

# PRINCIPIOS DE AUTO FABRICACIÓN ELECTRÓNICA

**ARDUINO  
+  
GROVE**

Guía académica



El contenido de software del presente taller ha sido creado y puesto a prueba en su totalidad por Ámbar Labs. Ámbar Labs no se responsabiliza por cualquier tipo de daño personal y/o material producto de no seguir el procedimiento indicado en esta guía para el trabajo de laboratorio.

El presente documento contiene información recopilada de:

[www.ambarlabs.com](http://www.ambarlabs.com)  
[www.openhardware.com](http://www.openhardware.com)  
[www.arduino.cc](http://www.arduino.cc)  
[www.seeedstudio.com](http://www.seeedstudio.com)

Imagen de fondo de portada:

[www.desktopwallpapers.org.ua](http://www.desktopwallpapers.org.ua)

Ámbar Labs © Lima, 2015  
Todos los derechos reservados.

# ÍNDICE

<b>1. Introducción</b>	<b>1</b>
<b>2. Lista de materiales</b>	<b>2</b>
• Arduino UNO	2
• Shield Grove	2
• Grove - Botón (Button)	3
• Grove - Sensor táctil (Touch sensor)	3
• Grove - LED (LED Socket)	3
• Grove - Zumbador (Buzzer)	3
• Grove - Potenciómetro (Rotary Angle Sensor)	3
• Grove - Micrófono (Sound Sensor)	3
• Grove - LDR (Light Sensor)	4
• Grove - Termistor (Temperature Sensor)	4
• Grove - Relé (Relay)	4
• Grove - Servo (Servomotor)	4
• Sensor ultrasonido	4
• Grove - LCD (LCD-RGB Backlight)	4
• Grove – Cables	5
<b>3. Antes de iniciar</b>	<b>5</b>
<b>4. Ejercicios de aplicación</b>	<b>6</b>
• Ejercicio 1	6
• Ejercicio 2	7
• Ejercicio 3	7
• Ejercicio 4	8
• Ejercicio 5	9
• Ejercicio 6	10
• Ejercicio 7	10
• Ejercicio 8	11
• Ejercicio 9	12
• Ejercicio 10	12
• Ejercicio 11	13
• Ejercicio 12	13
• Ejercicio 13	14
• Ejercicio 14	14
• Ejercicio 15	15
• Ejercicio 16	16
<b>5. Conversatorio</b>	<b>17</b>

# 1. INTRODUCCIÓN

*Arduino* es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios. Está orientada a personas con perfiles creativos, con o sin experiencia previa en temas de hardware y software, interesados en desarrollar proyectos electrónicos de forma fácil e intuitiva.

*Grove* es un set de herramientas electrónicas modulares listas para ser usadas. Su principal función es simplificar la tarea de trabajar con componentes electrónicos sin soldaduras ni errores de conexión posibles, facilitando al mismo tiempo el proceso de aprendizaje de manera significativa. Los módulos *Grove* varían yendo desde un simple botón hasta sensores complejos, cada uno acompañado de su propia documentación y ejemplos de aplicación.

*Arduino* y *Grove* trabajando en conjunto simplifican el proceso de desarrollo de prototipos y proyectos electrónicos, desde su concepción hasta su implementación final. Ambas son herramientas de hardware libre lo cual estimula y acelera el aprendizaje abriéndonos las puertas a la comunidad mundial de desarrolladores de tecnologías abiertas.

Este taller está orientado a público en general con espíritu creativo e interesado en desarrollar sus propios proyectos y/o productos electrónicos. No es necesario tener conocimientos previos en electrónica y programación. Este documento incluye toda la información necesaria para empezar a trabajar con la placa *Arduino UNO* en su computadora, una guía de los sensores *Grove* incluidos en el kit, y la guía de los ejercicios de aplicación que trabajaremos en el taller.

---

## Ámbar Labs

Somos un equipo profesional compuesto por ingenieros, programadores y diseñadores dedicados a la investigación constante sobre nuevas maneras de fusionar arte con tecnología. Desde el 2007 trabajamos en proyectos de innovación tecnológica, formación académica en temas de desarrollo de software y hardware libre y propuestas de arte contemporáneo basadas en tecnología.

Formamos parte de una sociedad estratégica con el Centro de Innovación y Desarrollo Emprendedor de la Pontificia Universidad Católica del Perú.

---

## Open Hardware

Somos una empresa del grupo *Ideas Múltiples* dedicada a la investigación, desarrollo y divulgación de nuevos productos basados en plataformas abiertas de hardware y software. A través de ella queremos poner al alcance de la comunidad los productos necesarios para poder investigar y desarrollar nuevos prototipos y promover el hardware libre como medio de compartir el conocimiento.

Somos distribuidores oficiales de *Arduino* y *Seeed Studio* en el Perú además de importar diversos componentes y complementos para robótica en general.

## 2. LISTA DE MATERIALES

### Arduino UNO

Se trata de una tarjeta electrónica programable basada en un microcontrolador ATmega328. Tiene 14 pines de entradas y salidas digitales (de las cuales 6 pueden utilizarse como ondas PWM para producir señales electrónicas analógicas), 6 entradas analógicas, un resonador cerámico de 16MHz, conexión USB, conexión de energía externa, un cabezal ICSP y un botón de Reset. Contiene todo lo que se necesita para soportar a un microcontrolador. Para empezar a trabajar, simplemente tiene que conectarlo a una PC a través de un cable USB, o energizarlo con un adaptador de corriente AC/DC, o una batería.

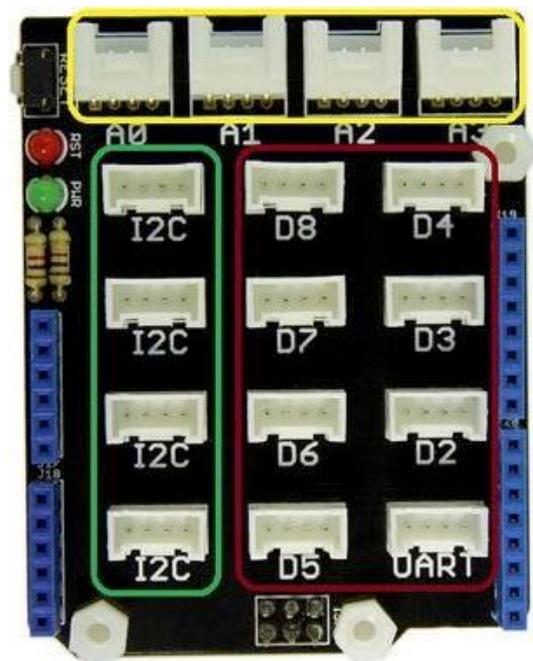


### Shield Grove

La placa base *Grove* se conecta a una placa *Arduino*, y es la base de todo el sistema *Grove*. Todos los puertos de entradas y salidas están expuestos y adaptados a 16 conectores Grove los cuales incluyen las entradas y salidas digitales, analógicas y puertos especializados (ICSP, I2C, UART).

En la parte superior, demarcado por la línea amarilla, se ubican los conectores de entradas analógicas. Las entradas analógicas se utilizan para leer sensores analógicos, tales como un potenciómetro o un sensor de temperatura; estos puertos pueden también ser utilizados como puertos digitales.

A la derecha, demarcado por la línea roja, se ubican los 8 conectores de entradas y salidas digitales. Estos pueden utilizarse para leer y controlar datos de los módulos Grove, tal como el sensor de luz y los LEDs. Algunos de estos pines pueden utilizarse como salidas de ondas PWM (Pulse Width Modulation: Modulación de Ancho de Pulso), las cuales le sirven a Arduino para controlar el movimiento de un motor de pasos, o controlar la intensidad de un LED. Finalmente hacia la izquierda, demarcado por la línea verde, se ubican los conectores de comunicación serial I2C. Estos se usan para conectar sensores y dispositivos más sofisticados como, por ejemplo, acelerómetros, módulos Bluetooth, pantallas LCD, etc.



## 2. LISTA DE MATERIALES

### Grove – Botón (Button)

Botón simple listo para ser usado como entrada digital.



### Grove – Sensor táctil (Touch sensor)

Sustituto para el botón simple que funciona en base a la medida de la capacitancia. Se activa al tocar el área circular del pad.



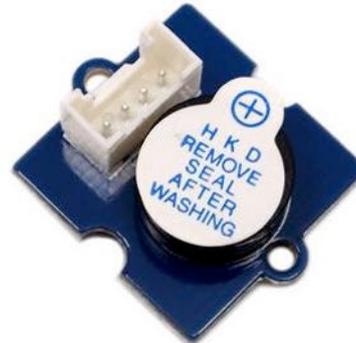
### Grove – LED (LED Socket)

El LED es la pieza más familiar en el aprendizaje de sistemas electrónicos. Se utilizan para indicar eventos, crear efectos de luz, etc.



### Grove – Zumbador (Buzzer)

Componente utilizado para generar tonos de audio de frecuencia fija y/o variables.



### Grove – Potenciómetro (Rotary Angle Sensor)

Este es un potenciómetro lineal rotativo de 10 kOhm (unidad resistiva) que posee un rango activo de 300 grados.



### Grove – Micrófono (Sound Sensor)

Combinación de un micrófono electret y un amplificador LM386. Cuenta con un potenciómetro que permite controlar el nivel de entrada.



## 2. LISTA DE MATERIALES

### Grove – LDR (Light Sensor)

El LDR es una resistencia cuyo valor depende de la intensidad de luz que recibe. A mayor cantidad de luz, menor resistencia.



### Grove – Termistor (Temperature Sensor)

El termistor es un componente cuya resistencia varía dependiendo de la temperatura al que es expuesto (de -40 a 120 grados Celsius).



### Grove – Relé (Relay)

Componente de tipo interruptor controlado digitalmente. Se utiliza para controlar circuitos de mayores niveles de corriente y/o voltaje.



### Grove – Servo (Servomotor)

Motor de precisión cuyo eje tiene un radio de acción de 180°.



### Sensor ultrasonido

Componente utilizado para medir distancias basado en la emisión/recepción de ondas de ultrasonido.



### Grove – LCD (LCD-RGB Backlight)

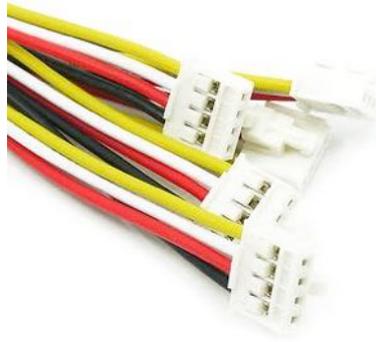
Pantalla de tipo LCD y fondo de color ajustable. Nos permite visualizar datos directamente sin necesidad de una computadora.



## 2. LISTA DE MATERIALES

### Grove – Cables

Cables de conexión entre el shield *Grove* y los diversos componentes del kit.



## 3. ANTES DE INICIAR

Una vez reconocidos todos los materiales y teniendo todo dispuesto sobre su mesa de trabajo, debemos hacer algunas comprobaciones previas para empezar a trabajar con el kit:

- Instalar el software *Arduino IDE* en su PC para poder trabajar con su placa *Arduino UNO*. Para realizar la instalación, por favor revise los documentos anexos que vienen en su memoria USB Flash titulados *Arduino – Mac.pdf* y *Arduino – Windows.pdf* y siga los pasos según el sistema operativo que le corresponda.
- Conectar el shield Grove a la placa Arduino Uno.
- En el *Arduino IDE* ir a **Herramientas -> Puerto Serial (Tools -> Serial Port)** y verificar que se reconoce la placa conectada; debería aparecer el indicador **COM#** donde **#** corresponde a un número. Si no se reconoce la placa *Arduino Uno* desconéctela y vuélvala a conectar. Si esto no soluciona el problema reinicie el *Arduino IDE* y vuelva a intentarlo. Si aún no se reconoce la placa comuníquelo al instructor.

Luego de haber comprobado que el sistema funciona, ya estamos listos para comenzar con los ejercicios de aplicación.

## 4. EJERCICIOS DE APLICACIÓN

### EJERCICIO 1

#### Descripción:

Parpadeo automático de un LED.

En este ejercicio conectaremos uno de los puertos digitales del shield *Grove* a un LED, el cual será controlado por un programa que lo enciende y apaga de forma automática en intervalos de 1 segundo. Los puertos digitales del *Arduino* pueden ser configurados como salidas o como entradas. En este caso, el puerto estará configurado como salida para controlar al LED. Este ejercicio nos es útil para comprobar que la combinación *Arduino* + *Grove* + PC están funcionando sin problemas.

#### Procedimiento:

1. Conectar la placa *LED Socket* al puerto **D3** del shield *Grove*.
2. Conectar un led a los pines correspondientes de la placa *LED Socket*; verificar la polaridad del led.
3. En el *Arduino IDE* abrir el archivo de nombre "**\_01\_salidaDigital\_1.ino**".
4. Cargar el programa en la placa *Arduino UNO* usando alguno de los siguientes métodos:
  - Ir a **Archivo -> Cargar (File -> Load)**.
  - Usando la combinación de teclas **Ctrl + U**.
  - Haciendo clic con el mouse en el botón de carga (botón redondo ubicado a la izquierda con la flecha apuntando hacia la derecha).
5. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.
6. Con el circuito en funcionamiento desconectar la placa *Relay* y conectar la placa *Buzzer* a la misma salida **D3**; verificar el funcionamiento del circuito.
7. Con el circuito en funcionamiento desconectar la placa *LED Socket* y conectar la placa *Relay* a la misma salida **D3**; verificar el funcionamiento del circuito.

## 4. EJERCICIOS DE APLICACIÓN

### EJERCICIO 2

#### Descripción:

Generación de melodías simples.

Esta vez utilizaremos la placa *Buzzer* para escuchar una melodía generada por el *Arduino*, la cual se escucha una sola vez. Para volver a escucharla es necesario reiniciar el *Arduino*. Una melodía es una secuencia de notas musicales y silencios ordenados con cierta pauta, velocidad y ritmo, tal como podremos ver en el código de este ejercicio. En este caso el *Arduino* utiliza una salida digital para generar señales eléctricas vibrando a distintas frecuencias, y la placa *Buzzer* se encarga de traducir esta señal eléctrica en señales audibles que el oído reconoce como notas musicales. Para cada frecuencia corresponde una nota musical.

#### Procedimiento:

1. Conectar la placa *Buzzer* al puerto **D3** del shield *Grove*.
2. En el *Arduino IDE* abrir el archivo de nombre “\_o2\_salidaDigital\_2.ino”.
3. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
4. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.

### EJERCICIO 3

#### Descripción:

Encendido y apagado de un LED de forma manual con un botón.

Esta vez introduciremos el uso de los puertos digitales del *Arduino* como entradas digitales. Una entrada digital permite introducir señales de dispositivos electrónicos -como botones, switches, pulsadores, sensores magnéticos, etc.- para controlar acciones con el *Arduino*. En este ejemplo utilizaremos la placa *Button* y la placa *Touch* para controlar el encendido y apagado de un LED.

#### Procedimiento:

1. Conectar la placa *Button* al puerto **D2** del shield *Grove*.
2. Conectar la placa *LED Socket* al puerto **D3** del shield *Grove*.

## 4. EJERCICIOS DE APLICACIÓN

3. Conectar un led a los pines correspondientes de la placa *LED Socket*; verificar la polaridad del led.
4. En el Arduino IDE abrir el archivo de nombre "**\_03\_entradaDigital.ino**".
5. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
6. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.
7. Con el circuito en funcionamiento desconectar la placa *Button* y conectar la placa *Touch* a la misma salida **D2**; verificar el funcionamiento del circuito.

### EJERCICIO 4

#### Descripción:

Lectura del nivel de señal de sensores analógicos.

En este ejercicio introduciremos el uso del potenciómetro como sensor analógico, y de la herramienta monitor serial para leer el valor del potenciómetro conectado al shield *Grove*. Al girar la perilla del potenciómetro, podremos ver que varía el valor numérico que se muestra en el monitor serial; a menor ángulo, menor valor numérico y viceversa. Los sensores analógicos son sensores que producen señales continuas y variables, como por ejemplo un potenciómetro (sensor de giro manual), un sensor de temperatura, de nivel de sonido, de nivel de luz, entre otros. Los sensores digitales (botones, switches, etc.) solo producen señales de ON/OFF, mientras que los sensores analógicos producen una variedad de niveles de señal; ambos tienen naturalezas distintas, y por ello sus comportamientos y aplicaciones también lo son.

#### Procedimiento:

1. Conectar la placa *Rotary Angle* al puerto **A0** del shield *Grove*.
2. En el *Arduino IDE* abrir el archivo de nombre "**\_04\_entradaAnalog\_1.ino**".
3. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
4. En el *Arduino IDE* abrir el monitor serial usando alguno de los siguientes métodos:
  - Ir a Herramientas -> **Monitor Serial (Tools -> Serial Monitor)**.
  - Usando la combinación de teclas **Ctrl + Shift + M**.
  - Haciendo clic con el mouse en el botón del monitor serial (botón cuadrado ubicado a la derecha con el ícono de lupa).

## 4. EJERCICIOS DE APLICACIÓN

5. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.
6. Con el circuito en funcionamiento desconectar la placa *Rotary Angle* y conectar la placa *Light Sensor* a la misma entrada **A0**; verificar el funcionamiento del circuito.
7. Con el circuito en funcionamiento desconectar la placa *Light Sensor* y conectar la placa *Sound Sensor* a la misma entrada **A0**; verificar el funcionamiento del circuito.

### EJERCICIO 5

#### Descripción:

Lectura del nivel de señal de un sensor analógico complejo.

Muchas veces los sensores analógicos no tienen un comportamiento lineal, algunos de ellos podrían tener un comportamiento exponencial, logarítmico, geométrico, etc. Tal es el caso del sensor de temperatura, el cual tiene un comportamiento complejo que necesita de una fórmula de conversión para poder determinar el valor exacto de la temperatura. Este ejercicio es un ejemplo de cómo adaptar el valor de entrada de un sensor no lineal con técnicas de software (uso de variables, funciones matemáticas, etc.) para obtener valores de salida más familiares para un usuario que no necesita entender las complejidades del mundo digital.

#### Procedimiento:

1. Conectar la placa *Temperature* al puerto **A0** del shield *Grove*.
2. En el *Arduino IDE* abrir el archivo de nombre "**\_05\_entradaAnalog\_2.ino**".
3. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
4. Abrir el monitor serial en el *Arduino IDE* tal como se explicó en el ejercicio 4.
5. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.

## 4. EJERCICIOS DE APLICACIÓN

### EJERCICIO 6

#### Descripción:

Encendido y apagado de un LED de forma progresiva y automática.

En el primer ejercicio introducimos el control del encendido y apagado de un LED, pero ¿cómo lograr que el LED varíe su intensidad de luz de forma progresiva? Para esta aplicación introduciremos el uso de señales PWM (Pulse Width Modulation, o Modulación de Ancho de Pulso). Una señal PWM es una señal digital variable que se comporta como una salida analógica y nos permite controlar dispositivos de forma paulatina, como por ejemplo la potencia de la intensidad de un LED, el giro de un motor, el volumen de un sonido, etc. *Arduino* tiene ciertos puertos digitales asignados con la capacidad para generar este tipo de señales PWM.

#### Procedimiento:

1. Conectar la placa *LED Socket* al puerto **D3** del shield *Grove*.
2. Conectar un led a los pines correspondientes de la placa *LED Socket*; verificar la polaridad del led.
3. En el *Arduino IDE* abrir el archivo de nombre “\_o6\_salidaPWM.ino”.
4. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
5. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.
6. Con el circuito en funcionamiento desconectar la placa *LED Socket* y conectar la placa *Buzzer* a la misma salida **D3**; verificar el funcionamiento del circuito.

### EJERCICIO 7

#### Descripción:

Control de giro de un servomotor de forma automática.

En este ejercicio aplicaremos una señal PWM para controlar el giro de un servomotor de 0 a 180 grados y viceversa. Un servomotor es un tipo de motor eléctrico con control de posición angular, el modelo incluido en el kit *Grove* tiene un rango de movimiento limitado a 180 grados, controlado de forma digital y que es muy popular en aplicaciones robóticas.

## 4. EJERCICIOS DE APLICACIÓN

### Procedimiento:

1. Conectar el *Servo* al puerto **D3** del shield *Grove*.
2. Acoplar al eje del *Servo* uno de los brazos de accesorio incluidos.
3. En el *Arduino IDE* abrir el archivo de nombre "**\_07\_servoMotor.ino**".
4. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
5. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.

### EJERCICIO 8

### Descripción:

Detección de distancias con el sensor de ultrasonido.

Este sensor está basado en un sistema de emisión y recepción de pulsos de ultrasonido. Tomando como referencia el tiempo transcurrido entre la emisión y recepción de un pulso de señal se calcula la distancia física entre el sensor y el objeto. Utiliza el mismo principio que los delfines y los murciélagos para calcular su posición en el espacio. Para este ejercicio, hemos limitado la distancia máxima de medición a 200cm la cual será mostrada en el monitor serial.

### Procedimiento:

1. Conectar el cable especial del sensor ultrasonido al puerto **D7** del shield *Grove*.
2. Conectar el pin *Trigger* del ultrasonido al cable de color blanco.
3. Conectar el pin *Echo* del ultrasonido al cable de color amarillo.
4. Conectar el pin *VCC* del ultrasonido al cable de color rojo.
5. Conectar el pin *GND* del ultrasonido al cable de color negro.
6. En el *Arduino IDE* abrir el archivo de nombre "**\_08\_ultraSonido.ino**".
7. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
8. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.

## 4. EJERCICIOS DE APLICACIÓN

### EJERCICIO 9

#### Descripción:

Uso de la pantalla LCD para visualizar datos.

El kit *Grove* incluye una pantalla LCD que es muy útil para visualizar información de sensores y dispositivos controlados por el *Arduino*. Gracias a esta pantalla, podemos prescindir del uso de la PC y el monitor serial al visualizar datos específicos. Esto se traduce en portabilidad y practicidad al diseñar sistemas electrónicos.

#### Procedimiento:

1. Conectar la placa *Grove-LCD* al cualquier puerto **I2C** del shield *Grove*.
2. En el *Arduino IDE* abrir el archivo de nombre “\_09\_pantallaLCD.ino”.
3. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
4. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.

### EJERCICIO 10

#### Descripción:

Theremin de luz.

A partir de este ejercicio, empezaremos a aplicar los conocimientos aprendidos hasta ahora para crear pequeños proyectos combinando algunos de los sensores y/o actuadores incluidos en el kit *Grove*. Un theremin es un instrumento musical eléctrico que a través de una antena, detecta la posición de la mano del músico, produciendo notas musicales según la variación de esta distancia. De forma similar, este proyecto produce notas musicales al variar el nivel de intensidad de luz que incide en el sensor de luz. Mientras menor sea la cantidad de luz detectada, menor será la frecuencia de la nota musical, y viceversa.

#### Procedimiento:

1. Conectar la placa *Light Sensor* al puerto **A0** del shield *Grove*.
2. Conectar la placa *Buzzer* al puerto **D3** del shield *Grove*.
3. En el *Arduino IDE* abrir el archivo de nombre “\_10\_thereminLuz.ino”.

## 4. EJERCICIOS DE APLICACIÓN

4. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
5. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.

### EJERCICIO 11

#### Descripción:

Control manual de la intensidad luminosa de un LED.

En este ejercicio, utilizaremos el potenciómetro para regular manualmente la potencia de la luz de un LED a través de la variación de la onda PWM que lo alimenta.

#### Procedimiento:

1. Conectar la placa *LED Socket* al puerto **D3** del shield *Grove*.
2. Conectar un led a los pines correspondientes de la placa *LED Socket*; verificar la polaridad del led.
3. Conectar la placa *Rotary Angle* al puerto **A0** del shield *Grove*.
4. En el *Arduino IDE* abrir el archivo de nombre "**\_11\_controlLed\_1.ino**".
5. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
6. Abrir el monitor serial en el *Arduino IDE* tal como se explicó en el ejercicio 4.  
Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.

### EJERCICIO 12

#### Descripción:

Control por distancia de la intensidad luminosa de un LED.

Este ejercicio es una variación del ejercicio anterior, pero en vez de utilizar el potenciómetro para variar la onda PWM utilizaremos el sensor de ultrasonido.

#### Procedimiento:

1. Conectar el cable especial del sensor ultrasonido al puerto **D7** del shield *Grove*.

## 4. EJERCICIOS DE APLICACIÓN

2. Conectar el pin *Trigger* del ultrasonido al cable de color blanco.
3. Conectar el pin *Echo* del ultrasonido al cable de color amarillo.
4. Conectar el pin *VCC* del ultrasonido al cable de color rojo.
5. Conectar el pin *GND* del ultrasonido al cable de color negro.
6. Conectar la placa *LED Socket* al puerto **D3** del shield *Grove*.
7. Conectar la placa *Grove-LCD* al cualquier puerto **I2C** del shield *Grove*.
8. En el *Arduino IDE* abrir el archivo de nombre "**\_12\_controlLed\_2.ino**".
9. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
10. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.

### EJERCICIO 13

#### Descripción:

Control manual de la posición de un servomotor.

Esta vez utilizaremos el potenciómetro para regular el ángulo de posición de un servomotor a través de la variación de la onda PWM que lo alimenta.

#### Procedimiento:

1. Conectar el *Servo* al puerto **D3** del shield *Grove*.
2. Conectar la placa *Rotary Angle* al puerto **A0** del shield *Grove*.
3. Conectar la placa *Grove-LCD* a cualquier puerto **I2C** del shield *Grove*.
4. En el *Arduino IDE* abrir el archivo de nombre "**\_13\_controlServo\_1.ino**".
5. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
6. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.

### EJERCICIO 14

#### Descripción:

Control por distancia de la posición de un servomotor.

Esta vez controlaremos el ángulo de giro del servomotor con el sensor de ultrasonido. El principio es similar al ejercicio anterior, pero considerando que el

## 4. EJERCICIOS DE APLICACIÓN

rango de distancias para controlar el giro del motor es de 0 a 30cm siendo este valor modificable.

### Procedimiento:

1. Conectar el *Servo* al puerto **D3** del shield *Grove*.
2. Conectar la placa *Grove-LCD* a cualquier puerto **I2C** del shield *Grove*.
3. Conectar el cable especial del sensor ultrasonido al puerto **D7** del shield *Grove*.
4. Conectar el pin *Trigger* del ultrasonido al cable de color blanco.
5. Conectar el pin *Echo* del ultrasonido al cable de color amarillo.
6. Conectar el pin **VCC** del ultrasonido al cable de color rojo.
7. Conectar el pin **GND** del ultrasonido al cable de color negro.
8. En el *Arduino IDE* abrir el archivo de nombre "**\_14\_controlServo\_2.ino**".
9. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
10. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.

### EJERCICIO 15

### Descripción:

Control por distancia de un relé.

Para este ejercicio se activará y desactivará un relé dependiendo de la distancia de un objeto a un sensor ultrasonido. Tomando como base esta distancia se podrán generar mensajes predeterminados en una pantalla LCD y así mismo activar un dispositivo externo, como una alarma, en caso sea necesario.

### Procedimiento:

1. Conectar la placa *Relay* al puerto **D2** del shield *Grove*.
2. Conectar la placa *Grove-LCD* a cualquier puerto **I2C** del shield *Grove*.
3. Conectar el cable especial del sensor ultrasonido al puerto **D7** del shield *Grove*.
4. Conectar el pin *Trigger* del ultrasonido al cable de color blanco.
5. Conectar el pin *Echo* del ultrasonido al cable de color amarillo.
6. Conectar el pin **VCC** del ultrasonido al cable de color rojo.
7. Conectar el pin **GND** del ultrasonido al cable de color negro.
8. En el *Arduino IDE* abrir el archivo de nombre "**\_15\_controlPorDistancia.ino**".
9. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
10. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.

## 4. EJERCICIOS DE APLICACIÓN

### EJERCICIO 16

#### Descripción:

Activación de un circuito externo a través de eventos sonoros.

En este ejercicio utilizaremos el sensor de sonido para conmutar la placa *Relay* que servirá de puente para alimentar un circuito externo de mayor potencia. Para activar el circuito de mayor potencia será necesario generar un sonido cuya intensidad sea mayor a un valor umbral determinado por software. Para desactivar la alimentación es necesario generar nuevamente un sonido de las mismas características. El led nos indica que el sistema está listo para recibir un nuevo impulso sonoro. Este circuito de mayor potencia podría estar conectado al sistema de corriente doméstica (220 VAC) y alimentar dispositivos de bajo consumo como lámparas, cargadores de celular, equipos de sonido, etc.

#### Procedimiento:

1. Conectar la placa *Sound Sensor* al puerto **A0** del shield *Grove*.
2. Conectar la placa *Relay* al puerto **D2** del shield *Grove*.
3. Conectar la placa *Grove-LCD* a cualquier puerto **I2C** del shield *Grove*.
4. Conectar la placa *LED Socket* al puerto **D3** del shield *Grove*.
5. Conectar un led a los pines correspondientes de la placa *LED Socket*; verificar la polaridad del led.
6. En el *Arduino IDE* abrir el archivo de nombre "**\_16\_controlPorSonido.ino**".
7. Cargar el programa en la placa *Arduino* tal como se explicó en el ejercicio 1.
8. Verificar el funcionamiento del circuito de acuerdo a lo indicado por el instructor.

## 5. CONVERSATORIO

¿Y ahora? Una vez que ya hemos aprendido el funcionamiento básico de los componentes del kit *Grove*, así como nuestras primeras nociones en lo que a programación se refiere, es hora de darse tiempo para un respiro. Aprovechemos esta pausa para exponer nuestras ideas y propuestas, lo que quisiéramos hacer en adelante, las nuevas combinaciones que se nos ocurren, lo que aún no hemos visto pero quisiéramos ver y un largo etcétera.

Demos rienda suelta a nuestro espíritu creativo y busquemos soluciones en conjunto, de manera que podamos complementar nuestras habilidades y talentos. Lo que el artista necesita puede estar alojado en la mente del ingeniero y viceversa.

No olvidemos que ya hemos dado el primer paso, nuestro ingreso honorario, a una comunidad de desarrolladores que se mantiene activa día a día gracias al perfil innovador y colaborativo de cada uno de sus miembros. A manera de inspiración les sugerimos tomar en cuenta esta breve lista de páginas web con información sobre electrónica, programación y tutoriales de diseño e implementación de productos de tipo *hágalo usted mismo* (DIY):

[www.adafruit.com](http://www.adafruit.com)  
[www.sparkfun.com](http://www.sparkfun.com)  
[www.instructables.com](http://www.instructables.com)  
[www.makezine.com](http://www.makezine.com)

Adicionalmente nuestras puertas siempre estarán abiertas para aquellos que deseen profundizar más sus conocimientos relacionados a temas de desarrollo, hardware & software libre, arte en general y afines.