

**José Duván Velásquez Guerrero**  
**Ingeniero Electrónico.**

**Estrategias de prolongación de la Batería.**

1. Reducir el brillo de la pantalla LCD.
2. Tras detectar un periodo de inactividad, por medio de un control (Transistores o Relé) desvincular la alimentación de la pantalla LCD.
3. Tras la detección de un mayor tiempo de inactividad someter al ESP32 a un modo deep sleep, para reducir al máximo su consumo.
4. A Través de interrupciones en el Teclado o botón AC creado, activar las interrupciones que despierten al ESP32 del ahorro máximo de energía.
5. Refrescar el contenido de la pantalla LCD cada vez que sea necesario. De esta forma se evita la comunicación constante a través de I2C o SPI

**Observaciones adicionales.**

1. Para un buen manejo de la alimentación y por protección del microcontrolador, el Buzzer debe ser conectado como carga a un Transistor que opere en corte-saturación, para así ser un driver de corriente que brinde el consumo al Buzzer desde la fuente de alimentación y no desde el pin del microcontrolador como suele ser usado en el simulador, ya que los pines soportan corriente pequeñas y este tipo de cargas puede causar inestabilidad de energía en el microcontrolador o averiarlo puesto que son salidas lógicas y no de corriente.
2. Un interruptor entre el circuito y la fuente de alimentación también es una forma de contribuir a la prolongación de la batería ya que desconecta la fuente por completo cuando no está en uso el dispositivo.

**Desarrollo de Algoritmo:**

1. La detección del teclado por iniciativa se realiza a través de un algoritmo de barrido de filas a través de corrimientos de bits desde una multitarea para realizarlo de forma constante, e interrupciones en las columnas para así detectar las posiciones de la Matriz e identificar cual tecla fue oprimida. (La postulación de este método se realiza como alternativa a la librería Keypad.h de arduino, únicamente con el fin de presentar habilidades de desarrollo con otras herramientas y creación de lógica apoyada en hardware).

2. Implementación de sistema Antirebote por software con temporizador.
3. Adición de Botón de Borrado (AC) para reiniciar las operaciones cuando no se quiere continuar operando sobre el resultado acumulado.
4. Se crea un algoritmo que cuenta con la posibilidad de realizar operaciones matemáticas de uno a enésimos términos. Cumpliendo con jerarquía de operadores a través de un diseño propio a partir de cadenas String, nuevamente con el fin de evidenciar el manejo y creación de lógica esta vez con posicionamiento en arreglos.
5. Soporta operaciones con Decimales.
6. Implementación de variables de temporización para detectar estados de inactividad para así apagar la pantalla, entrar en modo deep sleep y/o visualizar el conteo del tiempo restante para entrar en inactividad.
7. Implementación de interrupciones para salir del modo de inactividad a través del teclado y/o tecla AC añadida.
8. Implementación de Relé para desalimentar la pantalla LCD tras inactividad.
9. Implementación Buzzer que se activa con cada tecla oprimida.
10. Implementación Bienvenida LCD.

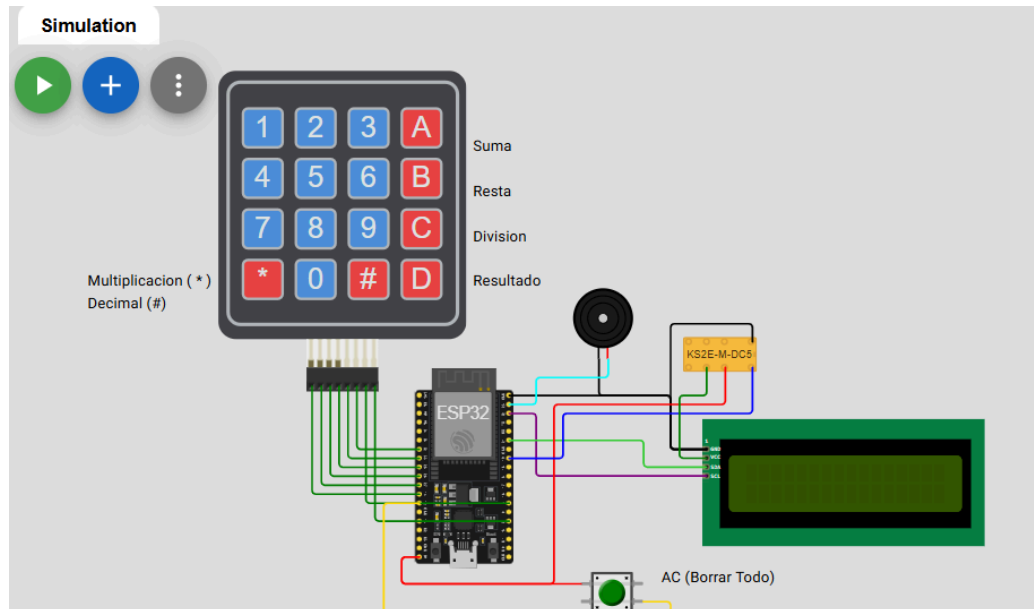
## Pruebas Unitarias

Estas pruebas realizadas con Unity contemplan las siguientes operaciones encargadas de realizar toda la lógica del cálculo de operaciones matemáticas de enésimos términos con jerarquía de operadores. Adjunto puede encontrarse el proyecto de Visual Studio Code.

```
RUN_TEST(test_Numero_operadores);  
RUN_TEST(test_Encontrar_Index_Operador_Ascendente);  
RUN_TEST(test_Encontrar_Index_Operador_Descendente);  
RUN_TEST(test_Operacion_Matematica);  
RUN_TEST(test_Operacion_Matematica_2);  
RUN_TEST(test_Realizar_Operacion);  
RUN_TEST(test_calcular_Operacion_Completa);
```

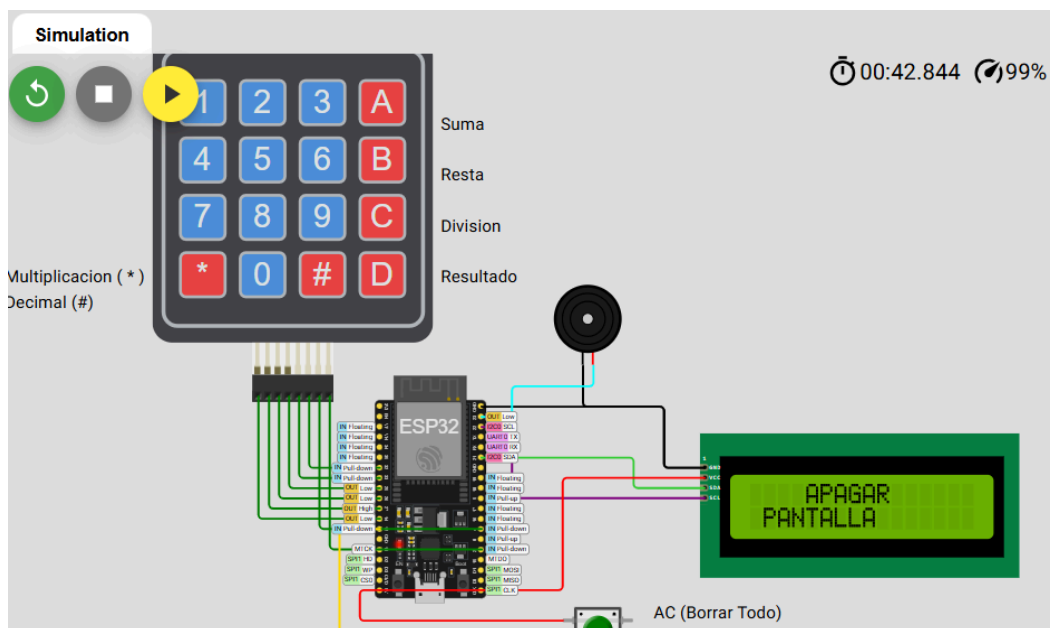
## Simulaciones en Wokwi:

El sistema de activación y desactivación de la pantalla LCD no fue posible a través de conmutación debido a que Woki no permite el apagado de la pantalla LCD.

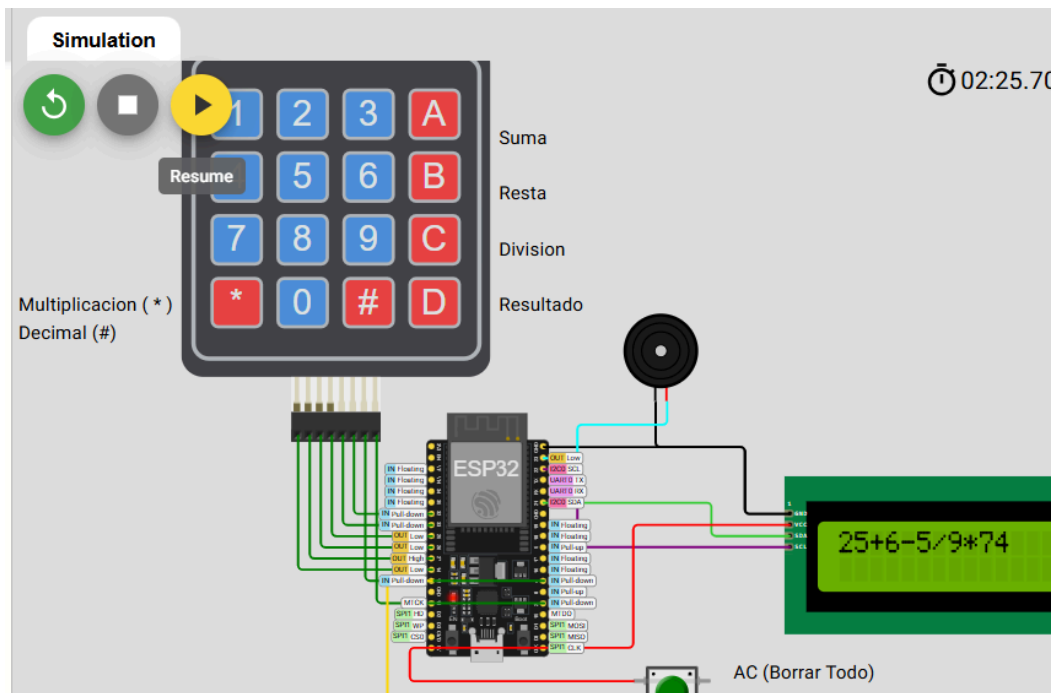


**Figura 1:** Imagen de implementación propuesta con un Relé.

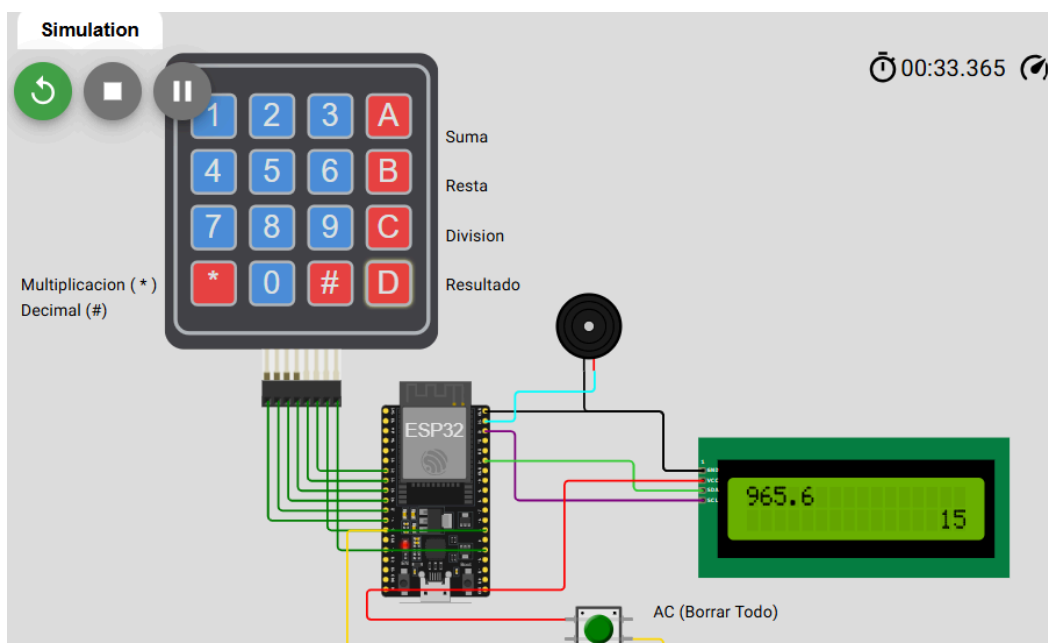
Como alternativa se propone la visualización de la ejecución de apagado en la pantalla LCD. La cual se “apaga” al cabo de 10 segundos de inactividad y se “enciende” al activar el teclado.



**Figura 2:** Emulación apagado de pantalla.



**Figura 3:** Funcionamiento de Calculadora con enésimos terminos y diferentes operaciones.



**Figura 4:** Número superior: Resultado u operaciones. Número inferior: Conteo de temporización de inactividad.

Adicionalmente el modo de ahorro de energía que fue implementado, es ejecutado por Wokwi. Sin embargo, la activación por medio de interrupciones no puede llevarse a cabo dado que el simulador Wokwi ejecuta la salida del modo deep sleep del ESP32 posterior a su entrada a ese estado.