

# LABORATORIO DE PRINCIPIOS DE MECATRÓNICA

25 de febrero de 2022

---

## Práctica #2

Arduino

### Grupo:

L001

### Estudiante:

- Bermúdez Guillen  
Alejandro
- Caballero Enciso  
Carla
- Montes de Oca Villa  
Luciano

### Profesor:

Benito Granados-Rojas

## Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Experimentos y Simulaciones</b>	<b>2</b>
2.1. Experimento 2.1: Convertidor Analógico Digital . . . . .	2
2.2. Experimento 2.2: Liquid-Crystal Display (LCD) . . . . .	2
2.3. Experimento 2.3: Servomotor . . .	2
<b>3. Conclusiones</b>	<b>3</b>
<b>4. Enlaces externos</b>	<b>3</b>

## 1. Introducción

El entorno Arduino se puede ampliar mediante el uso de librerías. Para poder utilizarlas, es necesario seleccionarlas de la siguiente manera: Sketch ¿Import Library. Existe una gran variedad de librerías; no obstante, para la práctica nosotros solamente utilizamos LiquidCrystal, Servo y SoftwareSerial. La primera, LiquidCrystal, es útil para controlar pantallas de cristal líquido (LCD); la segunda, Servo, sirve para dominar servomotores y, la tercera, SoftwareSerial, funciona para la comunicación en serie con cualquier pin digital [1].

## 2. Experimentos y Simulaciones

### 2.1. Experimento 2.1: Convertidor Analógico Digital

En el primer ejercicio construimos un circuito, en donde conectamos un potenciómetro de 10 kOhms a una terminal analógica A0. Después, añadimos un LED al puerto digital D13 con una resistencia de 220 Ohms. A continuación, mostramos en el monitor el voltaje entre 0 y 5 V y su valor depende de la posición del potenciómetro. Más tarde, modificamos el código para que el LED solo prendiera si el voltaje era mayor a 3V y utilizamos la siguiente expresión:  $volt = (val * 5.0) / 1023.0$  para identificar el voltaje de entrada. Finalmente, usamos la función AnalogWrite() para que el LED encienda con una intensidad proporcional al voltaje de entrada con el potenciómetro. Aquí nos dimos cuenta que los valores de AnalogRead() van de 0 a 1023 y los valores de AnalogWrite() van de 0 a 255 [2], entonces solo dividimos el valor de entrada entre cuatro. Los códigos de estos ejercicios que llevan por nombre “ADC3V” y “ADCIntVol” se encuentran en GitHub (véase *Enlaces externos*).

### 2.2. Experimento 2.2: Liquid-Crystal Display (LCD)

En este ejercicio, utilizamos el Arduino y un display de cristal líquido (LCD). Luego de realizar las conexiones necesarias en el circuito, realizamos la programación necesaria para desplegar el nombre de un integrante del equipo. En el programa que realizamos, tuvimos que incluir la librería “LiquidCrystal”. Fue necesario contar el número de filas y columnas en el LCD. En cada ciclo o estado, limpiamos el lcd, movíamos la posición del cursor e imprimimos las letras o palabras. Para que el mensaje se visualice correctamente, ocupamos algunos delay. El código de este ejercicio que lleva por nombre “LCD” se encuentra en GitHub (véase *Enlaces externos*).

### 2.3. Experimento 2.3: Servomotor

Para esta parte de la práctica usamos el código de la primera sección y parte de la anterior. Se podría decir que juntamos los 2 códigos y le metimos más código. La parte del potenciómetro la usamos para controlar en qué posición está el servomotor. Después usamos la parte del LCD para poder imprimir los datos que se nos pedían, los cuales son el voltaje suministrado y el ángulo en el que estaba el servomotor. Lo adicional que tuvimos que meterle fue la librería “myservo” para que hubiera

movimiento con el servomotor. El código de este ejercicio que lleva por nombre “Servomotor” se encuentra en GitHub (véase *Enlaces externos*).

### 3. Conclusiones

Esta práctica se nos hizo muy interesante, a diferencia de la práctica anterior, en esta utilizamos más componentes que cada vez se volvieron complejos en mayor medida pero también más fascinantes de usar y creemos que se pueden ocupar en un contexto profesional. Otra cosa que nos pareció relevante fue que tuvimos que ir, de cierta forma, “construyendo” la práctica para poder llegar de lo más básico a lo más complejo que fue la última parte, pero al ir haciéndola de poquito en poquito no estuvo tan complicado como si se hubiera hecho de golpe todo.

### 4. Enlaces externos

<https://github.com/Lucio27MV/Lab-Principios-Meca>

### Referencias

[1] Arduino. AnalogWrite. 2022/02/23, de Arduino. Sitio web: <https://www.arduino.cc/reference/en/language/functions/analog-io/analogwrite/>

[2] Arduino. Libraries. 2022/02/23, de Arduino. Sitio web: <https://www.arduino.cc/reference/en/libraries/>