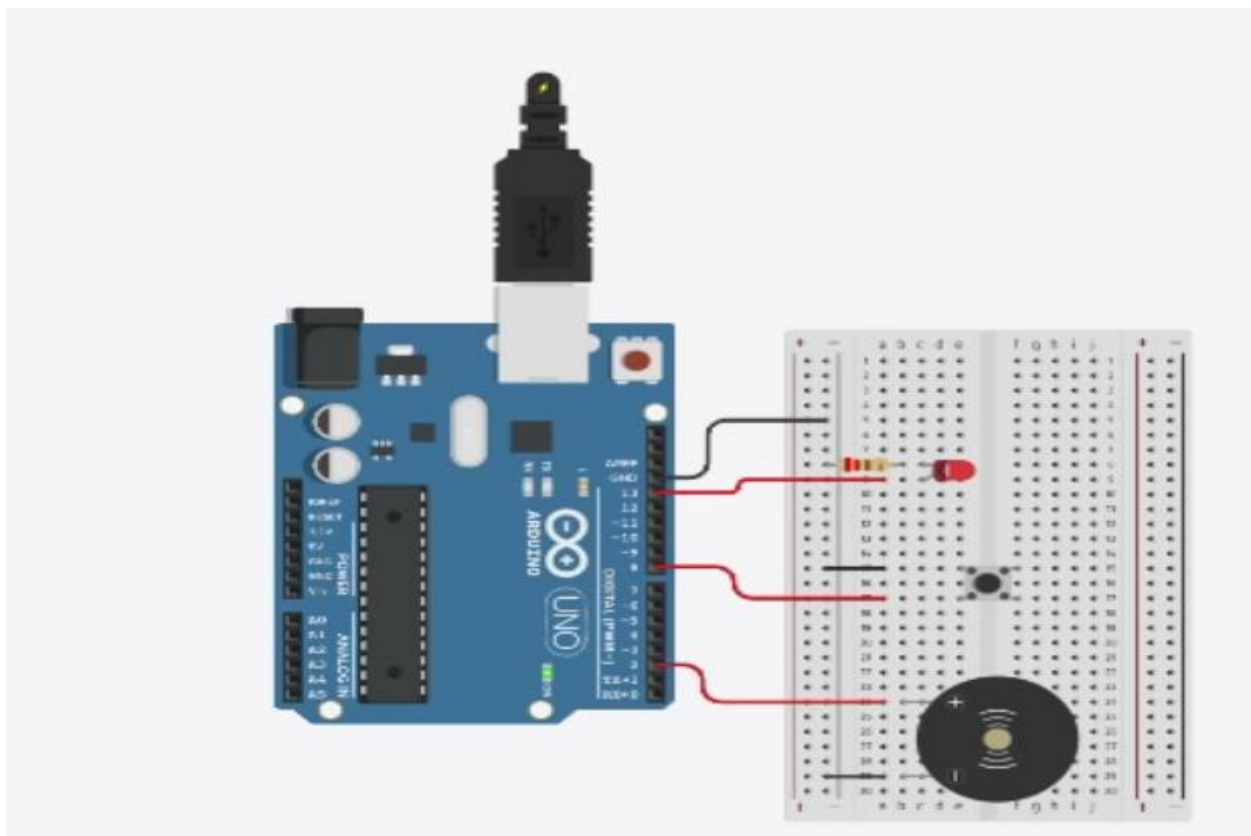


- Arduino UNO. \$294

$$\text{Precio} = \frac{\text{Costos y/o Gastos}}{1 - \% \text{Ganancia}} = \frac{\$590}{1 - 60\%} = \$1475.00$$

#### 4.5. SIMULACIÓN

Posterior a la fase del diseño del dispositivo, se experimentó utilizando el programa de simulación de la tarjeta de Arduino UNO para comprobar que el diseño del circuito funcionara correctamente. A continuación, se presenta la imagen de la configuración de la simulación, en ella se pueden observar los elementos conectados para las primeras pruebas.



*Ilustración 2 Simulación de primeras pruebas.*

## **FASE V: CONCLUSIÓN**

### **5.1. CONCLUSIONES**

Este proyecto facilita el aprendizaje del código Morse, en el cual por medio de la Programación se logró crear un traductor que permita traducir letras, palabras e incluso oraciones de lenguaje natural al código Morse y viceversa, mismo que durante su desarrollo nos permitió a nosotros como estudiantes interactuar con diferentes Softwares de aplicación como lo fue Arduino que como tal no es un lenguaje de programación sino que es una variante de C++ y Python mismo que es el encargado de hacer todo el proceso de traducción y de igual forma de mostrar las graficas de los tiempos que tardan en ser traducidos los mensajes.

Con este proyecto una vez mas nos pudimos dar cuenta que nada es tan sencillo como parece ya que aparte de que teníamos que lograr la traducción, se tenia que hacer notorio el código Morse tanto visual como auditivo y para que esto fuera posible se tuvo que diseñar un circuito de tal manera que por medio de un foco se apreciara una luz al ingresar el código Morse y un zumbador para poder ser escuchado.

Una de las principales experiencias que tuvimos al desarrollar este proyecto fue que trabajar en equipo facilita el desarrollo y por ende la ejecución de un proyecto, sin embargo, si una o varias partes de las involucradas no tienen iniciativa de continuar, no hay ningún motivo para detener el proyecto porque como se dice así sea uno contra el mundo entero, una mentalidad fuerte podrá con eso y más.

## **5.2. RECOMENDACIONES**

Se percibe que las acciones inmediatas para robustecer este dispositivo sean:

- a) Mejorar la interfaz gráfica de usuario.
- b) Generar un manual práctico para el aprendizaje del código Morse.

Posteriormente, para incrementar el impacto del presente proyecto se recomienda diseñar la comunicación entre dos dispositivos para facilitar el envío y recepción de mensajes codificados a largas distancias.

## ANEXO A: MATERIALES UTILIZADOS

**FOCO BASE E12:** Se utilizó un tipo de bombilla con un casquillo de rosca Edison y un diámetro de 12 milímetros. La designación E12 significa que es parte de la familia de bombillas Edison, con la “E” que representa a Edison y el “12” que indica el diámetro de la base de tornillo en milímetros, con una potencia de 7.5W, tensión de 130V y una frecuencia de operación de 50/60Hz. (RAYZEEK).



*Ilustración 3 Foco Base E12.  
Fuente: LUMMI*

**SOCKET BASE E12:** Dispositivo que conecta físicamente el foco o bombilla a una fuente de energía eléctrica. Su función principal es proporcionar un soporte seguro para el foco y facilitar el flujo de electricidad para que la luz funcione correctamente. También se le conoce

Comunicación en código Morse: Un sistema visual y auditivo en Programación.

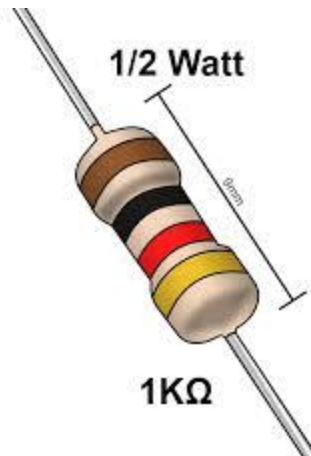
como casquillo eléctrico o zócalo, con una capacidad de 127V a 630W. (Elektron).



*Ilustración 4 Socket Base E12.  
Fuente: Sodimac.*

**RESISTENCIA GENÉRICA:** La resistencia utilizada es de  $1k\Omega$ . Datasheet:

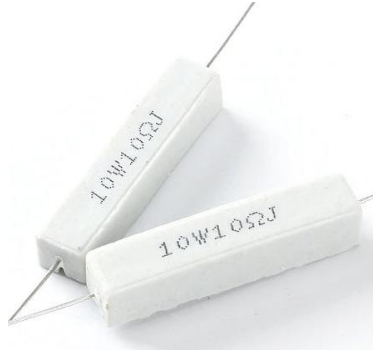
<https://www.alldatasheet.es/datasheet-pdf/view/232953/ETC2/30BJ250-1K.html>



*Ilustración 5 Resistencia Genérica 1KΩ.  
Fuente: MV ELECTRÓNICA.*

Comunicación en código Morse: Un sistema visual y auditivo en Programación.

**RESISTENCIA DE CERÁMICA:** La resistencia utilizada en este proyecto es de  $220\Omega$  a 10watts, que a diferencia de las resistencias genéricas es que generalmente se emplean en aplicaciones que requieren una disipación de potencia elevada. También se pueden usar como elemento calefactor. (SUMADOR).



*Ilustración 6 Resistencia de Cerámica.  
Fuente: Sumador.*

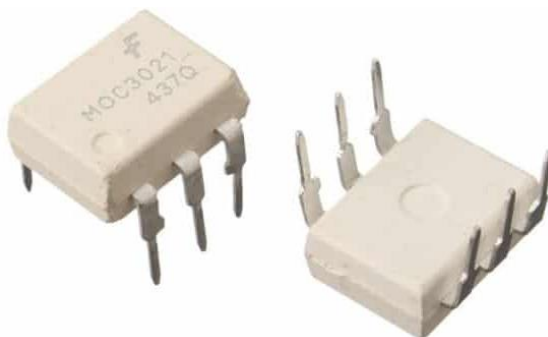
**TRIAC:** El TRIAC actúa como un interruptor de estado solido para corriente alterna permitiendo o bloqueando el flujo de corriente. (AREATECNOLOGIA). El TRIAC utilizado es un BTA08-600B. Datasheet: <https://www.allDatasheet.es/Datasheet-pdf/view/22031/STMICROELECTRONICS/BTA08-600B.html>



*Ilustración 7 TRIAC BTA08-600B.  
Fuente: UNIT Electronics.*

Comunicación en código Morse: Un sistema visual y auditivo en Programación.

**MOC:** Se utiliza en aislamiento eléctrico en diferentes partes de un sistema y proteger contra picos de voltaje y corriente. En corriente de cargas para controlar el encendido y apagado de cargas conectadas a un circuito. (UNIT Electronics 2016). El MOC3021 fue el que se utilizó en el circuito de este proyecto. Datasheet: <https://www.allDatasheet.es/Datasheet-pdf/view/53870/FAIRCHILD/MOC3021.html>



*Ilustración 8 MOC3021.  
Fuente: UNIT Electronics 2016.*

**PUSH-BUTTON (PULSADOR):** Los interruptores pulsadores sirven como dispositivos básicos de interfaz de usuario, así como sensores simples. Normalmente aparece como un circuito abierto y cambian a corto circuito cuando se presionan. (UPV/EHU). El pulsador utilizado fue de uso común.



*Ilustración 9 Push-button.  
Fuente: Electrónica Valtierra.*

Comunicación en código Morse: Un sistema visual y auditivo en Programación.

**BUZZER (ZUMBADOR):** Es un dispositivo que emite un sonido, ya sea continuo o intermitente, al recibir una señal eléctrica. (OSAKA Electronics).



*Ilustración 10 Buzzer:  
Fuente: Hardware Libre.*

**ARDUINO UNO:** Es la placa de desarrollo emblemática de Arduino Corporation y el modelo de referencia para herramientas de creación de prototipos para la educación y para creadores que necesiten un microcontrolador fácil de usar. (Arrow). Datasheet:

<https://docs.arduino.cc/resources/Datasheets/A000066-Datasheet.pdf>



*Ilustración 11 Arduino UNO.  
Fuente: Arduino Docs.*



Comunicación en código Morse: Un sistema visual y auditivo en Programación.

**CABLE ESTAÑADO:** Es un conductor eléctrico que esta recubierto de una fina capa de estaño para proteger el cobre de la corrosión. Es más fácil de soldar que el alambre de cobre no estañado. (ZW Cables).



*Ilustración 12 Cable Estañado.  
Fuente: Mercado Libre.*

## ANEXO B: DIAGRAMAS DEL DISPOSITIVO

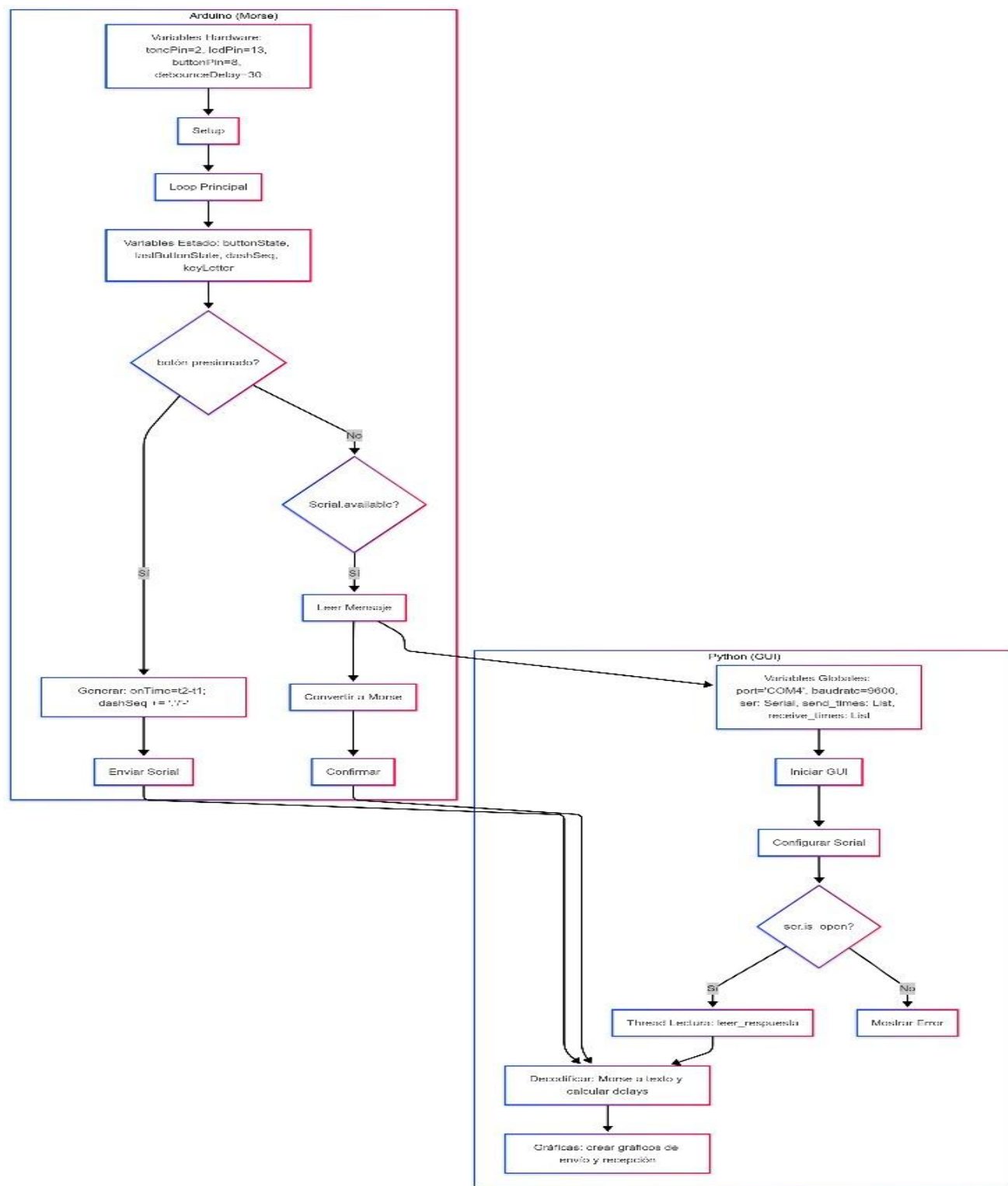
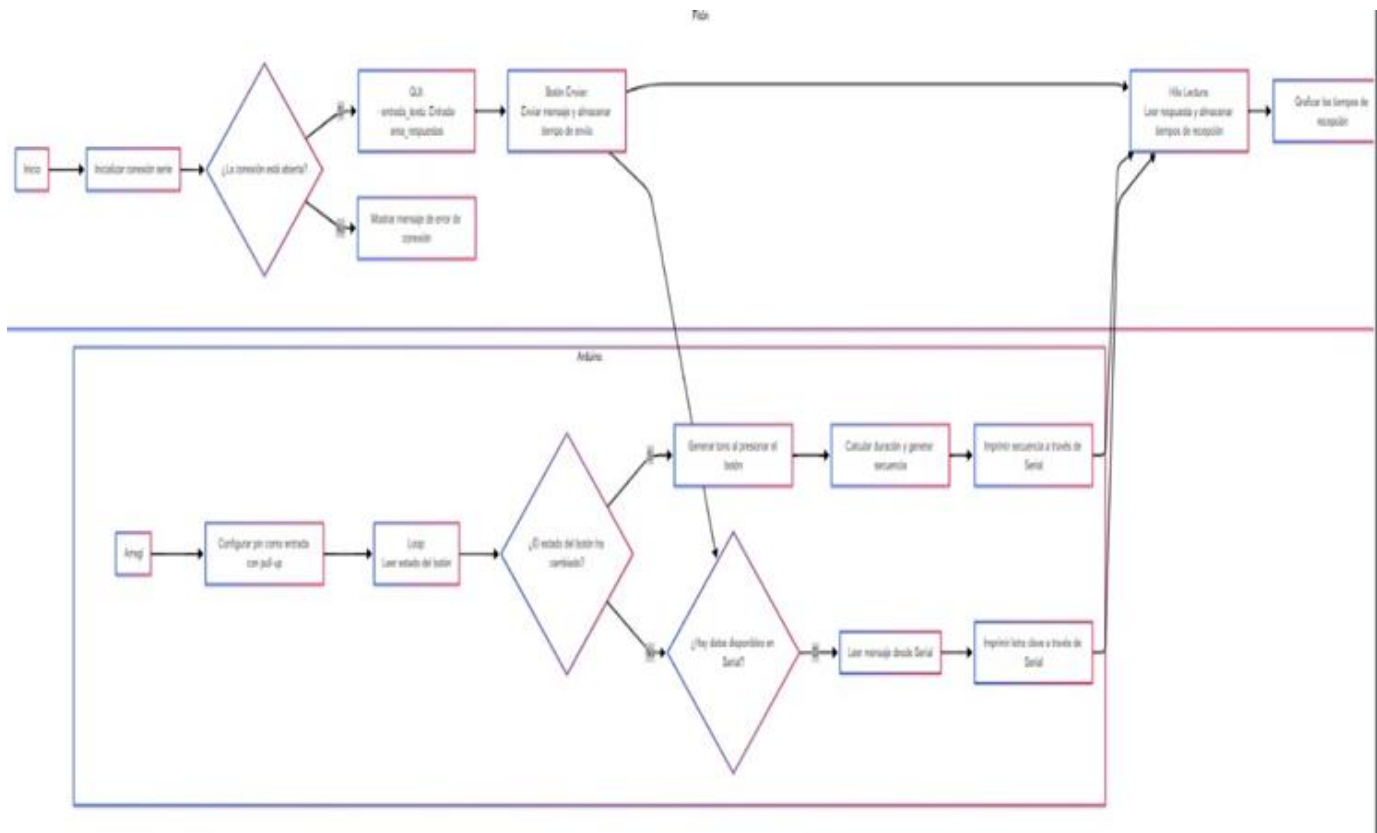


Ilustración 13 Diagrama de Flujo

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.



*Ilustración 14 Diagrama de Bloques*

## ANEXO C: CÓDIGOS DE LA PROGRAMACIÓN DEL DISPOSITIVO

### CÓDIGO EN ARDUINO.

```
int tonePin = 2;
```

```
int toneFreq = 1000;
```

```
int ledPin = 13;
```

```
int buttonPin = 8;
```

```
int debounceDelay = 30;
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```

int dotLength = 240;

int dotSpace = dotLength;

int dashLength = dotLength * 3;

int letterSpace = dotLength * 3;

int wordSpace = dotLength * 7;

float wpm = 1200. / dotLength;


int t1, t2, onTime, gap;

bool newLetter, newWord, letterFound, keyboardText;

int lineLength = 0;

int maxLineLength = 20;


char* letters[] = {

    ".-", "-...", "-.-.", "-..", ".", "...-", "--.", "....", "..", // A-I

    ".---", "-.-", "-..", "--", "-.", "---", ".---", "--.-", ".-.", // J-R

    "...", "-", "-.-", "-.-.", "-.-", "-.-", "-.-", "-.-." // S-Z

};

```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```

char* numbers[] = {

    "-----", ".----", "..---", "...--", "....-", // 0-4

    ".....", "-....", "--...", "---..", "----." // 5-9

};

String dashSeq = "";

char keyLetter, ch;

int i, index;

bool buttonState = HIGH; // Estado actual del botón

bool lastButtonState = HIGH; // Estado anterior del botón

void setup() {

    delay(500);

    pinMode(ledPin, OUTPUT);

    pinMode(tonePin, OUTPUT);

    pinMode(buttonPin, INPUT_PULLUP);

    Serial.begin(9600);

}

```

```
void loop() {  
  
    // Leer el estado del botón  
  
    buttonState = digitalRead(buttonPin);  
  
    // Detectar el flanco de bajada (botón presionado)  
  
    if (buttonState == LOW && lastButtonState == HIGH) {  
  
        newLetter = true;  
  
        newWord = true;  
  
        t1 = millis();  
  
        digitalWrite(ledPin, HIGH);  
  
        tone(tonePin, toneFreq);  
  
        delay(debounceDelay);  
  
        while (digitalRead(buttonPin) == LOW) {  
  
            delay(debounceDelay);  
  
        }  
  
        delay(debounceDelay);  
  
  
        t2 = millis();  
  
        onTime = t2 - t1;  
    }  
}
```

```
digitalWrite(ledPin, LOW);

noTone(tonePin);

if (onTime <= dotLength * 1.5) {

    dashSeq += ".";

    Serial.print("."); // Enviar punto a Python

} else {

    dashSeq += "-";

    Serial.print("-"); // Enviar raya a Python

}

Serial.print(","); // Separador

Serial.print(millis()); // Enviar el tiempo actual en milisegundos

Serial.println(",ms"); // Indicar que el tiempo está en milisegundos

}

// Actualizar el estado anterior del botón

lastButtonState = buttonState;

gap = millis() - t2;
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```
if (newLetter == true && gap >= letterSpace) {  
  
    letterFound = false;  
  
    keyLetter = '?'; // Carácter '?' si no se encuentra  
  
    for (i = 0; i < 26; i++) {  
  
        if (dashSeq == letters[i]) {  
  
            keyLetter = i + 'A';  
  
            letterFound = true;  
  
            break;  
  
        }  
  
    }  
  
    if (!letterFound) {  
  
        for (i = 0; i < 10; i++) {  
  
            if (dashSeq == numbers[i]) {  
  
                keyLetter = i + '0';  
  
                letterFound = true;  
  
                break;  
  
            }  
  
        }  
  
    }  
}
```



```
    }

    if (letterFound) {

        Serial.print(keyLetter); // Enviar letra decodificada a Python

        Serial.print(","); // Separador

        Serial.print(millis()); // Enviar el tiempo actual en milisegundos

        Serial.println(",ms"); // Indicar que el tiempo está en milisegundos

    }

    newLetter = false;

    dashSeq = "";

    lineLength++;

}

if (newWord == true && gap >= wordSpace * 1.5) {

    newWord = false;

    lineLength++;

    digitalWrite(ledPin, HIGH);

    delay(25);

    digitalWrite(ledPin, LOW);
```

```
}
```

```
if (lineLength >= maxLineLength) {
```

```
    Serial.println();
```

```
    lineLength = 0;
```

```
}
```

```
// Leer mensajes desde Python
```

```
if (Serial.available() > 0) {
```

```
    String message = Serial.readStringUntil('\n'); // Leer el mensaje
```

```
    message.toUpperCase(); // Convertir a mayúsculas
```

```
    for (int i = 0; i < message.length(); i++) {
```

```
        char c = message[i];
```

```
        if (c >= 'A' && c <= 'Z') {
```

```
            flashSequence(letters[c - 'A']); // Convertir letra a Morse
```

```
            Serial.print(c); // Enviar la letra decodificada
```

```
            Serial.print(","); // Separador
```

```
            Serial.print(millis()); // Enviar el tiempo actual en milisegundos
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```

        Serial.println("ms"); // Indicar que el tiempo está en milisegundos

    } else if (c >= '0' && c <= '9') {

        flashSequence(numbers[c - '0']); // Convertir número a Morse

        Serial.print(c); // Enviar el número decodificado

        Serial.print(","); // Separador

        Serial.print(millis()); // Enviar el tiempo actual en milisegundos

        Serial.println("ms"); // Indicar que el tiempo está en milisegundos

    } else if (c == ' ') {

        delay(wordSpace); // Espacio entre palabras

    }

}

}

}

}

```

```

void flashSequence(char* sequence) {

    int i = 0;

    while (sequence[i] == '.' || sequence[i] == '-') {

        flashDotOrDash(sequence[i]);

```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```

        i++;

    }

    delay(letterSpace); // Espacio entre letras

}

void flashDotOrDash(char dotOrDash) {

    digitalWrite(ledPin, HIGH);

    tone(tonePin, toneFreq);

    if (dotOrDash == '.') {

        delay(dotLength);

    } else {

        delay(dashLength);

    }

    digitalWrite(ledPin, LOW);

    noTone(tonePin);

    delay(dotLength); // Espacio entre puntos/rayas

}

```

### **CÓDIGO EN PYTHON**

```
import serial # type: ignore
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```
import time

import tkinter as tk

from tkinter import scrolledtext, messagebox

from threading import Thread

import matplotlib.pyplot as plt # type: ignore

import sys


# Configuración serial

port = 'COM4' # Cambiar al puerto correcto

baudrate = 9600


# Inicializa la conexión serial

ser = None

try:

    ser = serial.Serial(port, baudrate, timeout=1)

    time.sleep(2) # Espera a que la conexión se establezca

except serial.SerialException as e:

    print(f'Error al abrir el puerto serial: {e}', file=sys.stderr)
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```

    ser = None

except Exception as e:

    print(f'Error inesperado: {e}', file=sys.stderr)

    ser = None


# Variables globales

send_times = [] # Tiempos de envío (mensaje, tiempo)

receive_times = [] # Tiempos de recepción


# Mensajes del sistema

MENSAJES = {

    "error_generico": "Error: Mensaje no reconocido. Vuelva a intentar.",

    "error_formato": "Error: Formato de mensaje incorrecto. Se esperaba
'mensaje,tiempo,unidad'",

    "envio_exitoso": "Mensaje enviado exitosamente",

    "conexion_exitosa": "Conexión establecida con Arduino",

    "conexion_fallida": "Error en la conexión con Arduino",

    "grafica_no_datos": "No hay suficientes datos para graficar. Envíe y reciba más
mensajes primero."

```

```
}

# Configuración de colores

AZUL = '#1f77b4'

NARANJA = '#ff7f0e'

GRIS_CLARO = '#f0f0f0'

BLANCO = 'white'

class AplicacionSerial:

    def __init__(self, root):

        self.root = root

        self.root.title("Sistema de Comunicación Serial")

        self.root.configure(bg=GRIS_CLARO)

        # Crear frames para cada pantalla

        self.ventana_principal = tk.Frame(self.root, bg=GRIS_CLARO)

        self.ventana_menu = tk.Frame(self.root, bg=GRIS_CLARO)

        self.ventana_comunicacion = tk.Frame(self.root, bg=GRIS_CLARO)
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```
self.crear_ventana_principal()

self.crear_ventana_menu()

self.crear_ventana_comunicacion()


# Mostrar ventana principal al inicio

self.mostrar_ventana_principal()


# Iniciar hilo para lectura serial

if ser and ser.is_open:

    self.mostrar_respuesta("=== Sistema de Comunicación Serial ===\n", "blue")

    self.mostrar_respuesta(">> Listo para enviar y recibir mensajes\n\n", "blue")

    self.mostrar_estado_conexion(True)

    self.thread_lectura = Thread(target=self.leer_respuesta_automaticamente,
daemon=True)

    self.thread_lectura.start()


def mostrar_ventana_principal(self):

    self.ocultar_todas_ventanas()

    self.ventana_principal.pack(padx=10, pady=10)
```



```
def mostrar_ventana_menu(self):

    self.ocultar_todas_ventanas()

    self.ventana_menu.pack(padx=10, pady=10)


def mostrar_ventana_comunicacion(self):

    self.ocultar_todas_ventanas()

    self.ventana_comunicacion.pack(padx=10, pady=10)


def ocultar_todas_ventanas(self):

    for ventana in [self.ventana_principal, self.ventana_menu,
self.ventana_comunicacion]:

        ventana.pack_forget()


def crear_ventana_principal(self):

    # Ventana de bienvenida

    tk.Label(

        self.ventana_principal,

        text="Bienvenido al Sistema de Comunicación Serial",
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```
font=("Arial", 16),

bg=GRIS_CLARO

).pack(pady=20)


# Botones

frame_botones = tk.Frame(self.ventana_principal, bg=GRIS_CLARO)

frame_botones.pack(pady=20)


tk.Button(

    frame_botones,

    text="Menú Principal",

    command=self.mostrar_ventana_menu,

    bg=AZUL,

    fg=BLANCO,

    font=('Arial', 12, 'bold'),

    padx=20,

    pady=10

).pack(side=tk.LEFT, padx=20)
```

```
tk.Button(  
  
    frame_botones,  
  
    text="Salir",  
  
    command=self.cerrar_aplicacion,  
  
    bg=NARANJA,  
  
    fg=BLANCO,  
  
    font=('Arial', 12, 'bold'),  
  
    padx=20,  
  
    pady=10  
  
).pack(side=tk.RIGHT, padx=20)
```

```
def crear_ventana_menu(self):
```

```
    # Menú principal
```

```
tk.Label(  
  
    self.ventana_menu,  
  
    text="Menú Principal",  
  
    font=("Arial", 16),
```

```
        bg=GRIS_CLARO

    ).pack(pady=20)

# Botones del menú

frame_botones = tk.Frame(self.ventana_menu, bg=GRIS_CLARO)

frame_botones.pack(pady=10)

tk.Button(

    frame_botones,

    text="Comunicación Serial",

    command=self.mostrar_ventana_comunicacion,

    bg=AZUL,

    fg=BLANCO,

    font=('Arial', 12),

    width=20,

    pady=10

).pack(pady=10)
```

```
tk.Button(  
  
    frame_botones,  
  
    text="Gráficas",  
  
    command=self.mostrar_grafica,  
  
    bg=AZUL,  
  
    fg=BLANCO,  
  
    font=('Arial', 12),  
  
    width=20,  
  
    pady=10  
).pack(pady=10)  
  
tk.Button(  
  
    frame_botones,  
  
    text="Regresar",  
  
    command=self.mostrar_ventana_principal,  
  
    bg=NARANJA,  
  
    fg=BLANCO,  
  
    font=('Arial', 12),
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```
        width=20,

        pady=10

    ).pack(pady=10)


def crear_ventana_comunicacion(self):

    # Frame principal

    frame_principal = tk.Frame(self.ventana_comunicacion, bg=GRIS_CLARO)

    frame_principal.pack(padx=10, pady=10)


    # Campo de entrada para enviar mensajes

    self.entrada_texto = tk.Entry(

        frame_principal,

        width=50,

        font=('Arial', 11),

        bg='white',

        fg='black'

    )

    self.entrada_texto.grid(row=0, column=0, padx=5, pady=5)
```

```
# Botón para enviar mensajes
```

```
tk.Button(  
  
    frame_principal,  
  
    text="Enviar",  
  
    command=self.enviar_mensaje,  
  
    bg=NARANJA,  
  
    fg=BLANCO,  
  
    font=('Arial', 10, 'bold'),  
  
    relief=tk.RAISED,  
  
    borderwidth=2  
  
).grid(row=0, column=1, padx=5, pady=5)
```

```
# Área de texto para mostrar respuestas
```

```
self.area_respuestas = scrolledtext.ScrolledText(  
  
    frame_principal,  
  
    width=70,  
  
    height=20,
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```

        font=('Consolas', 10),

        wrap=tk.WORD,

        bg='white',

        fg='black'

    )

    self.area_respuestas.grid(row=1, column=0, columnspan=2, padx=5, pady=5)

    self.area_respuestas.config(state=tk.DISABLED)


# Frame para botones inferiores

frame_botones = tk.Frame(frame_principal, bg=GRIS_CLARO)

frame_botones.grid(row=2, column=0, columnspan=2, pady=5)


# Botones adicionales

tk.Button(

    frame_botones,

    text="Graficar",

    command=self.mostrar_grafica,

    bg=AZUL,

```



## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```
fg=BLANCO,  
  
font=('Arial', 9, 'bold')  
  
) .pack(side=tk.LEFT, padx=5)  
  
tk.Button(  
  
    frame_botones,  
  
    text="Limpiar",  
  
    command=self.limpiar_consola,  
  
    bg=AZUL,  
  
    fg=BLANCO,  
  
    font=('Arial', 9, 'bold')  
  
) .pack(side=tk.LEFT, padx=5)  
  
tk.Button(  
  
    frame_botones,  
  
    text="Regresar",  
  
    command=self.mostrar_ventana_menu,  
  
    bg=NARANJA,
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```
fg=BLANCO,

font=('Arial', 9, 'bold')

).pack(side=tk.RIGHT, padx=5)


def mostrar_estado_conexion(self, exito):

    estado = MENSAJES["conexion_exitosa"] if exito else
MENSAJES["conexion_fallida"]

    self.mostrar_respuesta(f"Estado: {estado}\n")


def enviar_mensaje(self):

    message = self.entrada_texto.get().strip()

    if message:

        try:

            send_time = time.time()

            ser.write((message + "\n").encode())

            send_times.append((message, send_time))

            self.entrada_texto.delete(0, tk.END)

            self.mostrar_respuesta(f"[ENVIADO] {message}\n", "blue")
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```
        self.mostrar_respuesta(f' > {MENSAJES['envio_exitoso']} a las
{time.strftime("%H:%M:%S')}\n", "blue")

    except serial.SerialException as e:

        messagebox.showerror("Error de envío", f"No se pudo enviar el mensaje: {e}")

    else:

        messagebox.showwarning("Campo vacío", "Por favor ingrese un mensaje antes
de enviar")

def leer_respuesta_automaticamente(self):

    while ser and ser.is_open:

        try:

            if ser.in_waiting > 0:

                response = ser.readline().decode().strip()

                receive_time = time.time()

                if not response:

                    continue

                if response == "ERROR":
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```
self.mostrar_respuesta(f"[ERROR] {MENSAJES['error_generico']}\n",
"red")

continue

if "," in response:

    parts = response.split(",")

    if len(parts) == 3:

        message, send_time_ms, unit = parts

        if unit == "ms":

            try:

                send_time = float(send_time_ms) / 1000

                # Buscar el mensaje correspondiente

                for i, (sent_msg, sent_time) in enumerate(send_times):

                    if sent_msg == message:

                        del send_times[i]

                        break

                receive_times.append((send_time, receive_time))
```

Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```

        self.mostrar_respuesta(f"[RECIBIDO] {message}\n", "orange")

        self.mostrar_respuesta(f" > Tiempo de procesamiento:

{send_time_ms} ms\n", "orange")

        self.mostrar_respuesta(f" > Recibido a las

{time.strftime('%H:%M:%S')}\n", "orange")

    except ValueError:

        self.mostrar_respuesta(f"[ERROR]

{MENSAJES['error_formato']}: {response}\n", "red")

    else:

        self.mostrar_respuesta(f"[ERROR] Unidad de tiempo no reconocida:

{unit}\n", "red")

    else:

        self.mostrar_respuesta(f"[ERROR] {MENSAJES['error_formato']}:

{response}\n", "red")

    else:

        self.mostrar_respuesta(f"[RECIBIDO] {response}\n", "orange")

        self.mostrar_respuesta(f" > Recibido a las

{time.strftime('%H:%M:%S')}\n", "orange")

    except UnicodeDecodeError:
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```

        self.mostrar_respuesta("[ERROR] No se pudo decodificar el mensaje
recibido\n", "red")

    except Exception as e:

        self.mostrar_respuesta(f"[ERROR] Error inesperado: {str(e)}\n", "red")

        break

def mostrar_respuesta(self, texto, color="black"):

    self.area_respuestas.config(state=tk.NORMAL)

    self.area_respuestas.tag_config("blue", foreground="blue")

    self.area_respuestas.tag_config("orange", foreground=NARANJA)

    self.area_respuestas.tag_config("red", foreground="red")

    self.area_respuestas.insert(tk.END, texto, color)

    self.area_respuestas.config(state=tk.DISABLED)

    self.area_respuestas.see(tk.END)

def mostrar_grafica(self):

    if len(receive_times) < 2:

        messagebox.showwarning("Datos insuficientes",

MENSAJES["grafica_no_datos"])

```

```
return
```

```
send_times_plot = [t[0] for t in receive_times]
```

```
receive_times_plot = [t[1] for t in receive_times]
```

```
delays = [recv - send for send, recv in receive_times]
```

```
plt.figure(figsize=(12, 5))
```

```
# Gráfico 1: Tiempos absolutos
```

```
plt.subplot(1, 2, 1)
```

```
plt.plot(send_times_plot, receive_times_plot, 'o-', color=AZUL)
```

```
plt.xlabel('Tiempo de envío (s)')
```

```
plt.ylabel('Tiempo de recepción (s)')
```

```
plt.title('Tiempos absolutos de comunicación')
```

```
plt.grid(True)
```

```
# Gráfico 2: Retrasos
```

```
plt.subplot(1, 2, 2)
```

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

```
plt.plot(delays, '*-', color=NARANJA)
```

```
plt.xlabel('Número de mensaje')
```

```
plt.ylabel('Retraso (s)')
```

```
plt.title('Retrasos en la comunicación')
```

```
plt.grid(True)
```

```
plt.tight_layout()
```

```
plt.show()
```

```
def limpiar_consola(self):
```

```
    self.area_respuestas.config(state=tk.NORMAL)
```

```
    self.area_respuestas.delete(1.0, tk.END)
```

```
    self.area_respuestas.config(state=tk.DISABLED)
```

```
    self.mostrar_respuesta("-- Consola limpiada --\n", "blue")
```

```
    self.mostrar_respuesta(f"-- {time.strftime('%H:%M:%S')} --\n\n", "blue")
```

```
def cerrar_aplicacion(self):
```

```
    if messagebox.askyesno("Confirmar", "¿Está seguro que desea salir?"):
```



```
        try:

            if ser and ser.is_open:

                ser.close()

        except:

            pass

        finally:

            self.root.destroy()

if __name__ == "__main__":

    try:

        root = tk.Tk()

        app = AplicacionSerial(root)

        root.protocol("WM_DELETE_WINDOW", app.cerrar_aplicacion)

        root.mainloop()

    except Exception as e:

        print(f'Error en la aplicación: {e}', file=sys.stderr)

        if ser and ser.is_open:

            ser.close()
```

**ANEXO D: CÓDIGO QR DEL MANUAL DE USUARIO***Ilustración 15 Código QR del manual de usuario.***ANEXO E: CRONOGRAMA DE ACTIVIDADES**

Actividad	Periodo (semanas/meses/semestres)			
	Semana 1-4	Semana 5-8	Semana 9-12	Semana 13-14
Planificación				
Documentación				
Diseño del sistema				
Codificación del Sistema				
Ensamblaje del Dispositivo de Clave Morse				
Conexión de Ambos Sistemas				
Fase de pruebas				
Entregable				

**5.3. FUENTES DE***Tabla 1 Cronograma de Actividades* **CONSULTA**

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

- Aero Museo. (2023, septiembre 11). “El código Morse en la aviación actual”  
Consultado en: <https://aeromuseo.org/codigo-Morse-en-la-aviacion-actual/#:~:text=Cada%20emplazamiento%20tiene%20un%20c%C3%B3digo,tien en%20experiencia%20en%20la%20radiotelegraf%C3%ADa>
- ALLDATASHEET. (1989, febrero). “BTA08-600B Datasheet (PDF) – STMicroelectronics”. Consultado en: <https://www.alldatasheet.es/Datasheet-pdf/view/22031/STMICROELECTRONICS/BTA08-600B.html>
- ALLDATASHEET. (2003, abril 30). “MOC3021 Datasheet (PDF) - Fairchild Semiconductor”. Consultado en: <https://www.alldatasheet.es/Datasheet-pdf/view/53870/FAIRCHILD/MOC3021.html>
- ALLDATASHEET. (2007, octubre 18). “30BJ250-1K Datasheet (PDF) - List of Unclassified Manufacturers” Consultado en: <https://www.alldatasheet.es/Datasheet-pdf/view/232953/ETC2/30BJ250-1K.html>
- AlmondSolutions. (2023, diciembre 15). “¿Cuáles son los usos modernos del código Morse?”. Consultado en: [https://www-almondsolutions-com.translate.goog/blog/what-are-the-modern-uses-of-Morse-code?\\_x\\_tr\\_sl=en&\\_x\\_tr\\_tl=es&\\_x\\_tr\\_hl=es&\\_x\\_tr\\_pto=sge](https://www-almondsolutions-com.translate.goog/blog/what-are-the-modern-uses-of-Morse-code?_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=sge) [Traducción de inglés a español].
- Ambientum (2024, octubre 23). “¿Qué es el Código Morse y para qué sirve?” Consultado en: <https://www.ambientum.com/ambientum/curiosidades/que-es-el-codigo-Morse-y-para-que-sirve.asp>

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

- Antena 3 Noticias. (2024, mayo 03). “¿Qué es el Código Morse y cómo funciona?” Consultado en: [https://www.antena3.com/noticias/sociedad/codigo-Morse-que-como-funciona\\_202405036634d12cc0b95c0001dc4ac8.html](https://www.antena3.com/noticias/sociedad/codigo-Morse-que-como-funciona_202405036634d12cc0b95c0001dc4ac8.html)
- Arduino Docs. (2025, marzo 28). “Datasheets”. [pdf.]. Consultado en: <https://docs.arduino.cc/resources/Datasheets/A000066-Datasheet.pdf>
- AREATECNOLOGIA. (2025). “TRIAC”. Consultado en: <https://www.areatecnologia.com/electronica/triac.html>
- Arrow. (2023, abril 19). “Uno: Una visión general de la computadora de placa única más popular de Arduino”. Consultado en: <https://www.arrow.com/es-mx/research-and-events/articles/arduino-uno-product-overview>
- Buenos Aires Radio Club. (2022, octubre 08). “Aprendiendo Morse para el examen de Novicio”. Consultado en: <https://lu4bb.com/blog/aprendiendo-Morse/>
- Electrónica Valtierra. (2025). “Switch interruptor pulsador cuadrado push button normalmente abierto 1V 125VAC”. JPEG. Consultado en: <https://electronicavaltierra.com.mx/producto/sw-742/>
- Elektron. (2025, enero 07). “Socket para focos: ¿Cómo aumentan la eficiencia de tu iluminación?”. Consultado en: <https://www.elektron.com.mx/blog/post/socket-para-focos-%C2%BFcomo-aumentan-la-eficiencia-de-tu->

iluminacion#:~:text=Un%20socket%20para%20focos%20es,que%20la%20luz%20funcione%20correctamente.

- Hardware Libre. (2024, octubre 18). “Buzzer: todo sobre este dispositivo para emitir sonido”. Consultado en: <https://www.hwlibre.com/buzzer/>
- LA NACIÓN. (2024, septiembre 23). “¿Cómo funciona el código Morse?” Consultado en: <https://www.lanacion.com.ar/lifestyle/en-las-redes/como-funciona-el-codigo-Morse-nid19092024/>
- LUMMI. (2025). “Foco tipo vela base E12 (C7.5)”. JPEG. Consultado en: <https://www.lummi.com.mx/products/foco-vela-6>
- Mercado Libre. (2025). “Rollo Cable Estañado Unipolar Calibre #22 Color Rojo 100m”. JPEG. Consultado en: [https://articulo.mercadolibre.com.mx/MLM-2249293578-rollo-cable-estanado-unipolar-calibre-22-color-rojo-100m-\\_JM](https://articulo.mercadolibre.com.mx/MLM-2249293578-rollo-cable-estanado-unipolar-calibre-22-color-rojo-100m-_JM)
- MV ELECTRÓNICA. (2018). “1k Ohm Resistencia De 1/2 Watt”. JPEG. Consultado en: <https://mvelectronica.com/producto/1k-ohms-resistencia-de-1-2-watt>
- National Geographic España. (2023, abril 26). “Código Morse: ¿qué es, ¿cómo funciona y qué tiene que ver con el Titanic?” Consultado en: [https://www.nationalgeographic.com.es/ciencia/codigo-Morse-que-es-como-funciona-que-tiene-que-ver-titanic\\_19830](https://www.nationalgeographic.com.es/ciencia/codigo-Morse-que-es-como-funciona-que-tiene-que-ver-titanic_19830)
- Nugar. (2025). “Resistencias de Cerámica”. Consultado en: <https://nugar-resistor.com/resistencias-de->

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

ceramica/#:~:text=Las%20resistencias%20de%20cer%C3%A1mica%20son,una%20disipaci%C3%B3n%20elevada%20de%20potencia.

- OSAKA Electronics. (2024, abril 18). “Todo sobre Buzzers: Cómo Conectarlos y Para Qué Sirven”. Consultado en:  
<https://osakaelectronicsltda.com/blog/biblioteca/que-es-un-buzzer#:~:text=%C2%BFQu%C3%A9%20es%20un%20Buzzer?,de%20notificaci%C3%B3n%20pueden%20ser%20ineficaces.>
- Preciado S.E. (2025). “Un intérprete de Morse”. [pdf.]. Consultado en:  
[https://www.uam.mx/difusion/casadeltiempo/38\\_39\\_iv\\_dic\\_ene\\_2011/casa\\_del\\_tiempo\\_eIV\\_num38\\_39\\_36\\_39.pdf](https://www.uam.mx/difusion/casadeltiempo/38_39_iv_dic_ene_2011/casa_del_tiempo_eIV_num38_39_36_39.pdf)
- RAYZEEK. (2023, diciembre 26). “Que es la bombilla E12”. Consultado en:  
<https://www.rayzeek.com/es/glossary/que-es-la-bombilla-e12#:~:text=Una%20bombilla%20E12%2C%20tambi%C3%A9n%20conocida,de%20acento%20en%20diversos%20entornos.>
- Sodimac. (2025). “Socket de cerámica tipo E12 Blanco”. JPEG. Consultado:  
<https://www.sodimac.com.mx/sodimac-mx/product/389951/socket-de-ceramica-tipo-e12-blanco/389951/>

## Comunicación en código Morse: Un sistema visual y auditivo en Programación.

- UNIT Electronics. (2016). “¿MOC3021 Optoacoplador para que se utiliza?”. Consultado en: <https://uelectronics.com/producto/moc3021-optoacoplador-dip-6/#:~:text=El%20MOC3021%20se%20utiliza%20para,cargas%20conectadas%20a%20un%20circuito.>
- UNIT Electronics. (2016-2025). “TRIAC BTA08-600B”. Consultado en: <https://uelectronics.com/producto/triac-bta08-600b/>
- UPV/EHU. (2025). “Interruptor pulsador”. Consultado en: <https://www.ehu.eus/es/web/tutorial-myrio/3.-push-button#:~:text=INTERRUPTOR%20PULSADOR&text=Los%20interruptores%20pulsadores%20sirven%20como,corto%20circuito%20cuando%20se%20presiona>  
[n.](https://www.ehu.eus/es/web/tutorial-myrio/3.-push-button#:~:text=INTERRUPTOR%20PULSADOR&text=Los%20interruptores%20pulsadores%20sirven%20como,corto%20circuito%20cuando%20se%20presiona)
- ZW Cables. (2024). “Cable Estañado”. Consultado en: <https://zwcables.com/es/tinned-cable/#:~:text=El%20cable%20esta%C3%B1ado%20es%20un,resistentes%20a%20la%20corrosi%C3%B3n%20atmosf%C3%A9rica.>