

# Proyecto 2 - Sensor de presión atmosférica SPI, TFT y SD

Karla Melissa 19374, Ingeniería Biomédica, UVG  
Noviembre, 2021

## Resumen

Utilizando el microprocesador ESP32 se realizó un sistema de medición de presión atmosférica junto con el sensor BMP180, además se implementó el uso de una pantalla LCD de 16x2 para desplegar los valores del sensor. Además se utilizó el microprocesador Tiva C y se desempeñó una rutina donde el ESP32 por medio de la comunicación UART se mandaran los datos tanto a la Tiva C como a la computadora. A parte de mantener esta comunicación con la Tiva se usó la comunicación SPI para que la Tiva C se comunicará con una pantalla TFT con memoria micro SD, para desplegar los datos obtenidos del sensor de presión y enviados por el ESP32 donde se mostrará el dato y con la memoria SD por medio de una rutina con los botones de la Tiva se desplegará un nuevo dato o se guardará en la memoria micro SD y con una rutina usando un modulo Buzzer identificar con una melodía la acción que se realizó.

## Palabras claves

*Microcontrolador, Tiva C, ESP32, SPI, TFT y SD.*

## Sensor de presión atmosférica

El módulo BPM 180 es un módulo que incluye un sensor de presión el cual se utilizó para las mediciones en este proyecto, para su uso se utilizó el microprocesador ESP 32 y se programó en el programa *Arduino* para crear una rutina donde se comunicara con el sensor.. Para esta comunicación se requirió utilizar una librería específica para el módulo, la cual se descargó de Internet para poder usar distintas funciones del sensor. Además de la comunicación con el sensor, se implementó el uso de una pantalla LCD16x2 para mostrar los datos del sensor, como se puede ver en la figura 1.

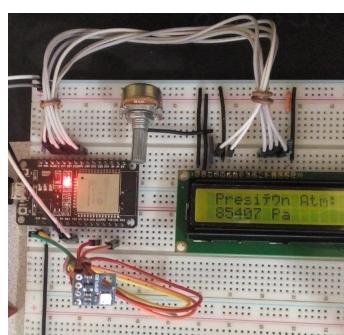


Figura 1. Circuito con el sensor de presión

## Comunicación UART

La comunicación UART (Universal Asynchronous Receiver-Transmitter) se usó para la comunicación con la Tiva y la computadora, para ello se utilizó la función de serial. Utilizando el botón 1 de la Tiva se comunicaba con el ESP 32 por medio de los puestos serie de cada placa para mandar el dato del sensor de presión y la Tiva fuera capaz de recibirlos. La comunicación UART permitió mandar los datos del sensor de una manera secuencial en serie, mandando los datos en bits individuales por un puerto y usando la misma lógica se inició la comunicación con la computadora.

## Pantalla TFT

Para esta parte se utilizó una pantalla TFT y se programó usando el programa *Energia* y utilizando distintas librerías y funciones para poder configurar y ejecutar los comandos de inicio de la pantalla TFT. Usando distintos programas de diseño de imágenes y convertidores de imágenes se diseñó una interfaz que por medio de la comunicación SPI se logró transmitir en la pantalla. Para poder usar imágenes con peso mayores de los que soporta la ram del microprocesador, se agregó un archivo de gráficos.c para almacenar la matriz de la imagen y que esta no ocupa mucha memoria del microprocesador. Además de desplegar el diseño de la interfaz, también despliega el valor del sensor que se obtuvo en la comunicación UART con el ESP 32, y por medio del botón 1, se va actualizando el valor del sensor.



Figura 2. Diseño de la pantalla TFT

## Almacenamiento en memoria micro SD

Para esto se utilizó una memoria micro SD compatible con la pantalla TFT, es una memoria micro SD HC de nivel 1 con una capacidad de 16GB. Para almacenar los datos se utilizó una librería SD y se implementó una rutina con el botón 2 de la Tiva C, para que se guardaran los datos en un archivo .csv que permitiera almacenar los datos del sensor. Para esta función se utilizó la librería SD para abrir el archivo, guardar el dato y cerrarlo, de esta

manera se llamaba la función y se aseguraba que al guardarla y cerrarlo el archivo no quedara como corrupto y fuera posible leerlo en la computadora.

The terminal window shows the following output:

```
Writing to data.csv...
85522
done.
85515
6
8
5
5
1
5

Writing to data.csv...
85515
done.
```

The Excel spreadsheet has columns A and B, with the following data:

	A	B
1	85508	
2	85512	
3	85522	
4	85522	
5	85522	
6	85515	
7		
8		

Figura 3. Almacenamiento de los datos del sensor en la memoria SD

## Modulo Buzzer

Por último, esta parte se implementó utilizando un módulo buzzer conectado a la Tiva C para que al momento de implementar una de las rutinas con los botones de la Tiva C, ya sea actualizar el dato del sensor con el botón 1 o guardar el dato con el botón 2, el buzzer sonará una melodía que significa e indica que la acción se estaba ejecutando, de esa manera actuará como un indicador en cada una de las rutinas.

## Anexos

Link a un video de Youtube

<https://youtu.be/fUwHkHtuYDY>

Link de Repositorio de GitHub

[https://github.com/Karla19374/PROYECTO\\_2\\_DIGITAL\\_19374](https://github.com/Karla19374/PROYECTO_2_DIGITAL_19374)

Imagen completa del proyecto

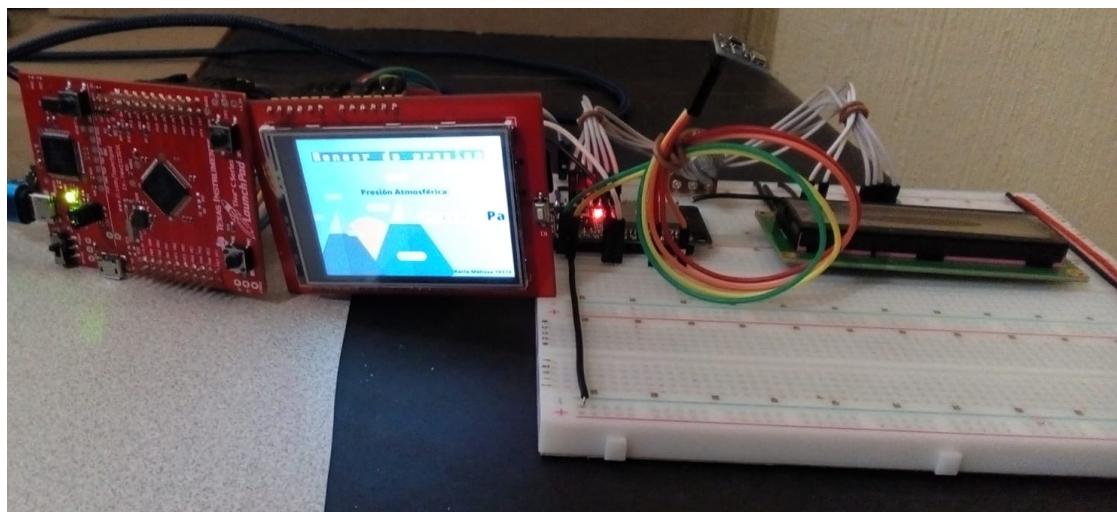


Diagrama de flujo del proyecto.

