

Guía Teórica #2 Empezando con Arduino

1. Introducción

En esta última experiencia presentaremos a un nuevo elemento, será el actor más importante en nuestros proyectos, el **microcontrolador**, básicamente es un solo chip que contiene en su interior todos los elementos fundamentales de un computador, una CPU, una memoria RAM y puertos de entrada y salida. Su principal característica es esa, un computador que cabe en la palma de la mano o inclusive en un dedo, que puede funcionar con un mínimo de elementos extra.

Sin embargo debido a su reducido tamaño y a el gran número de situaciones en las que se utiliza un microcontrolador (μC), hay una gran cantidad de modelos distintos, cada uno con distinto número de entradas y salidas, diferentes capacidades de memoria y alguna otra característica especial que lo distinga.

Arduino es una plataforma de cómputo con un entorno de programación simple que posibilita programar un μC fácilmente. Esto nos permite realizar cálculos, enviar o recibir información por los puertos, almacenar datos y transmitir órdenes a algún otro dispositivo.



Figura 1: Placa de desarrollo Arduino UNO.

Los puertos pueden ser de entrada o salida, lo que permite enviar datos o instrucciones hacia el exterior, por ejemplo para encender un LED, o desde el exterior hacia el μC , por ejemplo un botón para activar alguna rutina. Cada uno de estos puertos es un pin y posee un número asociado para identificarlo, este se utiliza para dar las instrucciones y realizar las funciones requeridas en el programa en arduino (sketch).

Estos pines pueden estar asociados a varias funciones específicas, lo típico es que sean entradas o salidas digitales. Pero también pueden ser entradas análogas (entregando un número proporcional al voltaje al que está sometido el pin) o bien de comunicaciones, como el puerto serial que se utiliza para programar Arduino.

2. Servomotor

Un servomotor es un motor con un sistema para controlar su posición, este tipo de dispositivos electrónicos es muy común, especialmente en el aeromodelismo, la robótica e incluso algunos juguetes. Estos servomotores (servos) reciben una señal a través de uno de sus cables, esta representa la posición a la que debe estar, el servo rota su eje hasta alcanzarla e intenta mantenerla en caso de que alguna fuerza externa se oponga. Comúnmente los servos se mueven en rangos de 180° , 90° o 360° y su movimiento está limitado a este rango, pudiendo dañarse si se los fuerza a los límites o más allá.



Figura 2: Servomotor común.

El tipo señal de control para el servomotor es conocida como señal PWM (*Pulse-Width Modulation*), consiste en una serie de pulsos de una duración de 0.7 a 2.0 ms repetidos a una frecuencia de 50 Hz ($T = 20$ ms), la figura 3 indica la respuesta del servomotor a distintos anchos de pulso, los que resultan proporcionales a la posición del servomotor. Este tipo de salidas PWM no solo sirve para transmitir información, sino que puede utilizarse para controlar la potencia dada a una carga, por ejemplo controlar la velocidad de un motor DC o intensidad de un LED.

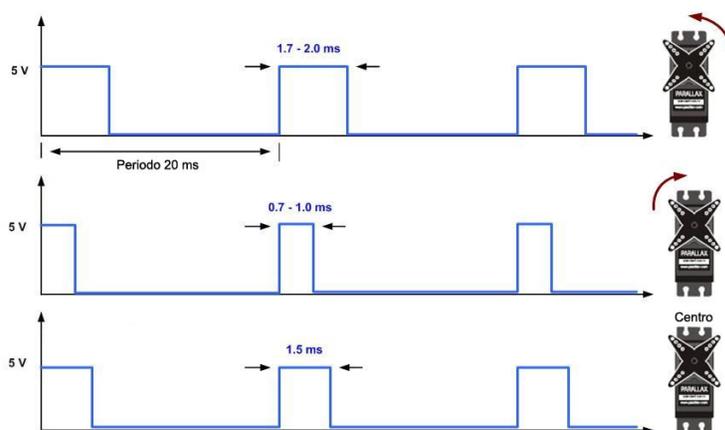


Figura 3: Señal PWM para control de servomotores.

Arduino implementa una librería `Servo` para controlar servomotores, se encarga de la generación de la señal PWM y del manejo de ángulos.

3. Sensor de ultrasonido

Un sensor de ultrasonido es un dispositivo, que bajo el mismo principio de la eco-localización de delfines o murciélagos, emite un sonido inaudible para el ser humano y registra el tiempo en que este sonido tarda en viajar hasta algún objeto, rebotar y ser captado por el sensor. Sabiendo a la velocidad que el sonido en el aire permite conocer la distancia a la que se encuentra el objeto, la cual es traducida como voltaje a uno de sus pines, donde puede ser leído por otro dispositivo electrónico.



Figura 4: Sensor de ultrasonido.

4. Programación en Arduino IDE

El lenguaje de programación que emplea Arduino IDE es muy similar C/C++, para familiarizarse con su sintaxis y estructuras de control se recomienda la lectura de una guía rápida enfocada en Arduino.

- *Guía de usuario de Arduino* por Rafael Enríquez H. Capítulo 5: Comenzando con Arduino. (Disponible en U-Cursos).
- [Página web oficial de Arduino.](#)