

## Taller de Introducción a Arduino

Impartido por Abel Arellano y Omar Valencia

Duración: 5 sesiones de 4 horas  
del -- al ---

### Presentación:

La presencia de computadoras en sus diferentes formas es cada vez más frecuente en objetos de uso cotidiano, el desarrollo paralelo de la industria electrónica ha reducido en gran medida el costo de microcontroladores haciendo accesibles dispositivos con gran potencial de aplicaciones en diferentes ámbitos como el arte, el diseño y arquitectura; entre otras, permitiendo que cada vez más personas adquieran fácilmente conocimientos en electrónica y con esto amplíen los alcances de su trabajo o se aproximen a la interdisciplina como punto de partida para nuevos proyectos.

Arduino se ha constituido como una plataforma de gran popularidad entre la oferta de microcontroladores disponibles actualmente pues reúne características como: Hardware libre y abierto o a un bajo costo, un software con una interfaz intuitiva, un lenguaje de programación muy accesible y una gran comunidad de usuarios y desarrolladores que aportan contenido y ayuda a los usuarios nuevos y experimentados.

### Objetivo general:

Este curso se plantea como una introducción para el uso de la plataforma Arduino, con ejercicios prácticos el asistente conocerá las primeras etapas de construcción y configuración de circuitos electrónicos que permiten a la tarjeta reaccionar e interactuar con el entorno.

El asistente conocerá las diferentes tarjetas disponibles y su proceso de instalación; así como las características de funcionamiento más usadas en el diseño y construcción de instalaciones y objetos interactivos.

El asistente, reconocerá las etapas de programación, compilación, comunicación de la computadora con la tarjeta y los ciclos de ejecución de la tarjeta una vez que el programa se encuentra “grabado” en el microcontrolador

Dotar a los asistentes de conocimientos prácticos en electrónica y a sus proyectos con características que les permitan aprovechar las posibilidades de una plataforma tan versátil como Arduino.

## Sesión 1

El Ambiente de Desarrollo Integrado (Integrated Developed Environment) de Arduino.

Arduino como plataforma de Hardware libre esta diseñado para trabajar conjuntamente con un entorno de programación (software abierto y libre) que permite explorar y explotar fácilmente las posibilidades de utilización, mediante un lenguaje accesible estructurar programas muy sencillos o programas que incluyan diferentes funciones y librerías externas.

Por otro lado la realización de proyectos con Arduino a menudo requiere un circuito electrónico auxiliar que se puede diseñar y probar en un protoboard, una vez obtenido el comportamiento deseado se construye una versión definitiva que se puede ensamblar fácilmente al microcontrolador (shield) para evitar desconexiones y fallas.

Seleccionar el software a instalar dependiendo de la computadora del usuario.

Una vez instalados los controladores y el IDE, explicar botones y menús

Características físicas y electrónicas de Arduino.

Mostrar las diferentes etapas de desarrollo de un proyecto: Configuración y ensamblado de circuitos en en protoboard, optimización de variable eléctricas y soldado de un circuito auxiliar en una tarjeta externa (shield).

Material: Tarjetas Arduino, cables USB, protoboards, LED, resistencias a  $\frac{1}{4}$  de watt de 220 ohm.

Ejercicio: Observar las conexiones que se realizan a través de los puertos seriales de una computadora y cómo son traducidos por la tarjeta microcontroladora para comunicarse con la computadora personal.

Observar mensajes de error frecuentes en el uso de Arduino y explicar posibles soluciones.

Ensamblar un circuito en protoboard para encender un led alimentado con la fuente de voltaje integrada en Arduino.

Comentario de práctica: Los microcontroladores funcionan como microcomputadoras que se pueden comunicar entre otros microcontroladores y con computadoras personales, a través de estas conexiones se programan rutinas que Arduino automáticamente ejecutara una vez que se conecte a un suministro eléctrico

Habilidades adquiridas: Resolver problemas que pueden surgir en la comunicación de arduino y los puertos seriales de la computadora.

Conocimientos de los menús de opciones y herramientas del software para programar Arduino.

Reconocer características y funciones básicas de circuitos eléctricos.

Diagramas: Conexiones de un circuito para encender un led en un protoboard, símbolos de los componentes electrónicos usados y símbolos de circuitos eléctricos.

Material complementario en:

<http://www.andreadicastro.com/academia/electronica/proto/>

## Sesión 2

Puertos de entrada y salida de la tarjeta microcontroladora.

Las aplicaciones de Arduino abarcan un amplio rango de situaciones en las que es posible utilizar al coordinadamente microcontroladores y circuitos electrónicos; ya sea para registrar impulsos externos como el encendido/apagado de un switch u obtener una lectura aproximada de señales análogas.

Mediante la programación de Arduino es posible generar comportamientos reactivos e interactivos según los datos de entrada con lo que se reconocerá la relación entre circuitos electrónicos (hardware) y rutinas que pueden ser programadas en ellos (software), lo que permite dotar a estos dispositivos de cierta autonomía

Introducción a la programación de microcontroladores en lenguaje Arduino.

Partes principales de un programa de Arduino: declaración de variables, setup() y loop()

Recepción de impulsos por medio de DigitalRead y Analog Read en la tarjeta microcontroladora. Serial monitor.

Material: Tarjetas Arduino, cables USB, LDR (Light Dependant Resistor) de 10Mohms, Laser TTL, pushbutton, interruptores (normalmente abierto), cable 18, 20 awg, Leds, resistencias a  $\frac{1}{4}$  de watt de 10Kohm y 220ohms, multímetro, conectores macho-macho, caimanes, protoboards.

Ejercicio: Escribir programa configurando los pines digitales como entrada para registrar el encendido apagado de un switch y de un pushbutton, escribir un programa que registre la interrupción de un switch laser y encienda un led, escribir un programa que registre la cantidad de luz recibida por un LDR y encienda un led con una intensidad proporcional a la iluminación recibida, observar los datos en un Serial monitor.

Comentario de práctica: Los microcontroladores están diseñados como pequeñas computadoras con puertos para entradas y salidas de datos, es posible por medio de la programación activar comportamientos combinados de entrada y salida así como diseñar funciones en código para responder a determinados eventos y reutilizarlas como componentes de otros programas

Habilidades adquiridas: Reconocer diferencias de datos en formatos digitales y análogos

Diagramas: Conexiones un interruptor con resistencia pulldown, conexión de un LDR para registrar un laser y conexión de un led a pin digital, conexión de un LDR a las entradas de análogas de arduino y conexión de un Led a pin PWM.

## Sesión 3

### Sensores

Existen en el mercado una extensa cantidad de sensores que transforman variables presentes en el mundo físico a señales eléctricas o electrónicas cuyos datos es posible interpretar con la tarjeta microcontroladora Arduino. Para las prácticas de esta sesión utilizaremos el emisor-sensor infrarrojo, el PIR (passive infrared sensor) y el sensor de ultrasonido. Estos tres sensores facilitan la conexión con Arduino y la interpretación de datos permite utilizarlos como sensores de proximidad

Obtención de datos de componentes electrónicos de las hojas de datos (datasheets).  
Definición de funciones y programación orientada a objetos.

Material: Tarjetas Arduino, cables USB, sensores Infrarrojos SHARP (de corto o largo alcance) , Sensores de ultrasonido (LV-MaxSonar EZ1), Sensores PIR, cable 18, 20 awg, capacitores de 100uF, resistencias a  $\frac{1}{4}$  de watt de 220Ohm, multímetro, conectores macho-macho, caimanes, protoboards.

Ejercicio: Escribir programa que inicialice la comunicación serial entre el microcontrolador y la computadora, Conectar y comparar entre la formas de recibir datos de los sensores en formato analógico y PWM, filtrar los datos por promedio, observar los datos en un Serial monitor.

Comentario de práctica: Muchos de los sensores utilizados para registrar presencia y distancia de personas son en sí mismos circuitos electrónicos diseñados para comunicarse con microcontroladores sin embargo es necesario procesar las señales que envían para obtener una mejor respuesta del microcontrolador

Habilidades adquiridas: Organización del código que integra un programa de Arduino en funciones específicas.  
Obtener referencias de componentes electrónicos y circuitos integrados por medio de las hojas de datos (datasheets).  
Identificación de cualidades de diferentes sensores.  
Adecuación de escalas para los datos provenientes de sensores

Diagramas: Conexiones de sensor sharp 2Y0A21 con la tarjeta Arduino (ver esquema en pág. 267 del libro Making Things Talk), Conexión del sensor LV-MaxSonar EZ1 con la tarjeta microcontroladora, ver foto adjunta (lleva una resistencia de 220Ohm a la entrada de 5v y un capacitor de 100uF entre la línea de tierra para filtrar componente de DC).

Material complementario en:

<http://www.andreadicastro.com/academia/ComputoFisico/CF.html>

<http://www.andreadicastro.com/academia/electronica/ARDUINO2html/Sensores.html>

[http://bildr.org/2011/06/pir\\_arduino/](http://bildr.org/2011/06/pir_arduino/)

## Sesión 4

### Relays

Las funciones de control que permiten los microcontroladores abarca rangos de voltaje y corriente relativamente pequeños que son suficientes para alimentar un par de leds o actuadores pequeños como piezoeléctricos, sin embargo es posible controlar cargas mayores por medio de circuitos eléctricos auxiliares que son activados mediante los impulsos que proporciona la tarjeta microcontroladora, dichos circuitos pueden soportar cargas como la iluminación de una estancia o dispositivos fotográficos.

Actuadores, control por medio de relays

Control de circuitos de mayor voltaje (127 v aprox.) por medio de un Arduino

Tipos de Relays; de bobina y reed

Material: Tarjetas Arduino, cables USB, transistores npn 2222, diodo 4004, resistencias a 1/4 de watt de 330 ohms, cable 18 o 20 awg, relays Sun Hold MD-5, relays Sun Hold RAS-0510, multímetro, conectores macho-macho, caimanes, protoboards.

Ejercicio: Por medio de un programa de Arduino que utilice los pines digitales en modo de digitalWrite, conmutar conexiones de voltajes relativamente altos con diferentes tipos de relays, para activar diferentes dispositivos que funcionen con voltajes y corrientes mayores a los que utiliza la tarjeta, por ejemplo: electrodomésticos o circuitos para activar motores.

Comentario de práctica: Es posible comprender el funcionamiento eléctrico de un relay observando que las salidas digitales de la tarjeta, tienen una amplitud máxima de 5 volts y 40 mA, acoplándolas a circuitos de relays es posible controlar valores mayores de voltaje y corriente, de esta forma las posibilidades de control que brinda la tarjeta Arduino se amplían.

Habilidades adquiridas: Identificar corrientes y voltajes de circuitos externos que se pueden controlar con Arduino.

Distinguir que tipo de relay o acoplador es más adecuado para diferentes potencias.

Filtración de picos de corriente originados por bobinas de relays mecánicos

Diagramas: Conexiones de Arduino a relay por medio de un transistor npn y diodo, conexiones de circuitos externos

Material complementario en:

<http://www.andreadicastro.com/academia/electronica/ARDUINO2html/foto.html>

## Sesión 5

### Motores

Los microcontroladores permiten el control de una gran cantidad de motores, otro tipo de actuadores de gran importancia que facilitan el movimiento de objetos y otras posibilidades existentes en aparatos que basan su funcionamiento en el movimiento de uno o varios motores.

A menudo se requieren circuitos complementarios a la tarjeta Arduino para controlar eficazmente algunos tipos de motores.

Motores de DC

Motores stepper

Servomotores

Material: Tarjetas Arduino, cables USB, Protoboards, conectores macho-macho, Adafruit motorshield, Circuito integrado L293D, servomotores, motores de DC, motores stepper bipolares, resistencias a  $\frac{1}{4}$  de watt de 220Ohm, multímetro, conectores macho-macho, caimanes, protoboards.

Ejercicios: Activar y controlar un motor de dc por medio de un puente H y Arduino, activar un servomotor usando Arduino y la programación incluida en los ejemplos, Activar un motor stepper por medio del motorshield de Adafruit y verificar los diferentes tipos de paso y velocidades, comunicar Arduino con Processing para controlar la rotación de un servomotor desde la computadora.

Comentario de práctica: La mejor elección de un motor en un proyecto estará determinada por diferentes parámetros, el peso del objeto a mover o la velocidad y la precisión del movimiento requerido.

Habilidades adquiridas: Identificar potencia y variables de diferentes tipos de motores.  
Conocimiento de ejemplos y librerías incluidas en Arduino para manejo de diferente servomotores y stepper.  
Ensamblado de circuitos complementarios para alimentar y controlar que requieran mayor potencia.

Diagramas: Conexiones de un puente H a Arduino, conexión de un servomotor a Arduino, conexión de un motor stepper a Adafruit motorshield

Material complementario en:

<http://www.andreadicastro.com/academia/electronica/motors.html>