

**Universidad Autónoma de Baja California**  
**Facultad de Ciencias Químicas e Ingeniería**  
**Ingeniería en computación**



**MICROCONTROLADORES**

Reporte de practica #3

**Manejo de la sección de E/S del microcontrolador ESP32**

**GRUPO:** 561

**AUTOR:** Rico López Tamara Illian (1270673)

**MAESTRO:** Ricardo Castro Gonzalez

1-Marzo-2021

## Objetivo:

Mediante esta práctica el alumno analizará la implementación de retardos por software, así como también se familiarizará con la configuración y uso de puertos.

## Material:

- Computadora personal.
- Tarjeta de desarrollo del ESP32

## Teoría:

### Técnicas de anti-rebote de botones

Cuando se presiona un botón, un interruptor de micrófono o se mueve un interruptor de palanca, dos partes de metal se unen. Para el usuario, puede parecer que el contacto se realiza de forma instantánea; sin embargo, dentro del interruptor hay partes móviles y este contacto sólo se realiza por una fracción de segundo. Después, se vuelve a hacer contacto por un poco más de tiempo, algunas veces más hasta que al final, el interruptor está completamente cerrado.

Cuando el interruptor está cerrado, los dos contactos en realidad se separan y se vuelven a conectar, generalmente de 10 a 100 veces durante un período de aproximadamente 1 ms. Por lo general, el hardware funciona más rápido que el rebote, lo que hace que el hardware piense que está presionando el interruptor varias veces y dicho hardware suele ser un circuito integrado.

Cada interruptor tiene sus propias características respecto al rebote; ya que incluso dos botones iguales presentarán rebotes diferentes.

Una de las técnicas antirebotes es la configuración del hardware, donde todos los botones están conectados de la misma manera. En ese caso, primero se debe analizar cómo se comportan los interruptores de forma original. Para realizar este análisis, requiere de una señal de reloj, un osciloscopio y un circuito integrado que permita ver cómo presionar el botón impacta en la forma de onda y si podemos saber a través de ella si el botón fue presionado o no. Una vez hecho el análisis, es conveniente hacer uso de un capacitor que permite filtrar la señal; de esta forma, cuando el circuito podría bajar el nivel de voltaje, el capacitor lo compensa, por lo que el botón siempre se considera presionado una vez al ser presionado una sola vez.

Otra técnica para manejar los rebotes es mediante el uso de software, ya que al trabajar con microcontroladores se pueden tratar los rebotes de forma más eficaz y menos costosa si se elimina el uso de hardware externo. El control de los rebotes de los botones es crucial; sin embargo, una mala práctica común en el desarrollo de software es agregar un retraso de 50 ms para que un rebote se detenga y, hasta ese momento, no es posible continuar con el programa.

Una forma más efectiva de controlar los rebotes es el uso de interrupciones, teniendo en cuenta que la interrupción puede dispararse tanto en el borde ascendente como en el

descendente y, en ciertos microcontroladores, las interrupciones se pueden acumular en espera [1].

### Charlieplexing (aplicado para control de LEDs)

Cuando es necesario encender y apagar una cuadrícula de LEDs, se puede usar un pin para cada led; sin embargo, esta no es la mejor opción ya que se hace uso de múltiples pines y, en algún punto, se vuelve inmanejable. Una configuración de matriz puede multiplicar los LEDs controlados mediante el uso de una cuadrícula x/y, donde el número de salida bajo control es igual a  $IO(x)*IO(y)$ .

Una cuadrícula de 3x3 es capaz de controlar 9 LEDs usando solo 6 pines; mientras que una cuadrícula de 8x8 controla 64 LEDs. Este método funciona porque los pines en el eje y activan una fila a la vez, mientras que los pines en el eje x están configurados para recibir corriente a tierra u oponerse a este camino [2].

Los microcontroladores son capaces de configurar los pines en positivo y negativo y configurarlos para que se introduzcan. Este tercer estado, también conocido como alta impedancia, no permite que la corriente fluya de ninguna manera, lo que crea algunas posibilidades de control funcional.

Una técnica excelente para el manejo de LEDs es el "Charlieplexing", llamada así por Charlie Allen, la cual aprovecha al máximo la alta impedancia, con LEDs acomodados de tal forma que puedan pasar corriente en cualquier dirección. El charlieplexing es diferente a la red de LEDs en que el primero presenta una relación exponencial, es decir, el número de LEDs bajo control se puede expresar con la ecuación  $n^2-n$ , donde  $n$  es el número de pin utilizado para controlar los leds.

En particular, Charlieplexing se puede usar para aumentar el número de entradas. Al igual que con las salidas basadas en LED, el costo es una complicación. Además, dado que los LED/diodos no forman parte de la función del dispositivo, se debe tener esto en cuenta al elegir si es mejor usar IO adicional o configurar un circuito Charlieplexing.

## **Desarrollo:**

### Instalación del entorno de desarrollo

Para poder desarrollar código y programas para microcontroladores, se hará uso del editor de texto VS Code, el cual cuenta con una extensión útil llamada PlataformIO. Esta extensión es una alternativa al IDE arduino, ya que permite traba

## **Conclusiones y comentarios:**

El desarrollo de la práctica me permitió aprender a trabajar con la programación en C de forma distinta, al tener que ajustar el código a los requerimientos del microcontrolador. Asimismo, la práctica presentó una oportunidad para poder hacer un código, probar el programa con un simulador a falta del dispositivo físico, así como el analizar la salida de los leds mediante el uso de un analizador de señales. Por otro lado, la práctica fue un buen ejercicio para practicar la depuración del código de los microcontroladores y el uso del hardware como los leds y los puertos; así como el análisis del código morse.

En conclusión, la práctica representó una buena oportunidad para practicar los conocimientos adquiridos en clase, así como para superar algunas dificultades al tener que trabajar con el lenguaje C de forma diferente y no poder imprimir datos, así como el aprender la estructura del código morse.

## **Referencias:**

[1] J. Christoffersen, "Switch Bounce and How to Deal with It," Allaboutcircuits.com, Sep. 03, 2015.

<https://www.allaboutcircuits.com/technical-articles/switch-bounce-how-to-deal-with-it/>  
(accessed Mar. 02, 2022).

[2] J. S. Cook, "Charlieplexing Tutorial: Charlieplexing an LED Matrix," Arrow.com, Sep. 04, 2020.

<https://www.arrow.com/en/research-and-events/articles/charlieplexing-an-led-matrix>  
(accessed Mar. 02, 2022).