

## ANTEPROYECTO DE “PROYECTO MORSE”

Integrante 1: Santiago González Camiscia

Integrante 1: santiagogonzalezcamiscia@impatrq.com

Integrante 2: Mateo Keller

Integrante 2: mateokeller@impatrq.com

Integrante 3: Matias Leonel González Paurailly

Integrante 3: matiasgonzalezpaurailly@impatrq.com

### 1. INTRODUCCIÓN

Este proyecto de decodificador de código morse surge como una manera de relacionar conceptos aprendidos en otro taller. Durante las lecciones sobre los distintos sistemas de aviación, se nos dio a conocer que algunos sistemas de navegación trabajan con el código morse, pero no si los pilotos tenían una manera fácil de entender lo que se les estaba comunicando.

Por esto, desarrollamos un programa que pudiera traducir de código morse a texto, permitiendo así una manera más accesible y rápida de entender estos mensajes, además fortaleciendo así nuestros conocimientos en lenguajes de programación y los distintos protocolos de comunicación como UART o I2C.

### 2. MARCO DE APLICACIÓN

A pesar de que el código morse no es ampliamente utilizado para la comunicación de hoy en día, este sigue siendo utilizado en algunos sistemas de aviación como el Marker o ADF.

Nosotros proponemos implementar este sistema de decodificación para poder otorgarle al piloto un indicador visual y más sencillo de lo que se intenta comunicar.

### 3. DESCRIPCIÓN TÉCNICA

La creación de este proyecto es un proceso sencillo. Para ello, utilizamos dos Raspberry Pi Picos y un display LCD con I2C para mostrar el texto traducido de manera clara.

Los dos microcontroladores se comunican entre sí a través de los puertos UART, conectando sus receptores y transmisores, y compartiendo el mismo GND, lo que asegura una conexión estable y eficiente para el intercambio de datos.

En todo momento, se monitorea el receptor UART del microcontrolador receptor para detectar si llega un nuevo mensaje que necesite ser decodificado. Si es el caso, el sistema iniciará automáticamente el proceso de decodificación para convertir el código morse en texto.

#### 3.1 SOBRE EL HARDWARE

##### 3.1.1 Pi Pico Transmisora

Una simple Pi Pico cuya única función es transmitir mediante sus puertos de UART cualquier tipo de texto en código morse.

##### 3.1.2 Pi Pico Receptora

La Pi Pico más importante, que se encarga de decodificar y mostrar el resultado en el display LCD.

##### 3.1.3 Display LCD 16x2 c/ I2C

El componente encargado de mostrar el texto traducido de forma fácil.

#### 3.2 SOBRE EL SOFTWARE

El código morse se envía a través del puerto transmisor UART del microcontrolador transmisor utilizando una función llamada “send”, en la cual se ingresa un string (texto de código morse) y se envía cada carácter de forma individual utilizando la función “uart\_putc”.

El microcontrolador está constantemente revisando mediante la función “uart\_is\_readable” si hay algún tipo de mensaje esperando. Si es así, lo guarda en un char mediante la función “uart\_getc” y dependiendo del tipo realiza una función distinta. Si es un punto, añade un 1 al array de decodificación de código morse, si es una barra un 0. Si es un espacio, indica el final de una letra e inicia el proceso de decodificación el cual una vez obtenida la letra la añade al string “texto” y actualiza el display con este (lo limpia y lo vuelve a escribir). Si

es una "/", indica una separación entre palabras, por lo que añade un espacio al string "texto". Si es una 'R', tanto el display como los dos arrays de texto y morse se reinician (un Reset total).

## 4. DIVISIÓN DE TAREAS

### 4.1 INTEGRANTE 1

- Investigación acerca del protocolo de comunicación UART.
- Adaptación del sistema desarrollado por el integrante 3 con el protocolo UART.

### 4.2 INTEGRANTE 2

- Búsqueda, obtención, investigación y armado de la parte física del proyecto.

### 4.3 INTEGRANTE 3

- Investigación y desarrollo de un sistema de decodificación de código morse.

## 5. LISTA DE MATERIALES

Básicos:

- 2 Raspberry Pi Pico (Utilizar una sola también es posible utilizando por ejemplo uart0 para transmitir y uart0 para recibir)
- Display LCD 16x2 mediante I2C
- Cables para comunicación
- Protoboard

Opcionales (en caso de no poder, se reemplazará por un cable cumpliendo la misma función):

- Led (IR)
- Fototransistor (IR)

## 5. REFERENCIAS

[1] Ejemplo de UART en Wokwi. Disponible en:  
<https://wokwi.com/projects/397890693498146817>