****

**Universidad Tecnológica de Panamá**

**Facultad de Ingeniería Eléctrica**

**Laboratorio de Computadores Digitales**

**Laboratorio #3**

**Fernando Guiraud**

**8-945-692**

**Profesor Elias Mendoza**

**Grupo: 4EE141**

**Semestre II 2022**

# Introducción

En esta experiencia de laboratorio se utilizará el microcontrolador ESP32. El módulo ESP32 es una solución de Wi-Fi/Bluetooth todo en uno, integrada y certificada que proporciona no solo la radio inalámbrica, sino también un procesador integrado con interfaces para conectarse con varios periféricos.



El objetivo de esta experiencia de laboratorio es generar un algoritmo que sea capaz de controlar dos leds de manera independiente por medio de un botón, generando una secuencia de tres estados, el primer estado consiste en los dos leds apagados mientras el botón se encuentre sin presionar, al presionar el botón se enciende un led y el otro se mantiene apagado, al volver a presionarlo se apaga el mismo y se enciende el siguiente y por último se vuelven a apagar ambos leds repitiendo el ciclo. Esto nos permitirá aprender a conocer el funcionamiento de la declaración de entradas y ademas la configuración por hardware de un pull up resistor.

# Objetivos

* Generar un algoritmo que sea capaz de controlar dos leds con tres estados apagado-apagado, encendido-apagado y apagado-encendido, por medio de interrupciones.

# Procedimiento

En este proyecto es necesario seleccionar el modelo de la tarjeta ESP32 y el framework que en este caso fue Espressif.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Al crear el proyecto se genera una sección llamada platformio.ini donde debemos agregar la siguiente línea para especificar la velocidad de transmisión de datos al monitor serial.

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Posteriormente dentro de la sección scr se encuentra un archivo llamado main.c, en este archivo escribimos el cuerpo del código.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

En las primeras líneas del código se declararon las librerías que contienen las funciones que fueron utilizadas posteriormente.

Texto

Descripción generada automáticamente

Se declararon las variables asociadas a los pines de las entradas y salidas a utilizar. Además se agrega una variable entera con valor inicial en cero para conteos y se configura el botón.

Texto

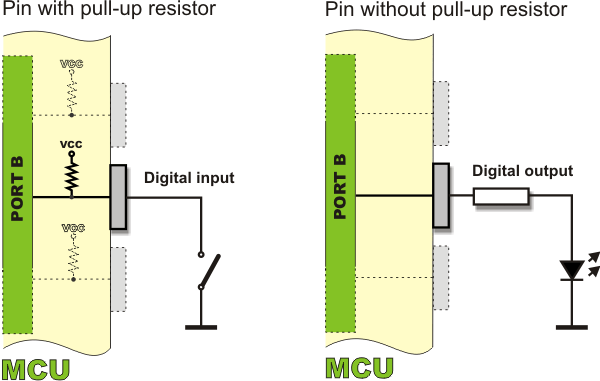
Descripción generada automáticamente

Después de esto se declararon los puertos de entrada y salida por medio de la librería GPIO. Es importante tomar en cuenta que se ha declarado una salida configurada con una resistencia de pull up, en la segunda declaración (io\_config.pull\_up\_en = 1), asignándole un 1 como valor boleando de true. También se establece la interrupción con el botón después de la configuración de las entradas y salidas.

Texto

Descripción generada automáticamente

En la siguiente imagen podemos ver una representación de cómo sería un pull up resistor dentro del ESP32.



Por último, se crea el contenido de la interrupción, que consiste en dos condiciones, cuando el valor de now es mayor a next, se invierte el estado de la variable booleana led\_state, la cual define el estado de encendido y apagado respectivo de cada uno de los leds.

Esta condición de inversión de estados de los leds se repetirá dos veces, encendiendo un led y apagando el otro, después lo inverso y al ocurrir por tercera vez, entra en una condición distinta donde se apagan ambos y luego se reinicia el ciclo.

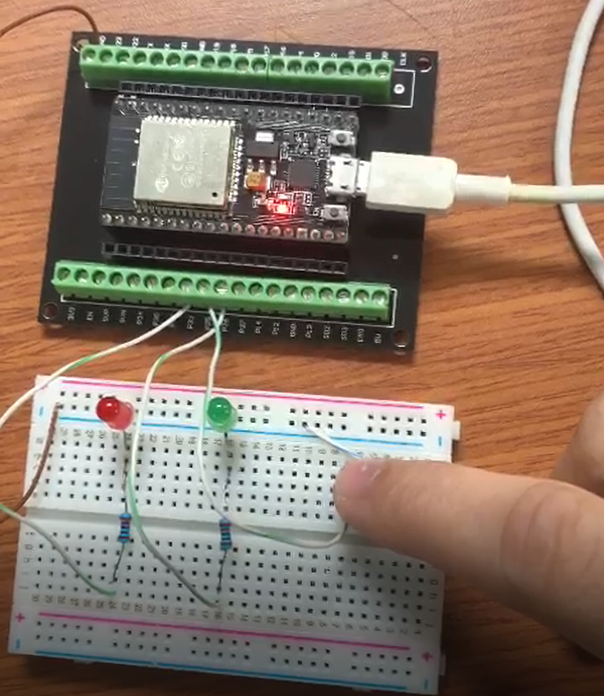
Texto

Descripción generada automáticamente

El codigo completo se ajunta en la sección de anexos.

# Resultados

En las siguientes imágenes se mostrará la secuencia de encendido y apagado de los leds. Adicionalmente se adjuntará un video del funcionamiento en la entrega del informe.



Un circuito electrónico sobre una superficie de madera

Descripción generada automáticamente con confianza baja

Un circuito electrónico

Descripción generada automáticamente con confianza baja

# Conclusiones

En esta experiencia de laboratorio logramos aprender sobre el funcionamiento de las interrupciones para lograr controlar acciones por medio de un botón, de forma que la interferencia producida por el rebote y también el tiempo de presionado del botón, no interfiriera en el cambio de estado de la acción a ejecutar.