



# **EL TUTORIAL KIT SUPER ARRANQUE PARA UNO**

## Prefacio

### Nuestra compañía

Establecido en 2011, Elegoo Inc. es un fabricante profesional y exportador que está involucrado en el diseño, desarrollo, producción y comercialización de Arduino, las impresoras 3D, Frambuesa Pi y STM32. Estamos ubicados en Shenzhen, que se conoce como el Silicon Valley Chino. Todos nuestros productos cumplen con las normas internacionales de calidad y son muy apreciadas en una variedad de diferentes mercados en todo el mundo.

Websitio Oficial: <http://www.elegoo.com>

Tienda de Amazon de Spain.: <http://www.amazon.es/shops/A1780XYQ9DFQM6>

### Nuestro Tutorial

Este tutorial está dirigido a principiantes. Usted aprenderá toda la información básica acerca de cómo utilizar la placa controladora de Arduino , sensores y componentes. Si quieres estudiar Arduino con mayor profundidad, se recomienda que lea el Recetario de Arduino escrito por Michael Margolis.

### Nuestro servicio

Si usted tiene cualquier pregunta, comentario o sugerencia acerca de nuestra empresa, producto o tutorial, por favor no dude en enviar un correo electrónico a [EUservice@elegoo.com](mailto:EUservice@elegoo.com). Estamos comprometidos con la mejora continua por lo que su opinión es muy importante.

## Lista de empaque

[www.elegoo.com](http://www.elegoo.com)

Motor De Pasos



Servo (SG90)



Módulo Del Receptor IR



Elegoo UNO R3 Board



Relé



LCD1602 Visualizador



ULN2003 Módulo Del Controlador De Motor De Pasos



Módulo De Fuente De Alimentación De Placa Universal



Prototipo De Tarjeta De Expansión



Contáctenos: [EUservice@elegoo.com](mailto:EUservice@elegoo.com)

 [www.elegoo.com](http://www.elegoo.com)

Visualizador De Segmentos



Visualizador De 7 Segmentos De 4 Dígitos



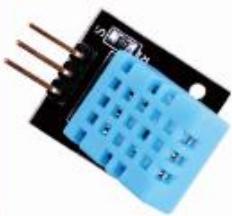
Interruptor De Bola



## 6v DC Motor



DHT11 Módulo



Módulo De Sensor Ultrasónico



Contáctenos: [EUservice@elegoo.com](mailto:EUservice@elegoo.com)

 [www.elegoo.com](http://www.elegoo.com)

Interruptores De Empuje  
5 Piezas



Prototipo De Tarjeta De Expansión



Zumbador Pasivo



Conecotor De La Batería De 9V Con Corriente Continua



Zumbador Activo



Módulo De Palanca



USB Cable



F-M Alambres  
10 Piezas



M-M Alambres  
65 Piezas



Remoto IR



Placa Universal



74HC595



Contáctenos: [EUservice@elegoo.com](mailto:EUservice@elegoo.com)

 [www.elegoo.com](http://www.elegoo.com)



Contáctenos: [EUservice@elegoo.com](mailto:EUservice@elegoo.com)

## Catálogo de cursos

Lección 0: Instalación IDE .....	8
Lección 1: Añadir Bibliotecas.....	16
Lección 2: Parpadeo .....	20
Lección 3: LED.....	27
Lección 4: RGB LED .....	33
Lección 5: Entradas Digitales.....	40
Lección 6: Ocho LEDs con 74HC595 .....	44
Lección 7: El Monitor Serial.....	50
Lección 8: Fotocélula.....	56
Lección 9: Haciendo sonidos .....	60
Lección 10: Zumbador Pasivo.....	64
Lección 11: Interruptor de bola.....	67
Lección 12: Relé.....	70
Lección 13: 74HC595 y Visualizador de Segmentos .....	73
Lección 14: Visualizador de 7 segmentos de 4 dígitos .....	78
Lección 15: Servo.....	82
Lección 16: Visualizador LCD .....	85
Lección 17: Termómetro .....	90
Lección 18: Módulo de Sensor Ultrasónico.....	94
Lección 19: DHT11 Sensor de Temperatura y Humedad .....	98
Lección 20: Módulo de palanca de mando analógica .....	103
Lección 21: Módulo de Receptor IR .....	106
Lección 22: DC Motores .....	110
Lección 23: Motor de Pasos .....	115
Lección 24: Motor de Pasos Controlando con Remoto.....	117

# Lección 0: Instalación IDE

## Introducción

En esta lección, aprenderá cómo configurar su ordenador con el fin de utilizar Arduino y cómo ir sobre las lecciones que siguen.

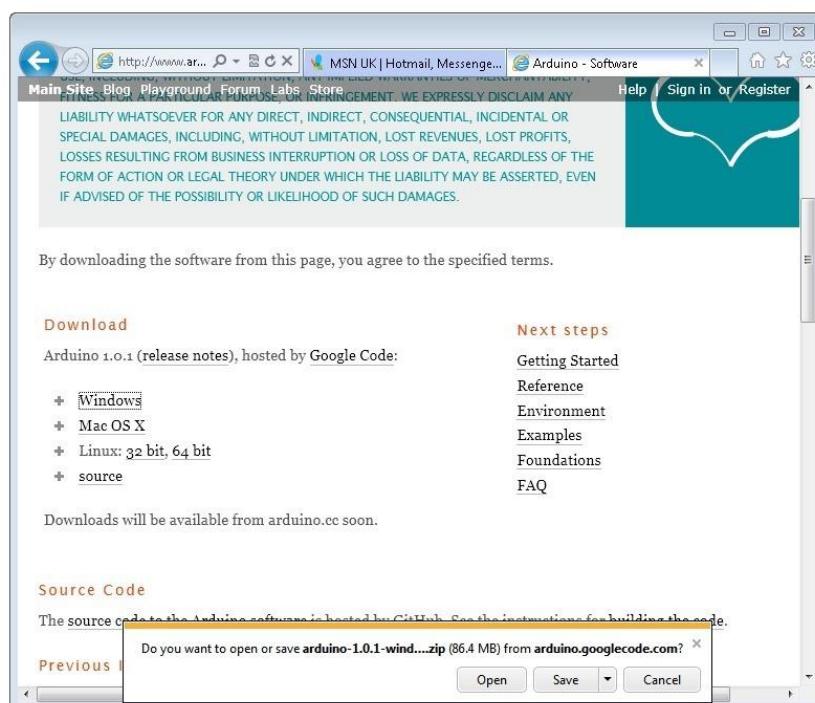
El software de Arduino que va a utilizar para programar el Arduino está disponible para Windows, Mac y Linux. El proceso de instalación es diferente para las tres plataformas y desafortunadamente hay una cierta cantidad de trabajo manual para instalar el software. No hay un programa de instalación; En su lugar, usted tiene que descomprimir un archivo que le da una carpeta de Arduino que contiene el programa de Arduino y algunos otros ítems.

En un paso separado, entonces debe instalar los controladores USB.

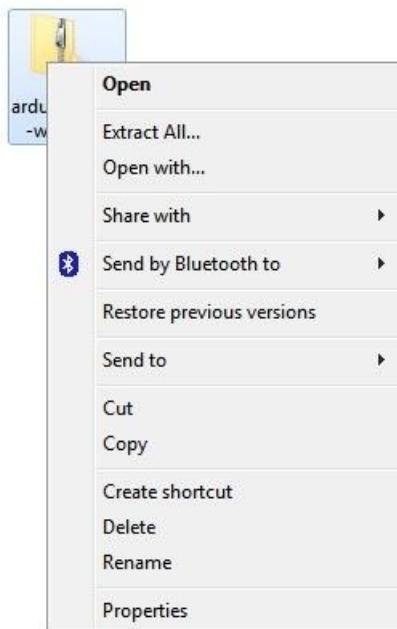
## Instalación de Arduino (Windows)

Conseguir comenzando visitando el sitio web Arduino.cc. A partir de abril de 2014, le sugerimos que utilice v1.0.5 como 1.5 es todavía en fase beta. Si ya no está en 1.5 beta es por el momento de leer esto, puede probarlo!

Comience por descargar el archivo zip para Windows. Sólo hay una versión del software, si está usando Windows XP a Windows 7.



Cuando el archivo zip ha descargado, extraiga el contenido en el escritorio, haciendo clic derecho sobre el archivo y seleccionar "Extraer todo ..." en el menú emergente.



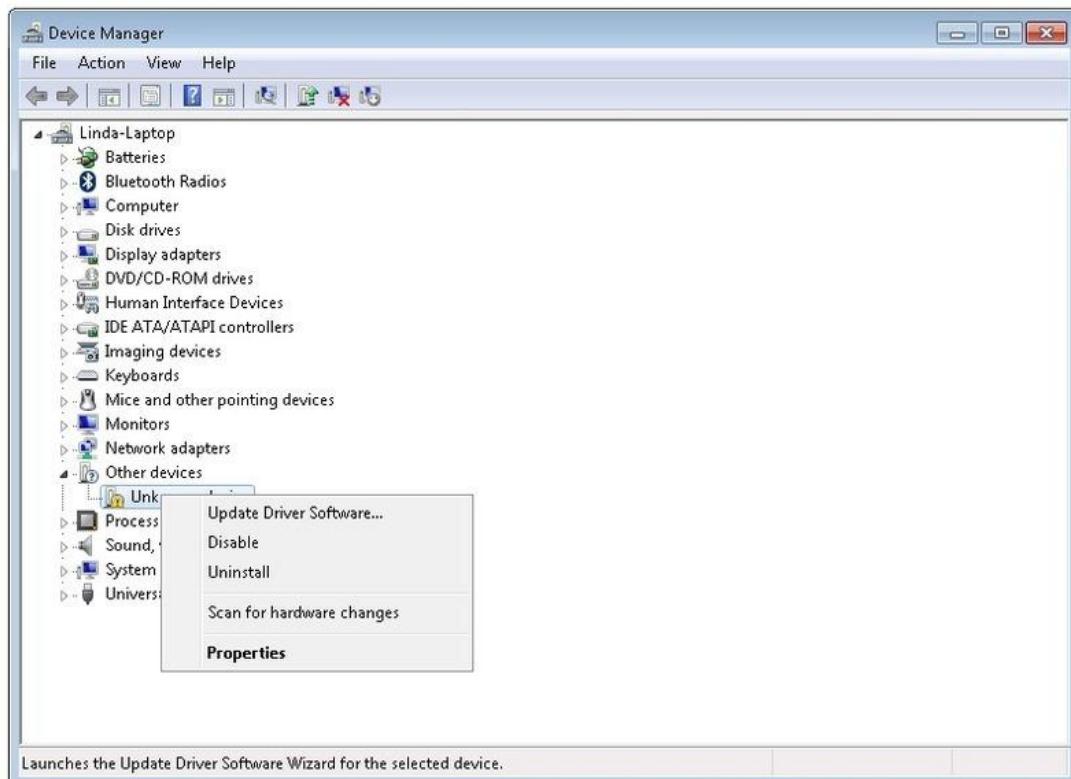
A continuación, seleccione el escritorio y haga clic en "Extraer". Puede moverlo a otro lugar en el equipo más tarde, con sólo mover la carpeta, pero por ahora, sólo mantenerlo en el escritorio.

La carpeta de Arduino contiene tanto el programa de Arduino sí mismo y los controladores que permiten el Arduino para ser conectado al ordenador mediante un cable USB. Antes de iniciar el software de Arduino, que se va a instalar los controladores USB.

Enchufe un extremo del cable USB en el Arduino y el otro en una toma USB de su ordenador. La luz de encendido en el LED se iluminará y se puede obtener un mensaje de "Nuevo Hardware Encontrado 'desde Windows. Ignorar este mensaje y cancelar cualquier intento que Windows hace para tratar de instalar controladores de forma automática para usted.

El método más fiable de la instalación de los controladores USB es utilizar el Administrador de Dispositivos. Esto se encuentra de diferentes maneras dependiendo de su versión de Windows. En Windows 7, primero hay que abrir el panel de control, a continuación, seleccione la opción de ver los iconos, y usted debe encontrar el Administrador de Dispositivos en la lista.

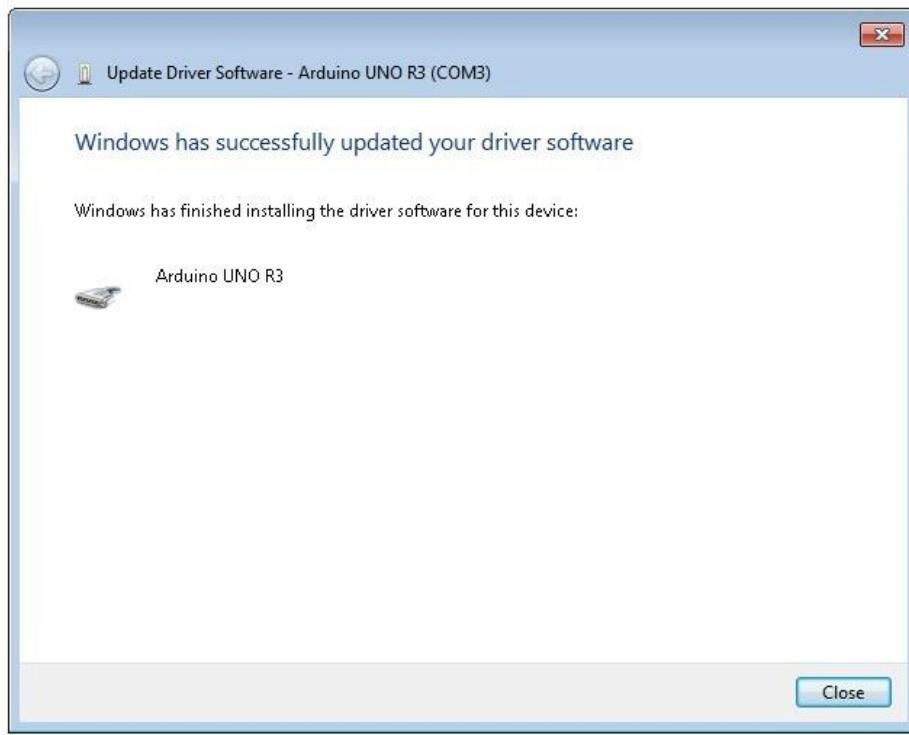
En "Otros dispositivos", debería ver un ícono de 'dispositivo desconocido' con un pequeño triángulo de advertencia amarillo junto a él. Esta es su Arduino.



Haga clic derecho en el dispositivo y seleccione la opción de menú superior (Actualizar software de controlador ...). A continuación se le pedirá que sea 'Buscar automáticamente el software de controlador actualizado "o" Examinar en mi ordenador para software de controlador '. Seleccionar la opción de explorar y navegar hasta el arduino-1.0.2-windows \ arduino1.0.2 \ drivers.



Haga clic en "Siguiente" y usted puede conseguir una advertencia de seguridad, si es así, permitir que el software se instale. Una vez que el software ha sido instalado, recibirá un mensaje de confirmación.



Eso es. Ahora está listo para la acción, así que evita la siguiente sección sobre la instalación en Mac y Linux y pasar directamente a 'Placas y Puertos'.

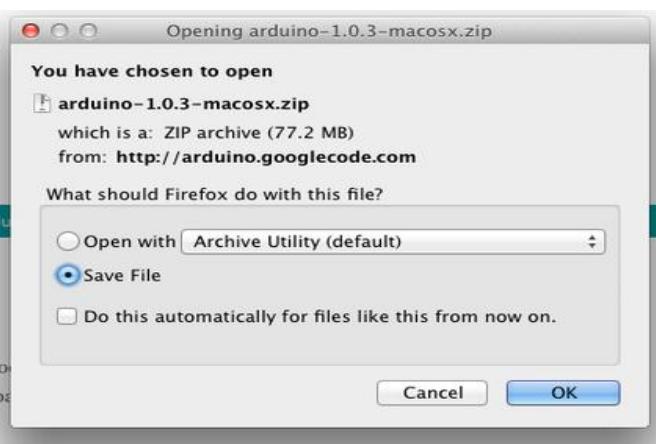
## Instalación de Arduino (Mac and Linux)

Conseguir comenzando visitando el sitio web Arduino.cc y descargar el IDE coincidente para su sistema operativo. A partir de abril de 2014, le sugerimos que utilice v1.0.5 como 1.5 es todavía en fase beta. Si ya no está en 1.5 beta es por el momento de leer esto, puede probarlo!

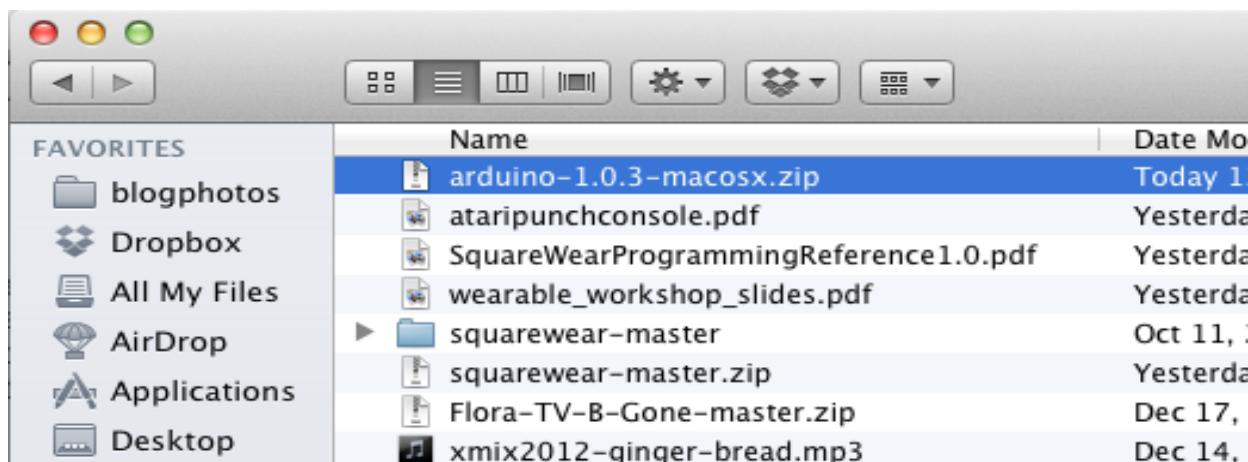


### Download the Arduino Software

The open-source Arduino environment makes it easy to write code, upload it to your Arduino board, and interact with sensors and actuators. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on the Processing software.

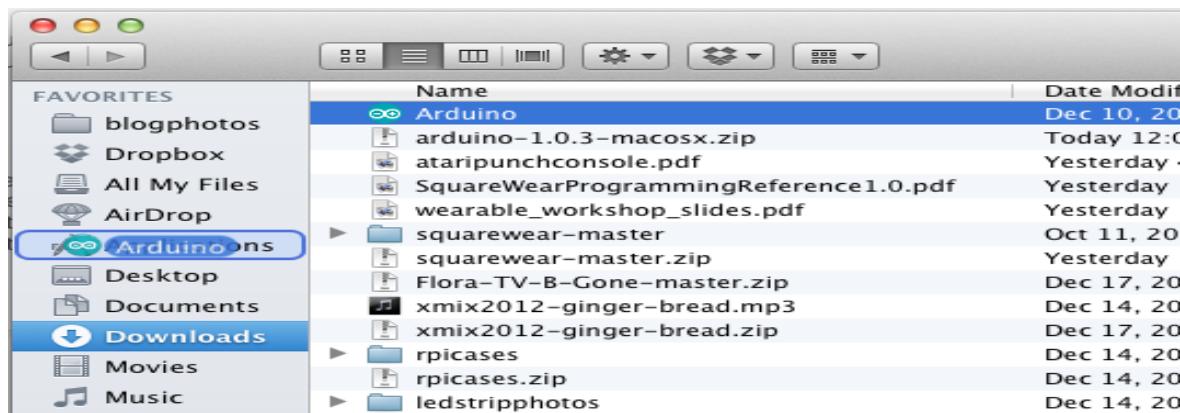


Guardar el software de instalación en el escritorio o en cualquier

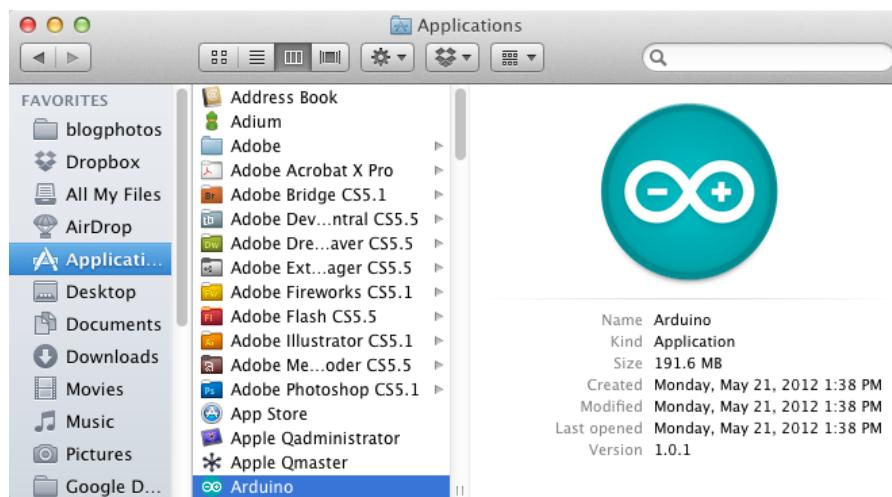


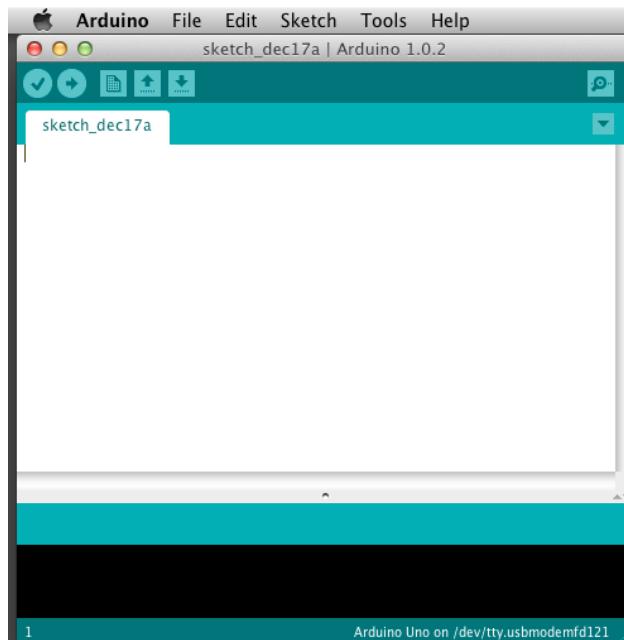
El proceso para instalar el software de Arduino en el Mac es mucho más fácil que en el PC.

Como antes, el primer paso es descargar el archivo. En el caso de Mac, es un archivo zip



Una vez descargado, haga doble clic en el archivo zip, que se va a extraer un único archivo llamado 'Arduino.app'. Esta es toda la aplicación de Arduino; simplemente arrastrarlo a la carpeta de aplicaciones.



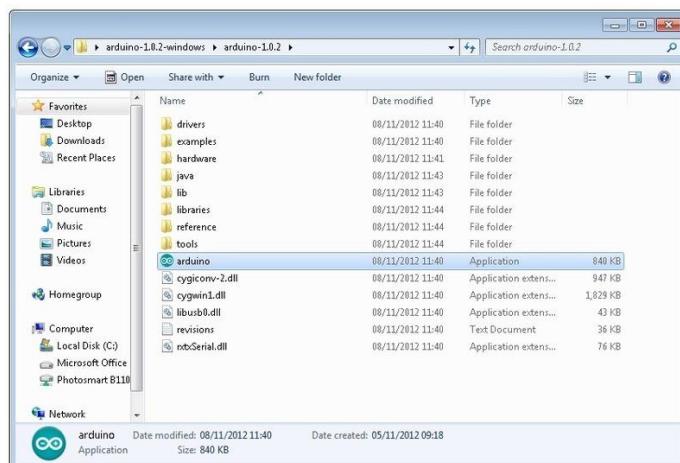


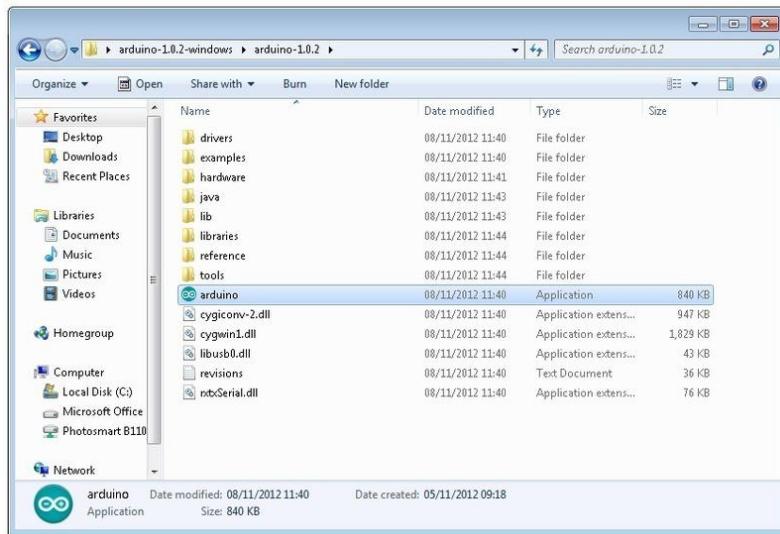
Ahora puede encontrar y poner en marcha el software de Arduino en la carpeta Aplicaciones. A medida que se va a utilizar con frecuencia, es posible que desee haga clic derecho en su ícono en dock y la puso a 'Mantener en el Dock'.

Hay muchos diferentes distribuciones de Linux y las instrucciones para cada distribución son un poco diferentes. La comunidad Arduino ha hecho un gran trabajo de reunir conjuntos de instrucciones para cada distribución.

## Placas y Puertos

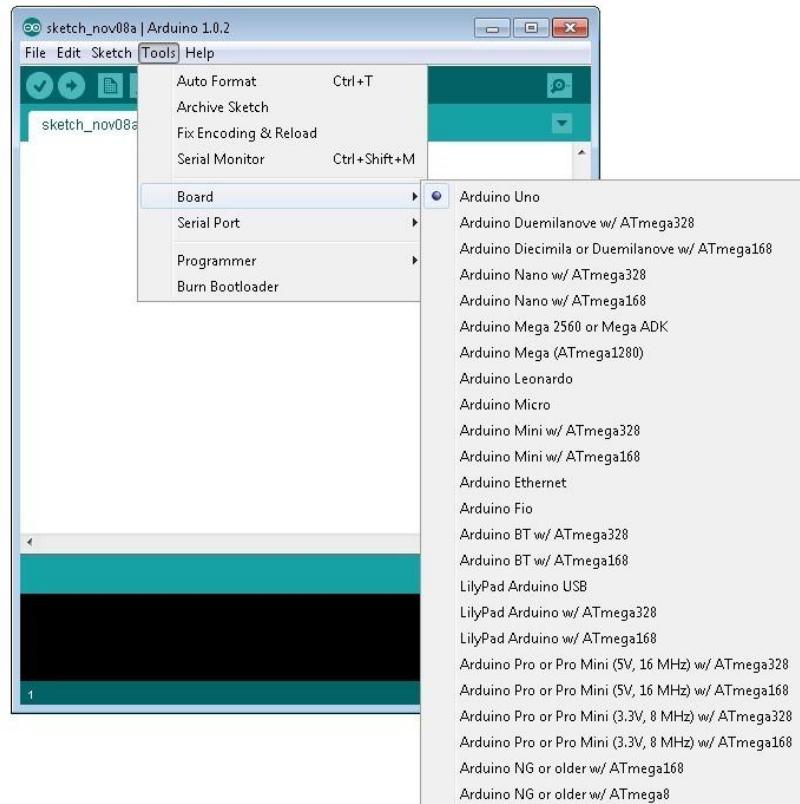
Ahora está listo para iniciar el software de Arduino, así que cualquier plataforma que está utilizando, abra la carpeta Arduino y abre la aplicación Arduino contenida dentro de ella.



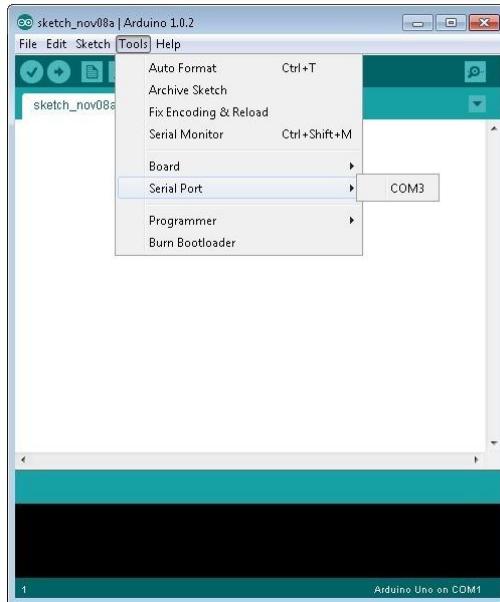


Esto iniciará el IDE Arduino, pero antes de poder iniciar la programación, hay que indicar al software de Arduino qué tipo de placa Arduino que está utilizando y también seleccionar el puerto al que se conecta.

A decir IDE Arduino qué tipo de placa que usted está utilizando. en el menú 'Herramientas', seleccione Placa y luego 'Arduino Uno' o 'Leonardo' , según sea apropiado.

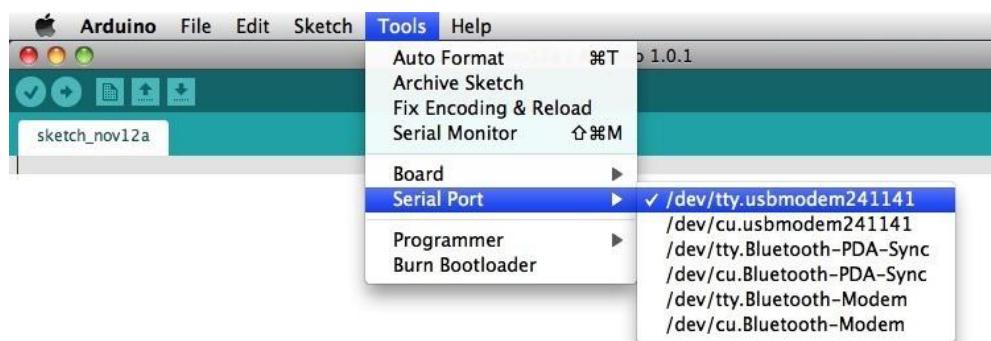


También en el menú 'Herramientas', se encuentra la opción 'Puerto Serial'. Seleccione esta opción.



Si está utilizando Windows, probablemente sólo habrá una opción aquí y será bien decir COM3 o COM4. A pesar de que sólo hay una opción, todavía tendrá que seleccionarlo.

Si está usando un Mac o Linux, habrá más opciones allí, pero lo más importante será la primera opción en la lista, ya que este será el dispositivo más recientemente enchufado. Esto es útil, como el nombre del puerto podrá no parecer que tenga nada que ver con Arduino. Es probable que se llame algo así como /dev/tty.usbmodemXXXX o /dev/ttyUSBn.



# Lección 1: Añadir Bibliotecas

Una vez que se sienta cómodo con el software de Arduino y el uso de las funciones incorporadas, es posible que desee ampliar su funcionalidad con bibliotecas adicionales.

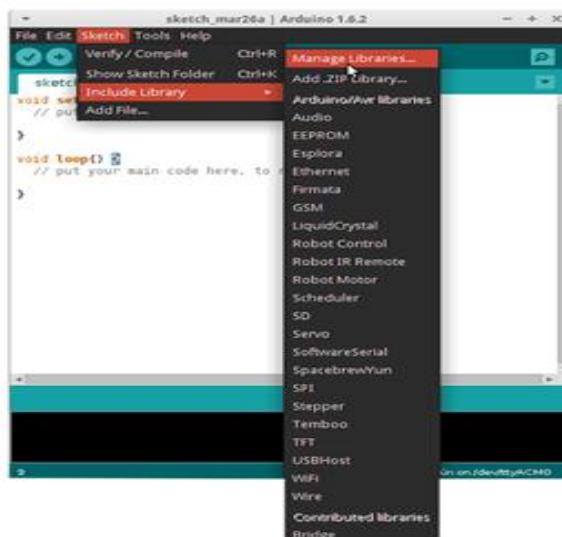
## ¿Cuáles son las bibliotecas?

Las bibliotecas son una colección de código que hace que sea fácil para que usted pueda conectarse a un sensor, pantalla, módulo, etc. Por ejemplo, la biblioteca de cristal líquido incorporada hace que sea fácil hablar con pantallas LCD de caracteres. Hay cientos de bibliotecas adicionales disponibles en Internet para su descarga. Las bibliotecas incorporadas y algunas de estas bibliotecas adicionales se enumeran en la referencia. Para utilizar estas bibliotecas adicionales, tendrá que instalarlos.

## Cómo instalar una Biblioteca

### Utilizando el Administrador de Biblioteca

Para instalar una nueva biblioteca en el IDE Arduino, puede utilizar el Administrador de Biblioteca (disponible en la versión 1.6.2 del IDE). Abrir el IDE y haga clic en Boceto> Incluir > Biblioteca> Administrar de bibliotecas.

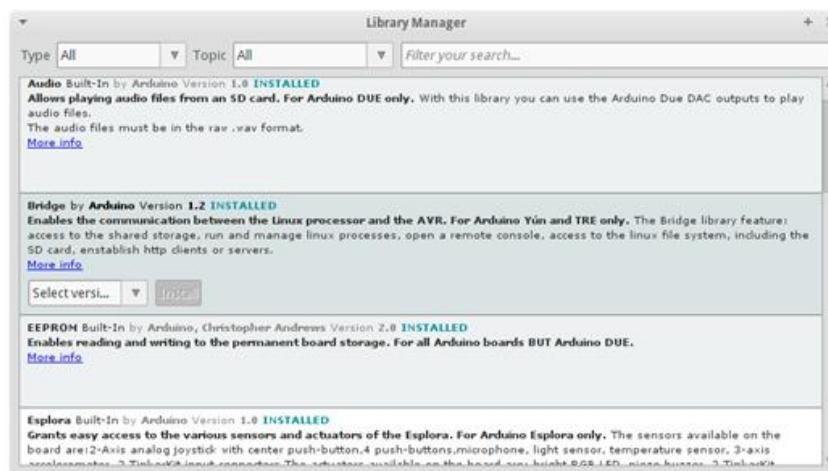


El administrador de la biblioteca se abrirá y usted encontrará una lista de las bibliotecas que ya están instalados o listos para su instalación. En este ejemplo, vamos a instalar la biblioteca de puente. Desplazarse por la lista para encontrarlo, a continuación, seleccione

la versión de la biblioteca que desea instalar. A veces, una sola versión de la biblioteca está disponible. Si no aparece el menú de selección de versiones, no se preocupe; es normal.



Por último, haga clic en Instalar y esperar a que el IDE para instalar la nueva biblioteca. La descarga puede tardar dependiendo de su velocidad de conexión. Una vez que haya terminado, una etiqueta instalada debe aparecer junto al puente de la biblioteca. Puede cerrar el administrador de bibliotecas.

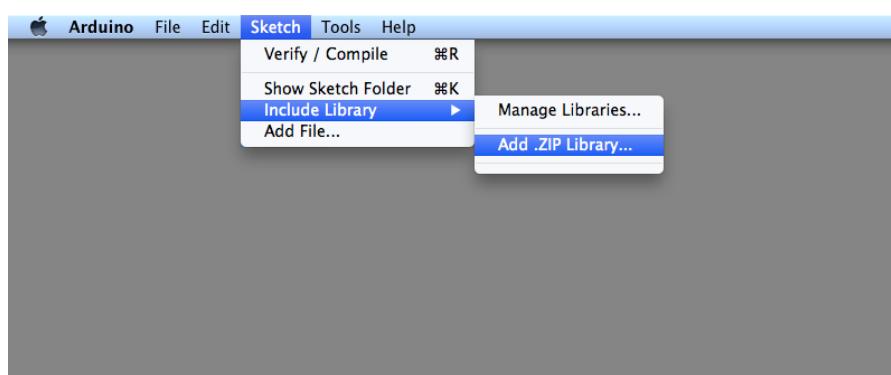


Ahora puede encontrar la nueva biblioteca disponible en el menú Biblioteca Incluido. Si desea añadir su biblioteca, abrir un nuevo tema en GitHub.

## Importando una biblioteca de .zip

Las bibliotecas se distribuyen a menudo como un archivo ZIP o carpeta. El nombre de la carpeta es el nombre de la biblioteca. Dentro de la carpeta será el siguiente: archivo de .cpp, archivo de .h, a menudo un palabra clave .archivo de txt, carpeta de ejemplos, y otros archivos requeridos por la biblioteca. Desde la versión 1.0.5, puede instalar las bibliotecas de terceros en el IDE. No descomprima la biblioteca descargado; dejarlo como está.

En el IDE de Arduino, navegar en boceto> Incluir Biblioteca> Añadir Biblioteca .ZIP.



Se le solicitará que seleccione la biblioteca que desea añadir. Navegue hasta la localización del archivo .zip y abrirlo.

Volver al menú de Boceto > Importar biblioteca. Ahora debería ver la biblioteca en la parte inferior del menú desplegable. Está listo para ser utilizado en su boceto. El archivo zip se han ampliado en la carpeta de bibliotecas en su directorio de bocetos de Arduino.

NB: La Biblioteca estará disponible para su uso en bocetos, pero los ejemplos de la biblioteca no se muestra en el Archivo> Ejemplos hasta después de DIE se ha reiniciado.

## Instalación manual

Para instalar la biblioteca, en primer lugar, salir de la aplicación de Arduino. A continuación, descomprima el archivo ZIP que contiene la biblioteca. Por ejemplo, si está instalando una biblioteca llamada "Parte de Arduino", descomprimir Parte de Arduino.zip. Debe contener una carpeta llamada Parte de Arduino, con los archivos como Parte de Arduino.cpp y Parte de Arduino.h interior. (Si los archivos .cpp y archivos.h no están en una carpeta, tendrá que crear una. En este caso, que serías una carpeta llamada " Parte de Arduino" y se mueven en él todos los archivos que estaban en la postal presentar, comoParte de Arduino.cpp yParte de Arduino.h.)

Arrastre la carpeta de Parte de Arduino en esta carpeta (carpeta de bibliotecas). En Windows, es probable que se llame "Mis documentos \ \bibliotecas de Arduino". Para usuarios de Mac, es probable que se llama "Documentos / Arduino / bibliotecas". En Linux, será la carpeta de "bibliotecas" en su cuaderno de bocetos.

Su carpeta de la biblioteca de Arduino ahora debería tener este aspecto (en Windows):

My Documents\Arduino\libraries\ArduinoParty\ArduinoParty.cpp

My Documents\Arduino\libraries\ArduinoParty\ArduinoParty.h

My Documents\Arduino\libraries\ArduinoParty\examples

....

o así (en Mac y Linux):

Documents/Arduino/libraries/ArduinoParty/ArduinoParty.cpp

Documents/Arduino/libraries/ArduinoParty/ArduinoParty.h

Documents/Arduino/libraries/ArduinoParty/examples

....

Es posible que haya más archivos que sólo los archivos de .cpp y los archivos de .h así que asegúrese de que están todos allí. (La biblioteca no funcionará si pones los archivos de .cpp y los archivos de .h directamente en carpeta delas bibliotecas o si están anidados en una carpeta adicional. Por ejemplo: Documentos \ Arduino \ bibliotecas \ Parte de Arduino.cpp y Documentos \ Arduino \ bibliotecas \ Parte de Arduino \ Parte de Arduino \ Parte de Arduino.cpp no funcionarán.)

Reinicie la aplicación de Arduino. Asegúrese de que la nueva biblioteca aparece en el Menú de Boceto> Importar biblioteca. Eso es. Usted ha instalado una biblioteca!

## Resumen

**En esta lección, vamos a instalar todas las bibliotecas que vamos a utilizar en el tutorial. Abra la carpeta bibliotecas e instalar los archivos ZIP uno por uno por lo que no necesita hacer este paso en los siguientes lecciones. Nosotros sólo tiene que conectar el componente como el boceto y cargar el código proporcionado. A continuación, el kit funcionará.**

# Lección 2: Parpadeo

## Visión general

En esta lección, usted aprenderá cómo el programa de su placa controladora de UNO R3 para hacer el parpadeo LED incorporado de Arduino.

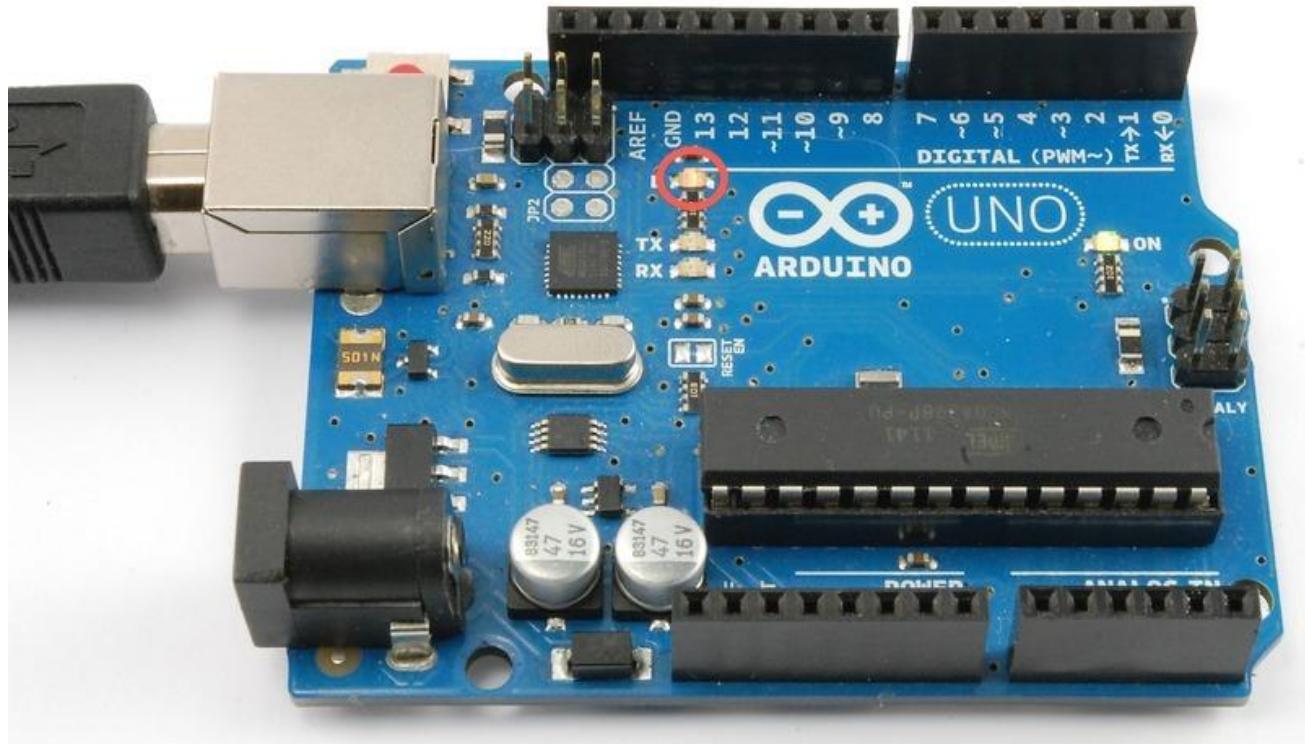
## Componente requerido

(1) Elegoo UNO R3

## Principio

La placa UNO R3 tiene filas de conectores a lo largo de ambos lados que se utilizan para conectarse a varios dispositivos electrónicos y enchufe en "escudos" que se extiende su capacidad.

También tiene un solo LED que se puede controlar desde sus bocetos. Este LED está construido en la placa UNO R3 y se refiere a menudo como la "L" LED ya que es la forma en que se marca en la placa.



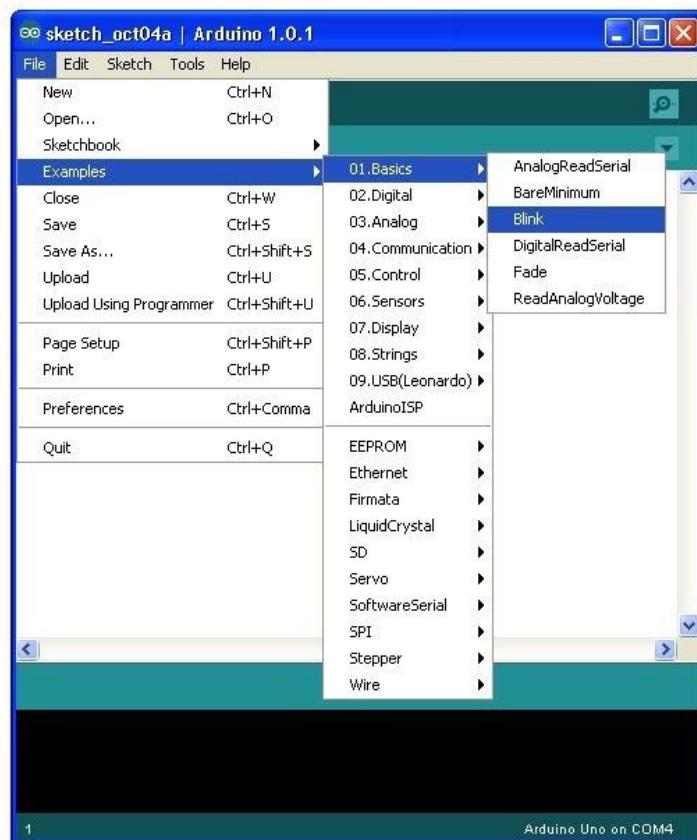
Usted puede encontrar que su placa de UNO R3 'L' LED ya parpadea cuando se conecta a un conector de USB. Esto se debe a que las placas se envían generalmente con el boceto del 'Parpadeo' pre-instalado.

En esta lección, vamos a reprogramar la placa de UNO R3 con nuestro propio boceto de parpadeo y luego cambiar la velocidad a la que parpadea.

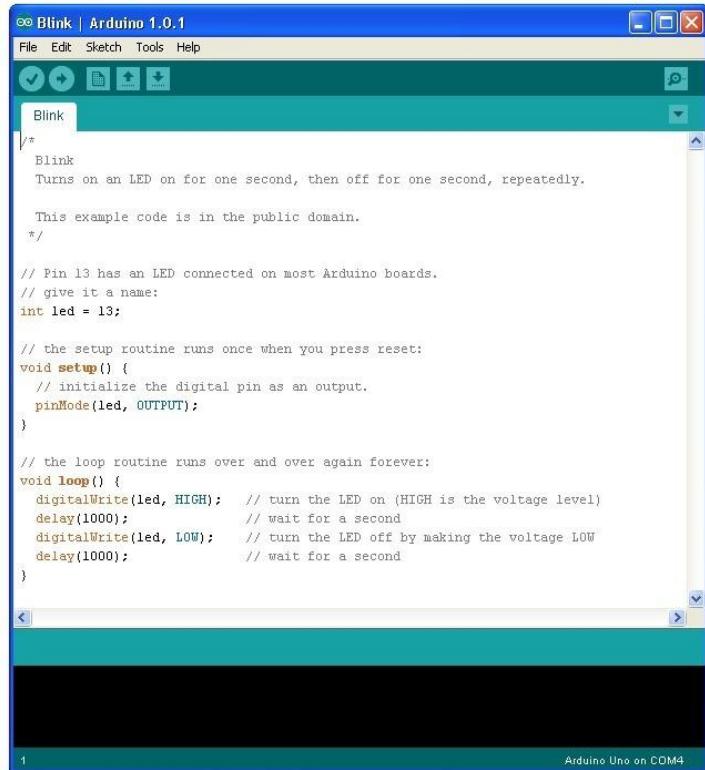
En lección 0, se configura el IDE de Arduino y se aseguró de que se pueda encontrar el puerto serial correcto para que se conecte a su placa de UNO R3. El tiempo ha llegado para poner esa conexión con la prueba y programar su placa de UNO R3.

El IDE de Arduino incluye una gran colección de bocetos ejemplo que puede cargar y utilizar. Esto incluye un boceto de ejemplo para hacer 'L' LED parpadea.

Cargar el boceto 'parpadeo' que se encuentra en el sistema de menús del IDE en Archivo> Ejemplos> 01. Básico



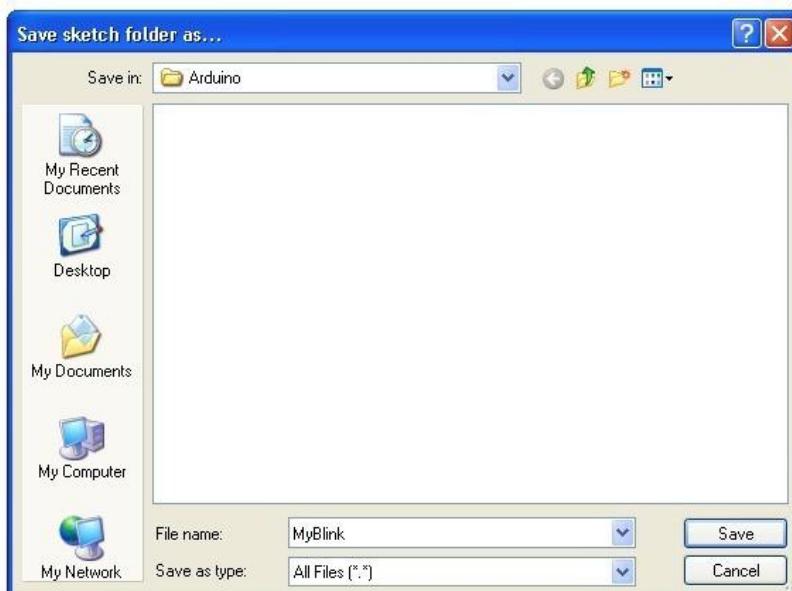
Cuando se abra la ventana de boceto, aumentarla de manera que se puede ver todo el boceto en la ventana.



Los bocetos de ejemplo incluidos con el IDE de Arduino son 'sólo lectura'. Es decir, usted puede cargarlos en una placa de UNO R3, pero si cambia de ellos, no se puede guardarlos como el mismo archivo.

Desde que vamos a cambiar este boceto, lo primero que hay que hacer es guardar su propia copia.

En el menú Archivo en el IDE de Arduino, selecciona "Guardar como .." y guarde el boceto con el nombre "Mi parpadeo '.

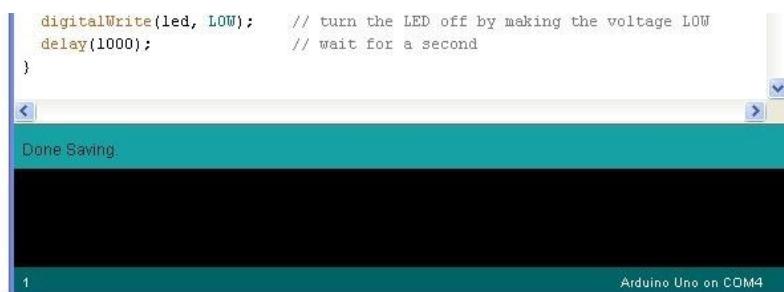


Ha guardado su copia de 'parpadeo' en su cuaderno de bocetos. Esto significa que si alguna vez quiere volver a encontrarlo, sólo puede abrirlo utilizando la opción de menú Archivo> cuaderno de bocetos .



Coloque tu placa Arduino al ordenador con el cable USB y comprobar que el "Tipo de placa" y "Puerto Serial" se han establecido correctamente. Puede que tenga que referirse de nuevo a lección 0.

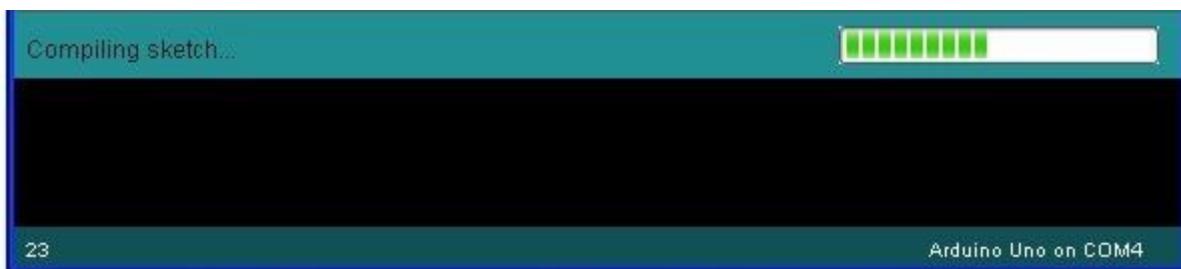
El IDE de Arduino le mostrará los valores actuales de la junta en la parte inferior de la ventana.



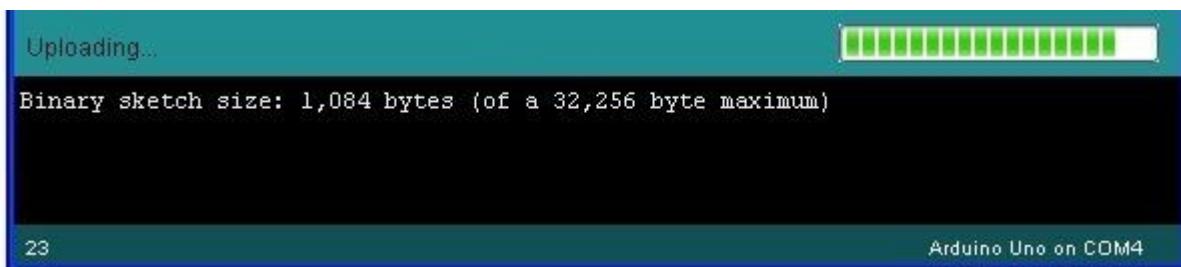
Haga clic en el botón "Cargar". Es el segundo botón de la izquierda en la barra de herramientas.



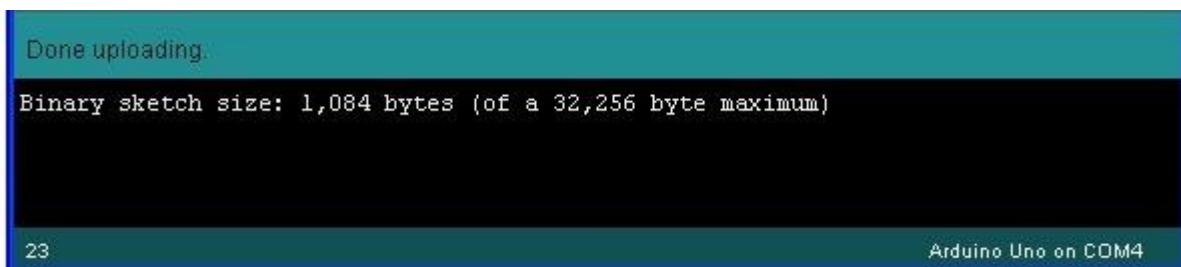
Si usted mira el área de estado de la IDE, verá una barra de progreso y una serie de mensajes. En un primer momento, dirá 'Compilación de Boceto ...'. Esto convierte el boceto en un formato adecuado para la carga a la placa.



A continuación, el estado cambiará a 'Cargando'. En este punto, el LED en el Arduino debería comenzar a parpadear como el boceto se transfiere.



Por último, el estado cambiará a "Carga Completa ". Aparecerá un mensaje que indica que el boceto está utilizando 1.084 bytes de los 32.256 disponibles.



Tenga en cuenta que durante el escenario de "Compilación de Boceto ... ", puede recibir el siguiente mensaje de error:



Puede significar que la tarjeta no está conectada en absoluto, o no se han instalado los controladores (si es necesario) o que se ha seleccionado el puerto serial incorrecto.

Si se encuentra con esto, volver a lección 0 y controlar la instalación.

Una vez que la carga ha finalizado, la junta debe reiniciar y empezar a parpadear.

[Abrir el código](#)

Tenga en cuenta que una gran parte de este boceto se compone de los comentarios. Estas no son las instrucciones reales del programa; más bien, que acaba de explicar cómo funciona el programa. Ellos están ahí para su beneficio.

Todo entre `/*` y `*/` en la parte superior del boceto es un comentario de bloque; lo que explica el boceto es para.

Los comentarios de una sola línea comienzan con `//` y todo hasta el final de la línea se considera un comentario.

La primera línea de código es:

Código de Copia

1. `int led = 13;`

Como el comentario anterior se explica, esto da un nombre a la clavija de que el LED está conectada. Esto es 13 en la mayoría de Arduino, entre ellos la UNO y Leonardo.

A continuación, tenemos la función 'configuración'. Una vez más, como dice el comentario, este se ejecuta cuando se pulsa el botón de reinicio. También se ejecuta cada vez que la placa se reinicia por ninguna razón, como la energía primero se aplican a él, o después de un boceto se ha cargado.

Código de Copia

1. `configuración vacía() {`  
2. `// inicializar el perno digital como una salida.`  
3. `Modo de perno(led, SALIDA);`  
4. `}`

Cada boceto de Arduino debe tener una función de 'configuración', y el lugar donde es posible que desee agregar instrucciones de su propia es entre el `{` y el `}`.

En este caso, sólo hay un comando de allí, lo cual, como dice el comentario le dice a la placa de Arduino que vamos a utilizar el perno de LED como salida.

También es obligatorio para un boceto de tener una función de 'bucle'. A diferencia de la función 'configuración' que sólo se ejecuta una vez, después de un reinicio, la función de 'bucle' será, después de que haya terminado de ejecutar sus órdenes, inmediatamente empezar de nuevo.

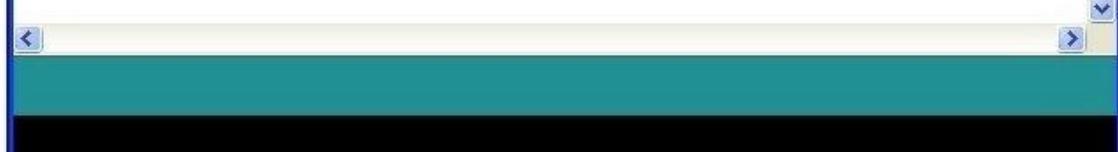
Código de Copia

```
1. bucle vacío () {  
2.   escritura digital (led, ALTA);      // encender el LED (ALTA es el nivel de tensión)  
3.   demorar (1000);                  // espera un segundo  
4.   escritura digital (led, BAJO);     // apagar el LED haciendo que la tensión BAJA  
5.   demorar (1000);                  // espera un segundo  
6. }
```

Dentro de la función de bucle, los comandos primero enciende el perno LED (ALTA), a continuación, "demorar" de 1000 milisegundos (1 segundo), luego apaga el perno LED y pausa de un segundo.

Ahora vamos a hacer que su LED parpadea más rápido. Como es de suponer, la diferencia reside en el cambio de parámetro en () para el comando "demorar".

```
// the loop routine runs over and over again forever:  
void loop() {  
    digitalWrite(led, HIGH);    // turn the LED on (HIGH is the voltage level)  
    delay(1000);              // wait for a second  
    digitalWrite(led, LOW);    // turn the LED off by making the voltage LOW  
    delay(1000);              // wait for a second  
}
```



Este período de retardo es en milisegundos, por lo que si desea que el LED parpadee dos veces más rápido, cambie el valor de 1000 a 500. Este sería entonces una pausa de medio segundo demora cada vez de un segundo entero.

Cargue el boceto de nuevo y debería ver la luz empieza a parpadear más rápidamente.

# Lección 3: LED

## Visión general

En esta lección, usted aprenderá cómo cambiar el brillo de un LED mediante el uso de diferentes valores de resistores.

## Componentes requeridos

(1) Elegoo UNO R3

(1) 5mm LED rojo

(1) 220 ohm resistor

(1) 1k ohm resistor

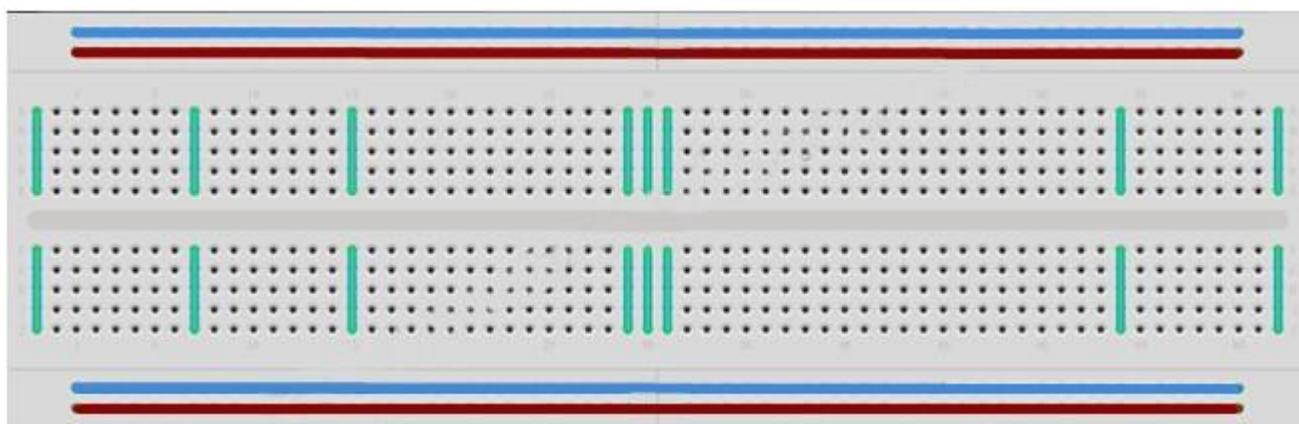
(1) 10k ohm resistor

(2) M-M alambres

## Introducción de Componente

### Placa universal MB-102:

La placa universal que permite a los circuitos de prototipo rápidamente, sin tener que soldar las conexiones. A continuación se muestra un ejemplo.



Las placas universales vienen en varios tamaños y configuraciones. El tipo más simple es sólo una rejilla de agujeros en un bloque de plástico. En el interior son tiras de metal que proporcionan conexión eléctrica entre los agujeros en las filas más cortas. Empujar las piernas de dos componentes diferentes en la misma fila los une eléctricamente. Un canal

profundo corriendo por el medio indica que hay una ruptura en las conexiones de allí, es decir, se puede insertar un chip en las piernas a cada lado de la canal sin conectarlos entre sí. Algunas placas universales tienen dos tiras de orificios que se ejecutan a lo largo de los bordes largos de la placa que se separa de la red principal. Estos tienen tiras corriendo por la longitud de la junta en el interior y proporcionar una manera de conectar una tensión común. Por lo general son en pares para +5 voltios y tierra. Estas tiras se conocen como los carriles y que le permiten conectar la alimentación de muchos componentes o puntos en la placa.

Mientras placas universales son ideales para la creación de prototipos, tienen algunas limitaciones. Debido a que las conexiones son de ajuste suave y temporal, que no son tan fiables como las conexiones soldadas. Si usted está teniendo problemas intermitentes con un circuito, que podría ser debido a una mala conexión en un circuito.

### **LED:**

LEDs hacen grandes luces indicadoras. Ellos utilizan muy poca electricidad y que prácticamente duran para siempre.

En esta lección, que va a utilizar tal vez el más común de todos los LEDs: un LED rojo de 5 mm. 5 mm se refiere al diámetro del LED. Otros tamaños comunes son de 3 mm y 10 mm.

No se puede conectar directamente a un LED a una batería o fuente de tensión debido a 1) el LED tiene un polo positivo y negativo y no se encenderá si se coloca el camino equivocado y 2) un LED debe ser utilizado con un resistor para limitar o 'estragulador' la cantidad de corriente que fluye a través de él; de lo contrario, ésta se quemará!



Si no se utiliza un resistor con un LED, entonces bien puede ser destruido casi inmediatamente, ya que demasiada corriente fluirá a través, calentándolo y la destrucción de la "unión", donde se produce la luz.

Hay dos maneras de decir cual es el cable positivo del LED y el que lo negativo.

En primer lugar, el cable positivo es más largo.

En segundo lugar, en el que el cable negativo entra en el cuerpo del LED, hay un borde plano para el caso de la LED.

Si le sucede que tiene un LED que tiene un lado plano junto a la preparación más largo, se debe asumir que el plomo ya es positivo.

### **RESISTORES:**

Como el nombre sugiere, los resistores resisten el flujo de electricidad. Cuanto mayor sea el valor del resistor, más se resiste y la corriente eléctrica menos fluirá a través de él. Vamos a usar esto para controlar la cantidad de electricidad fluye a través del LED y, por tanto, la forma en que brilla intensamente

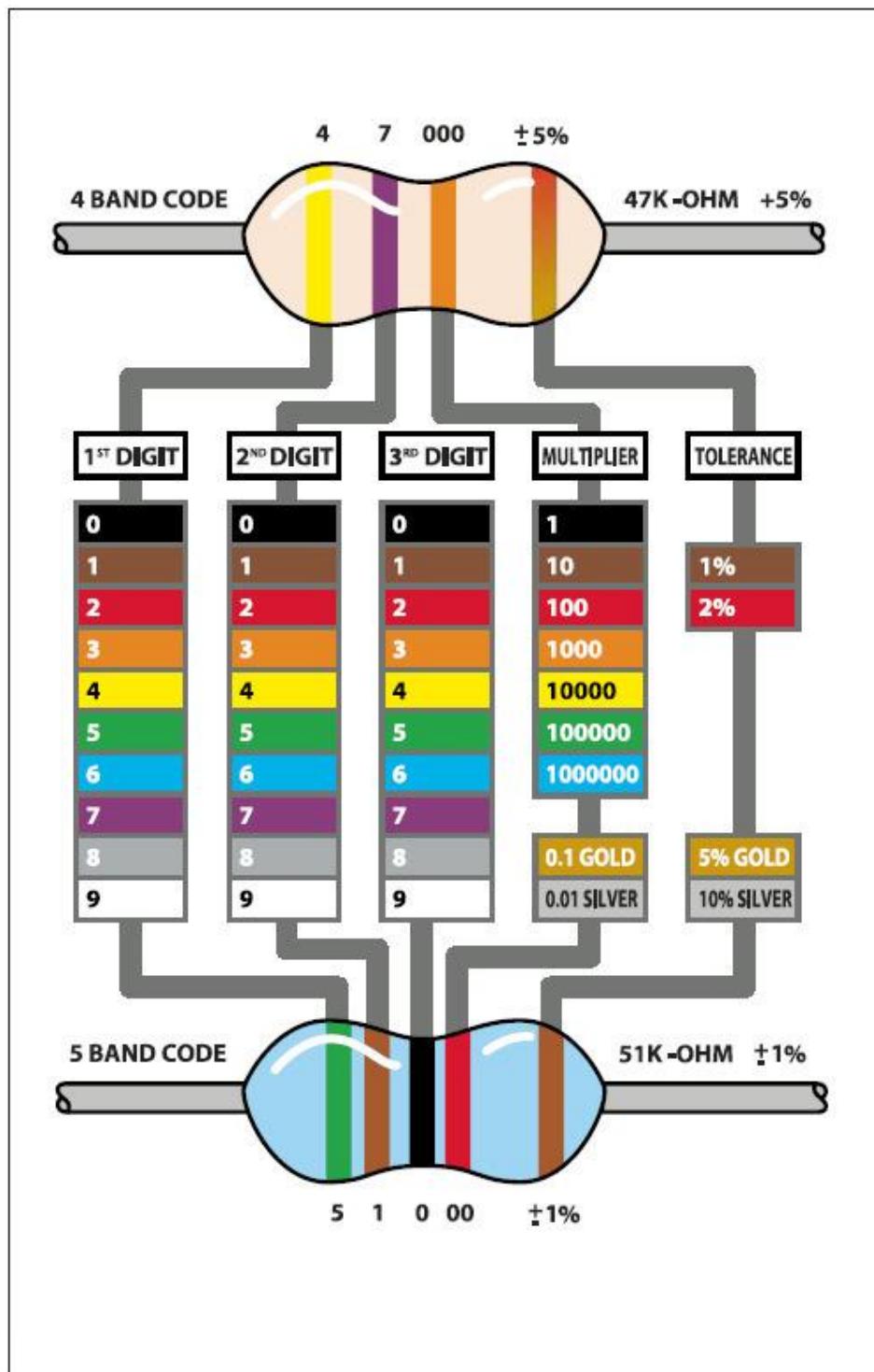


Pero en primer lugar, más sobre los resistores.

La unidad de resistor se llama Ohm, que por lo general se acorta a la letra griega  $\Omega$  como Omega. Debido a que un Ohm es un valor bajo de resistor (no resiste mucho a todos), también denotamos los valores de las resistencias en  $k\Omega$  (1.000  $\Omega$ ) y  $M\Omega$  (1.000.000  $\Omega$ ). Estos se llaman kilo-ohmios y mega-ohmios.

En esta lección, vamos a utilizar tres valores diferentes de resistores:  $220\Omega$ ,  $1k\Omega$  y  $10k\Omega$ . Todos estos resistores tienen el mismo aspecto, excepto que tienen diferentes franjas de colores en ellos. Estas rayas le indican el valor del resistor.

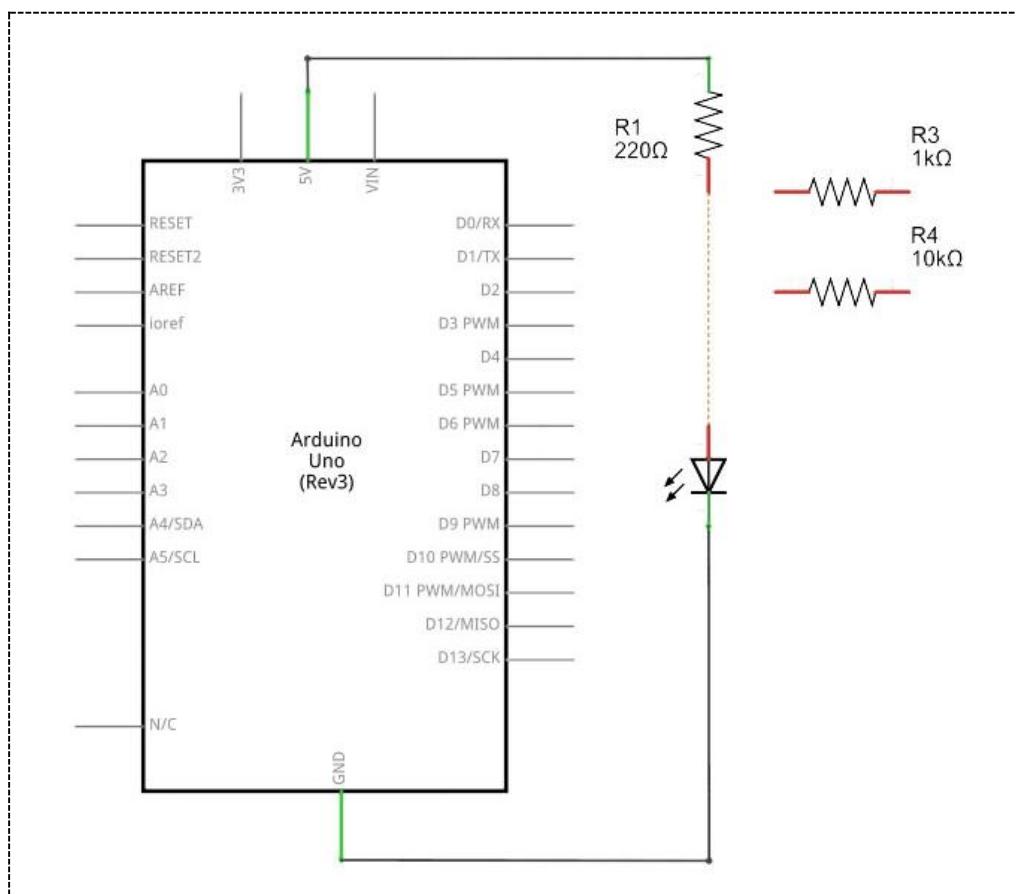
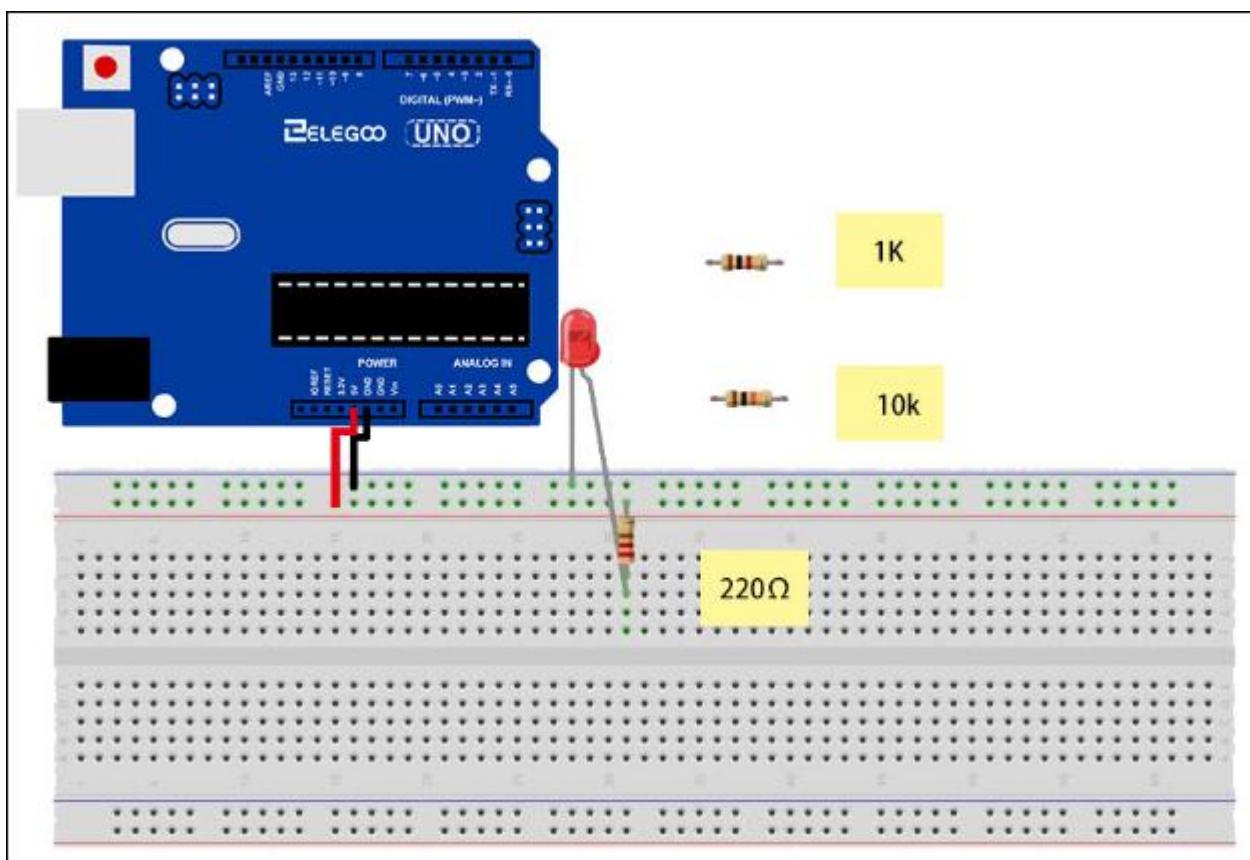
El código de color resistor tiene tres franjas de colores y, a continuación una banda de oro en un extremo.



A diferencia de los LEDs, los resistores no tienen un cable positivo y negativo. Se pueden conectar en cualquier dirección.

## Conexión

### Esquemático

**Diagrama de cableado**

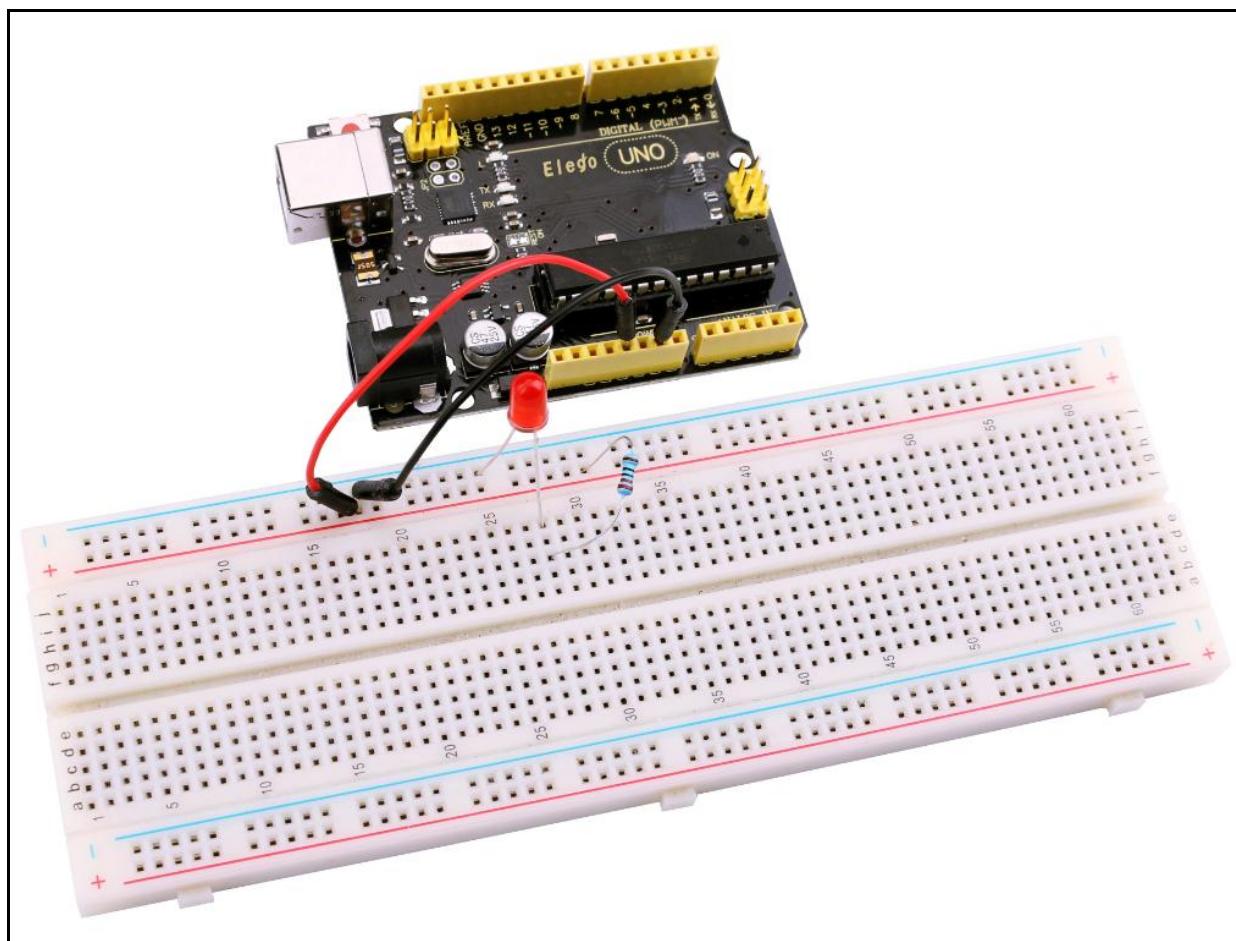
UNO es una fuente conveniente de 5 voltios, lo que vamos a utilizar para suministrar energía al LED y el resistor. Usted no tiene que hacer nada con su UNO, excepto para enchufarlo en un cable USB.

Con el resistor de  $220\ \Omega$  en su lugar, el LED debe ser muy brillante. Si se cambia el resistor de  $220\ \Omega$  para el resistor de  $1k\Omega$ , el LED aparecerá un poco el amortiguador. Finalmente, con el resistor de  $10\ k\Omega$  en su lugar, el LED estará casi visible. Tire del cable de puente rojo fuera del placa universal y toque en el agujero y eliminarlo, por lo que actúa como un interruptor. Usted sólo debe ser capaz de notar la diferencia.

Por el momento, tiene 5V ir a una pierna del resistor, la otra pierna del resistor va a la parte positiva del LED y el otro lado del LED va a GND. Sin embargo, si nos cambiamos el resistor de modo que se produjo después de que el LED, como se muestra a continuación, el LED se iluminará todavía.

Es probable que desee para poner el resistor de  $220\Omega$  en su sitio.

No importa de qué lado del LED ponemos el resistor, con tal de que está ahí en alguna parte.



# Lección 4: RGB LED

## Visión general

RGB LEDs son una manera divertida y fácil de añadir un poco de color a sus proyectos. Desde son como 3 LEDs regulares en uno, cómo utilizar y conectarlos no es muy diferente.

Vienen sobre todo en 2versiones: Ánode Común o Cátodo Común.

Ánode Común utiliza 5V en el perno común, mientras Cátodo Común se conecta a tierra.

El con cualquier LED, necesitamos conectar algunos resistores en línea (3 en total) para que podamos limitar la corriente está elaborando.

En nuestro boceto, vamos a comenzar con el LED en el estado de color Rojo, luego se desvanecen a Verde, luego se desvanecen a Azul y finalmente de vuelta al color Rojo. De esta manera vamos a recorrer la mayor parte de la cola de color se puede lograr.

## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) Placa Universal
- (4) M-M alambres
- (1) RGB LED
- (3) 220 ohm resistores

## Introducción de Componentes

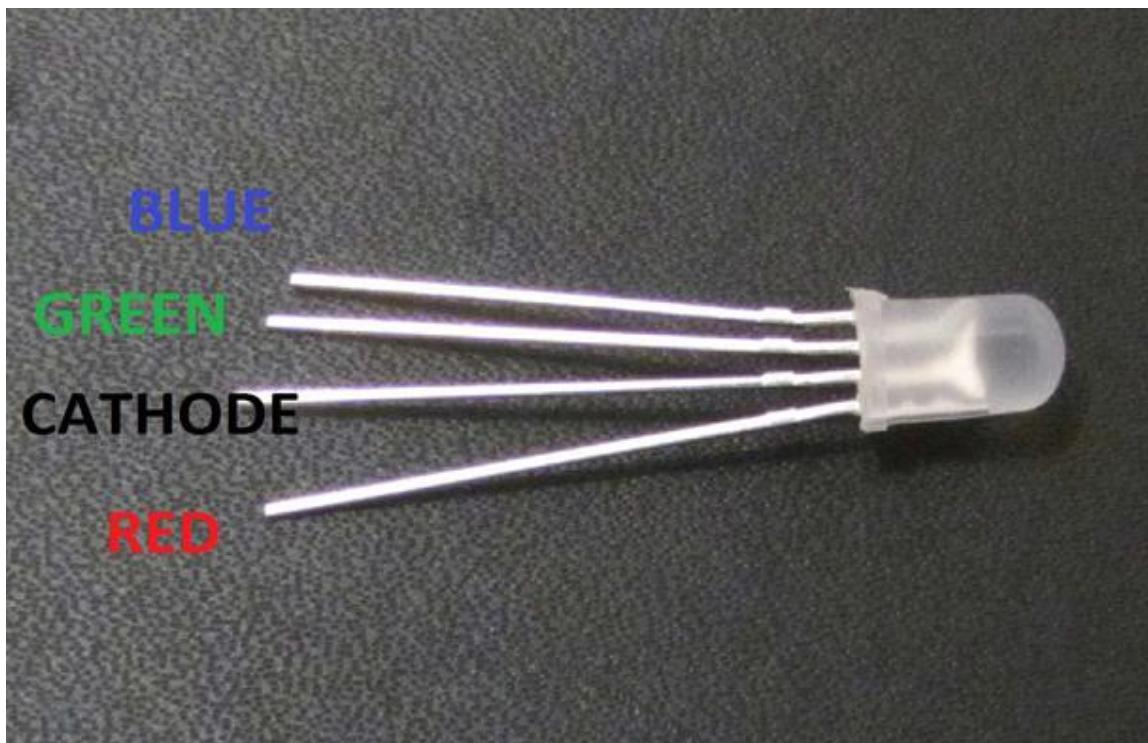
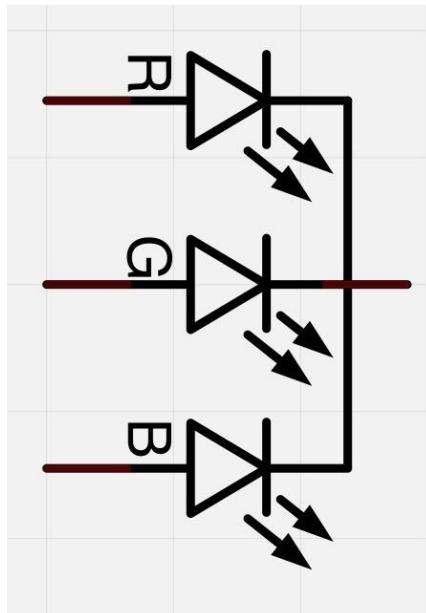
### RGB:

A primera vista, RGB (Rojo, Verde, Azul) LEDs se parecen a los LEDs regulares. Sin embargo, dentro del paquete de LED de costumbre, en realidad hay tres LED, uno rojo, uno verde y uno azul. Al controlar el brillo de cada uno de los LEDs individuales se pueden mezclar casi cualquier color que deseé.

Mezclamos colores de la misma forma que lo haría la mezcla de pintura en una paleta - ajustando el brillo de cada uno de los tres LEDs. La manera más difícil de hacer esto sería el uso de diferentes resistores de valor (o resistores variables) como lo hicimos con en lección 2, pero eso es un montón de trabajo! Afortunadamente para nosotros, la placa deUNO R3

tiene una función de escritura analógica que se puede utilizar con los pernos marcados con un ~ dar salida a una cantidad variable de energía a los LEDs correspondientes.

El RGB LED tiene cuatro conductores. Hay una ventaja de ir a la conexión positiva de cada uno de los LEDs individuales dentro del paquete y un solo plomo que está conectado a los tres lados negativos de los LEDs.



La conexión negativa común del paquete de LED es el segundo perno del lado plano. También es el más largo de los cuatro conductores y será conectado a la tierra.

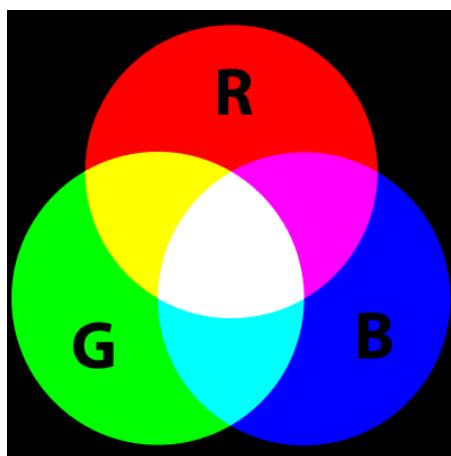
Cada LED dentro del paquete requiere su propia resistencia de  $270\Omega$  para evitar que la

corriente que fluye a través de él demasiado. Los tres conductores positivos de los LEDs (uno rojo, uno verde y uno azul) están conectados a los pernos de salida UNO utilizando estos resistores.

### COLOR:

La razón por la que usted puede mezclar cualquier color que te gusta variando las cantidades de luz roja, verde y azul es que el ojo tiene tres tipos de receptores de luz en él (rojo, verde y azul). Sus ojos y cerebro procesan las cantidades de rojo, verde y azul y convertirlo en un color del espectro.

En cierto modo, mediante el uso de los tres LED, estamos jugando un truco en el ojo. Esta misma idea se utiliza en televisores, en el que el LCD tiene puntos rojos, verdes y azules de color junto a la otra que componen cada píxel.



Si nos fijamos el brillo de los tres LED que ser el mismo, entonces el color general de la luz será blanco. Si nos desviamos el LED azul, de modo que sólo los LED rojo y verde son el mismo brillo, entonces la luz aparece de color amarillo.

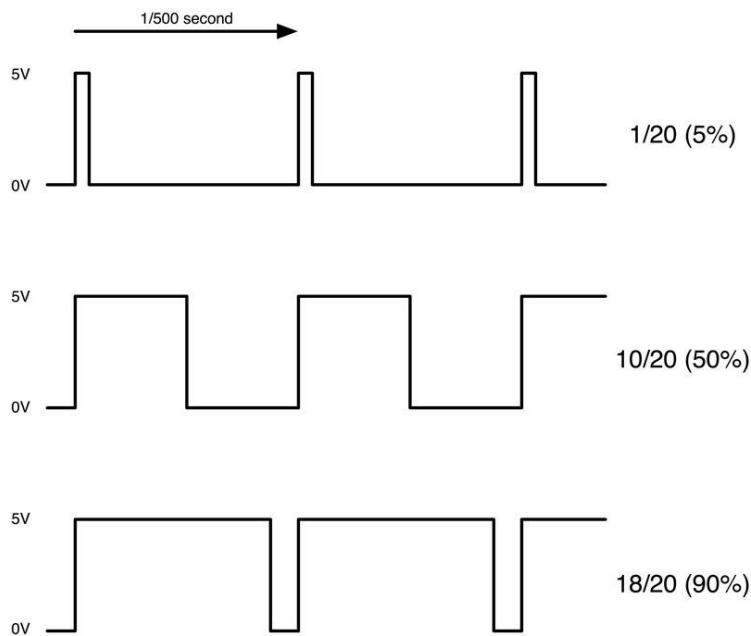
Podemos controlar el brillo de cada una de las partes rojas, verdes y azules del LED por separado, por lo que es posible mezclar cualquier color que nos gusta.

Negro no es tanto un color como una ausencia de luz. Por lo tanto, lo más cerca que podemos llegar a negro con nuestro LED es apagar los tres colores.

### Teoría PWM

Modulación Anchura de Pulso (PWM) es una técnica para controlar la potencia. También lo usamos aquí para controlar el brillo de cada uno de los LEDs.

El siguiente diagrama muestra la señal de uno de los pernos PWM en UNO



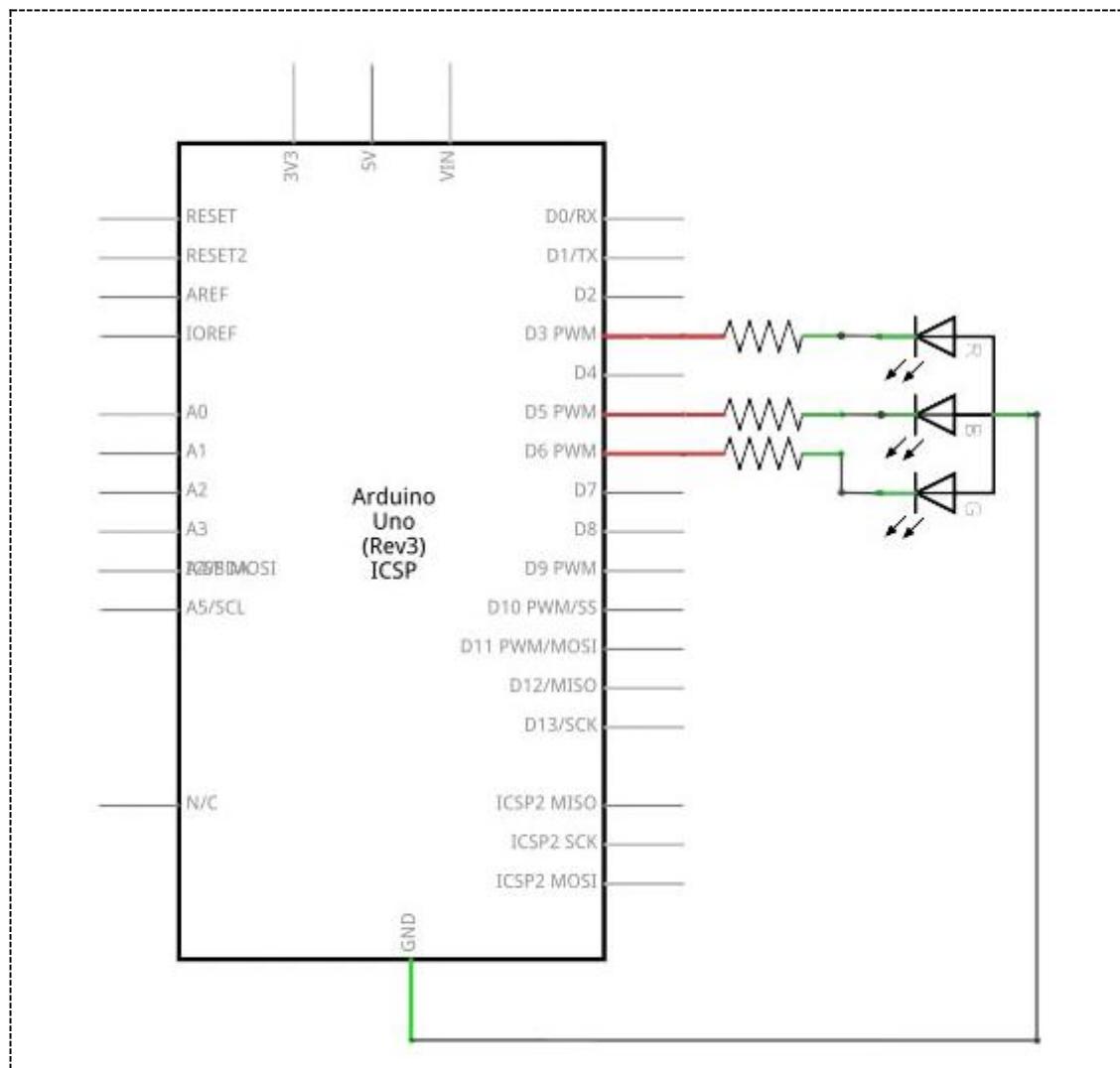
Aproximadamente cada 1/500 de segundo, la salida PWM producirá un pulso. La longitud de este pulso es controlada por la función de 'Escritura analógica'. Así 'Escritura analógica (0)' no producirá ningún pulso en absoluto y ' Escritura analógica (255)' producirá un pulso que tiene una duración de todo el camino hasta el siguiente impulso es debido, por lo que la salida es en realidad todo el tiempo.

Si se especifica un valor en la escritura analógica que está en algún lugar entre 0 y 255, a continuación, vamos a producir un pulso. Si el pulso de salida sólo es alta para el 5% de las veces, entonces lo que estamos manejando sólo recibirá el 5% de la potencia total.

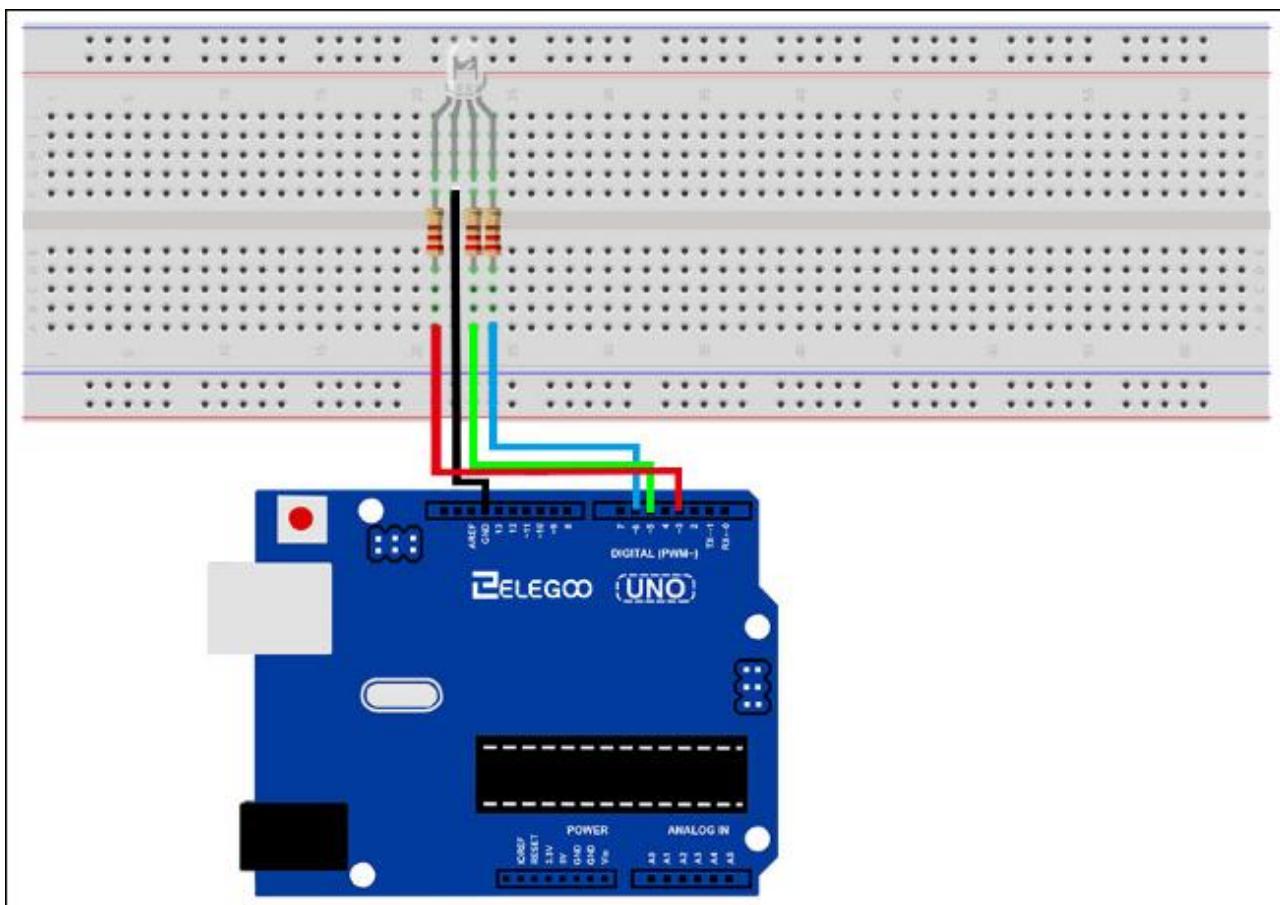
Si, sin embargo, la salida es a 5 V para 90% de las veces, a continuación, la carga se consigue 90% de la potencia entregada a la misma. No podemos ver los LED encendido y apagado a esa velocidad, por lo que a nosotros, que sólo se ve como el resplandor está cambiando.

## **Conexión**

### **Esquemático**



## Diagrama de cableado



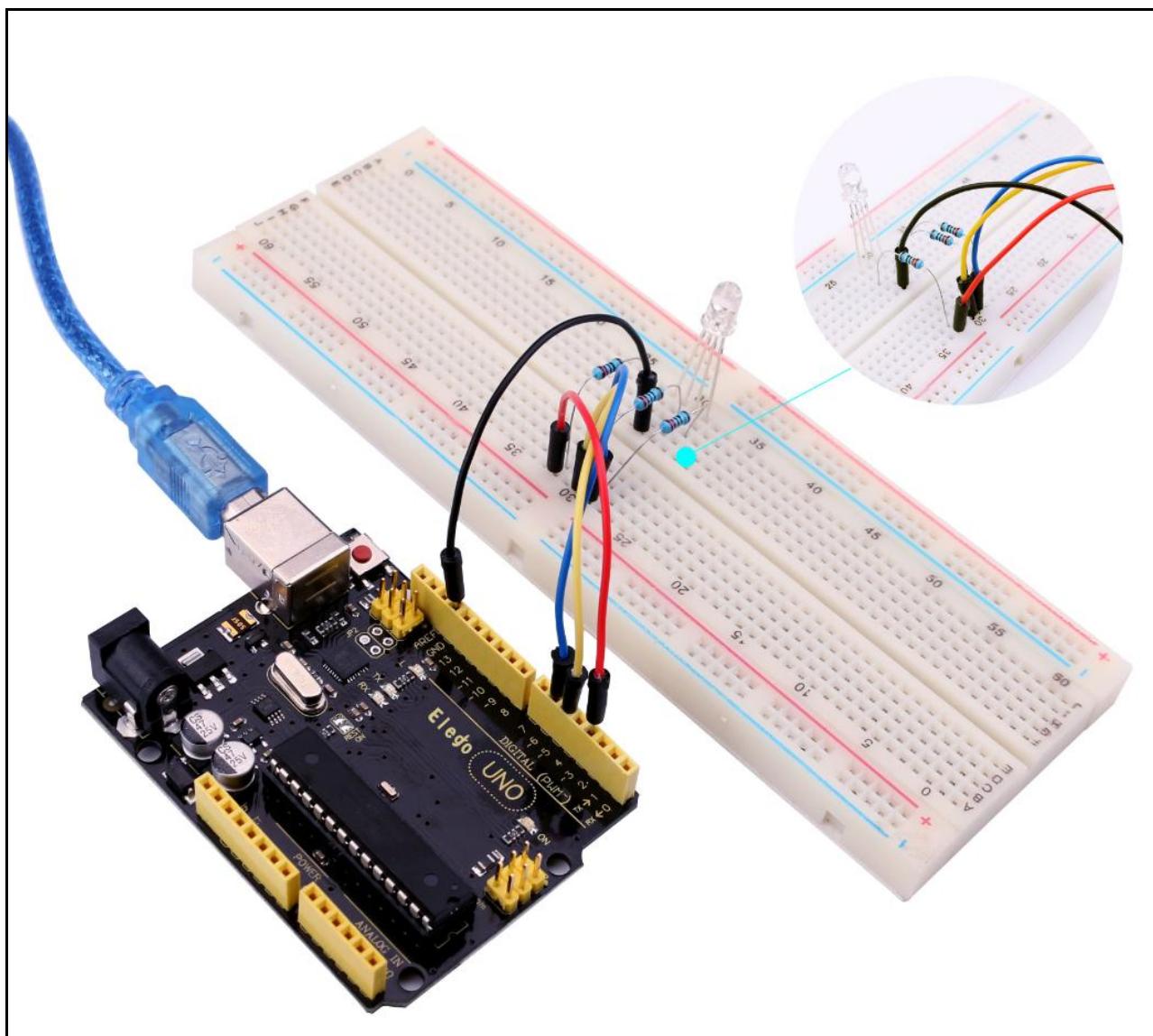
## Código

Nuestro código va a utilizar PARA bucles para desplazarse a través de los colores.

El primero PARA bucle pasará de ROJO a VERDE.

El segundo PARA bucle pasará de VERDE a AZUL.

El último PARA bucle pasará de AZUL a ROJO.



# Lección 5: Entradas Digitales

## Visión general

En esta lección, usted aprenderá a utilizar pulsadores con entradas digitales para activar un LED encendido y apagado.

Al pulsar el botón más cerca de la parte superior de la placa se enciende el LED; pulsar el otro botón apagará el LED.

## Componentes Requeridos

(1) Elegoo UNO R3

(1) Placa Universal

(1) 5mm LED rojo

(1) 220 ohm resistor

(2) Interruptores de empuje

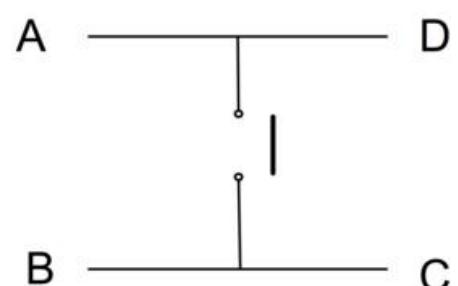
(6) M-M alambres

## Introducción de Componentes

### INTERRUPTORES DE EMPUJE:

Los interruptores son componentes muy simples. Cuando se pulsa un botón o una palanca da la vuelta, se conectan dos contactos de manera que la electricidad puede fluir a través de ellos.

Los pequeños interruptores táctiles que se utilizan en esta lección tienen cuatro conexiones, que puede ser un poco confuso.

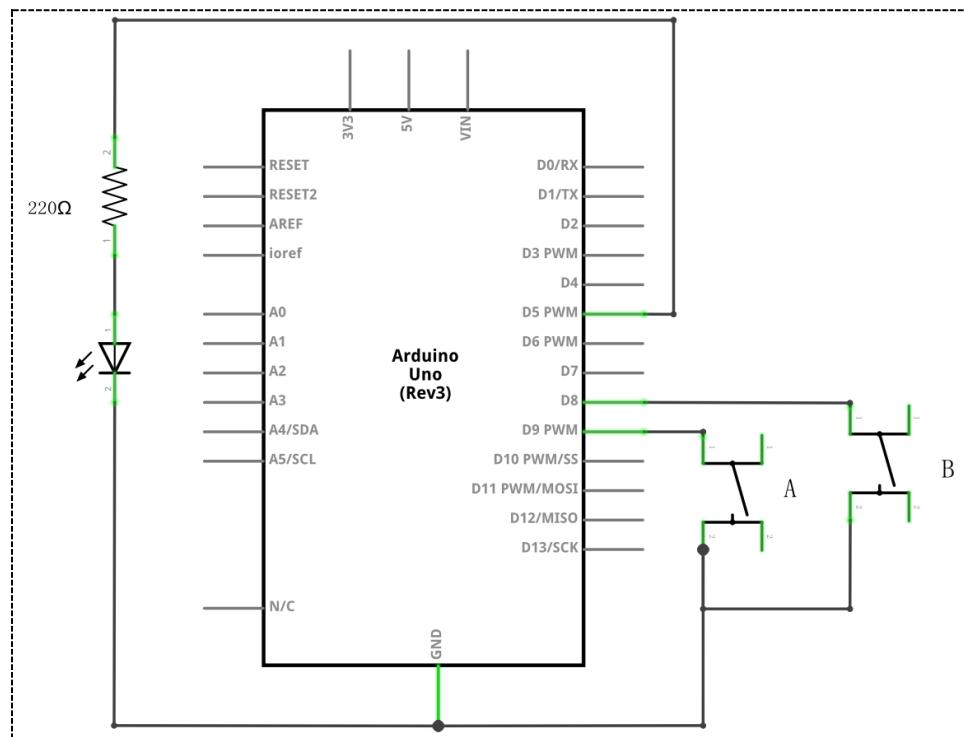


En realidad, sólo hay realmente dos conexiones eléctricas. Dentro del paquete de

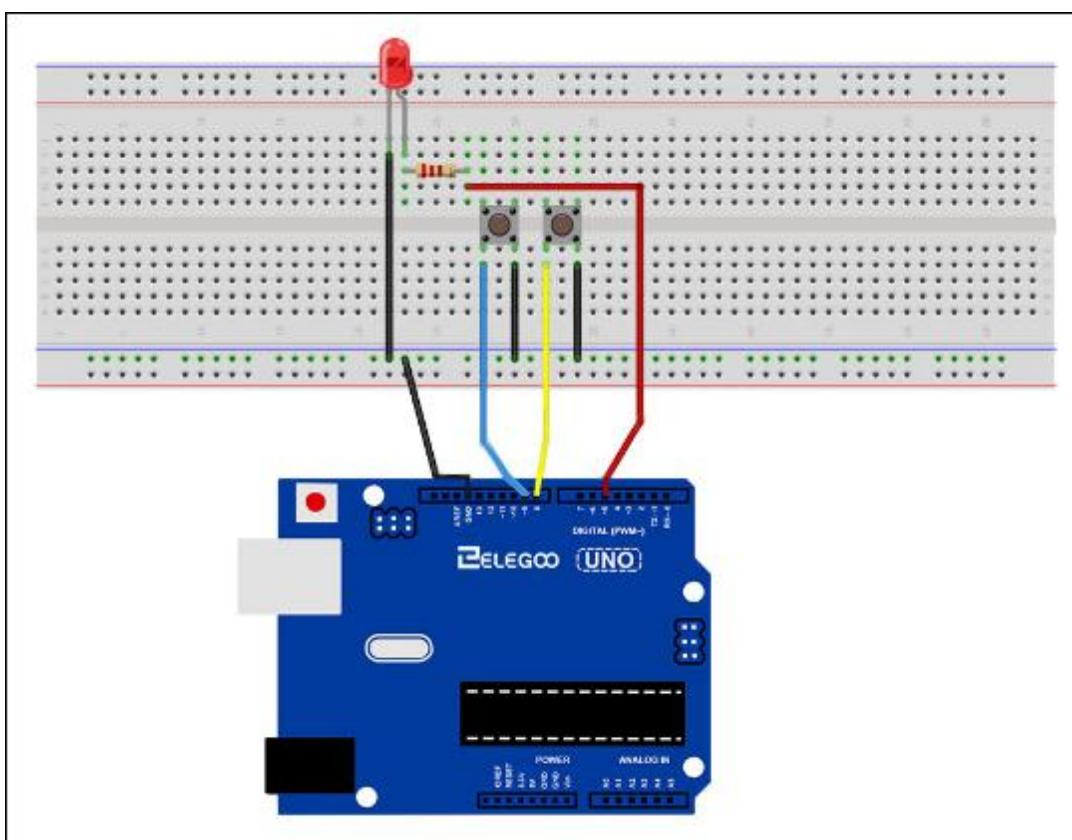
interruptores, pernos B y C están conectados entre sí, al igual que A y D.

## Conexión

### Esquemático



### Diagrama de cableado



Aunque los cuerpos de los interruptores son cuadrados, los pernos sobresalen de lados opuestos del interruptor. Esto significa que los pernos sólo serán lo suficientemente separados cuando se colocan correctamente en la placa.

Recuerde que el LED tiene que tener el cable negativo más corto hacia la derecha.

## Código

Cargue el boceto en su placa de UNO. Al pulsar el botón de arriba a su vez, el LED mientras pulsa el botón de abajo a lo apagará.

La primera parte del boceto define tres variables para los tres pernos que se van a utilizar. El 'Perno de led' es el pin de salida y 'perno de botón A' se referirán al interruptor más cerca de la parte superior de la placa y 'perno de botón B' para el otro interruptor.

La función de 'configuración' define el Perno de led como una SALIDA como normal, pero ahora tenemos las dos entradas de tratar. En este caso, se utiliza el conjunto de la Modo de perno ser 'ENTRADA\_PARAR' como esto:

```
pinMode(buttonApin, INPUT_PULLUP);
pinMode(buttonBpin, INPUT_PULLUP);
```

El modo de perno de ENTRADA\_PARAR significa que el perno se va a utilizar como una entrada, pero que si nada más se conecta a la entrada, que debe ser' estirado 'a ALTO. En otras palabras, el valor por defecto de la entrada es alta, a menos que se tira BAJA por la acción de pulsar el botón.

Es por esto que los interruptores están conectados a GND. Cuando se pulsa un interruptor, que conecta el perno de entrada a GND, de modo que ya no es ALTO.

Dado que la entrada es normalmente ALTA y sólo pasa a BAJA cuando se pulsa el botón, la lógica es un poco al revés. Vamos a manejar esto de la función de 'bucle'.

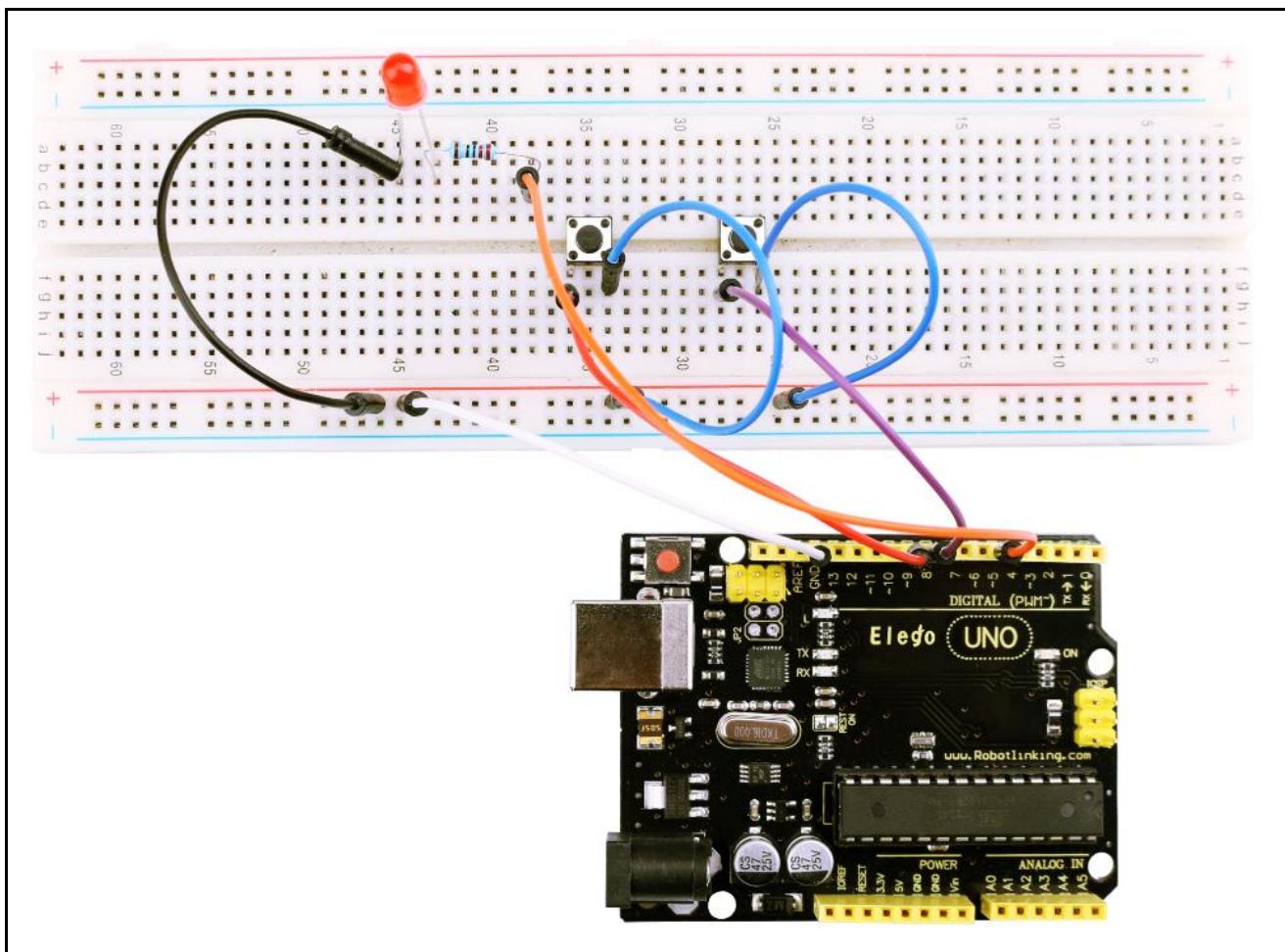
```
void loop()
{
if (digitalRead(buttonApin) == LOW)
{
digitalWrite(ledPin, HIGH);
}
if (digitalRead(buttonBpin) == LOW)
```

```
{  
  digitalWrite(ledPin, LOW);  
}  
}
```

En la función de 'bucle', hay dos sentencias "si" , uno para cada botón. Cada uno hace un "Lector digital 'en la entrada correspondiente.

Recuerde que si se pulsa el botón, la entrada correspondiente será BAJO. Si el botón A es bajo, entonces un "escritura digital 'en el perno de led enciende.

Del mismo modo, si se presiona el botón B, la mínima se escribe en perno de led.



# Lección 6: Ocho LEDs con 74HC595

## Visión general

En esta lección, usted aprenderá cómo utilizar ocho grandes LEDs rojos con un UNO sin necesidad de renunciar a 8 pernos de salida!

Aunque se puede cablear ocho LEDs, cada uno con un resistor a un perno de la UNO que le rápidamente comenzar a quedarse sin pernos en el UNO. Si usted no tiene un montón de cosas conectada a su UNO. Está bien para hacerlo - pero muchas veces queremos botones, sensores, servos, etc, y antes de saber que tienes no pernos dejaron. Así, en lugar de hacer eso, usted va a utilizar un chip llamado la Serie74HC595 a Convertidor Paralelo. Este chip tiene ocho salidas (perfecto) y tres entradas que se utilizan para alimentar datos en él un poco a la vez.

Este chip hace que sea un poco más lento para conducir los LED (solo puede cambiar los LED alrededor de 500.000 veces por segundo en lugar de 8.000.000 una segunda) pero aún así es muy rápido, mucho más rápido que los humanos pueden detectar, por lo que vale la pena!

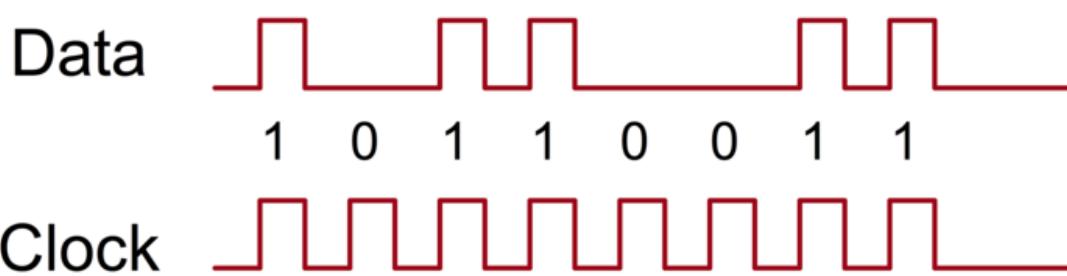
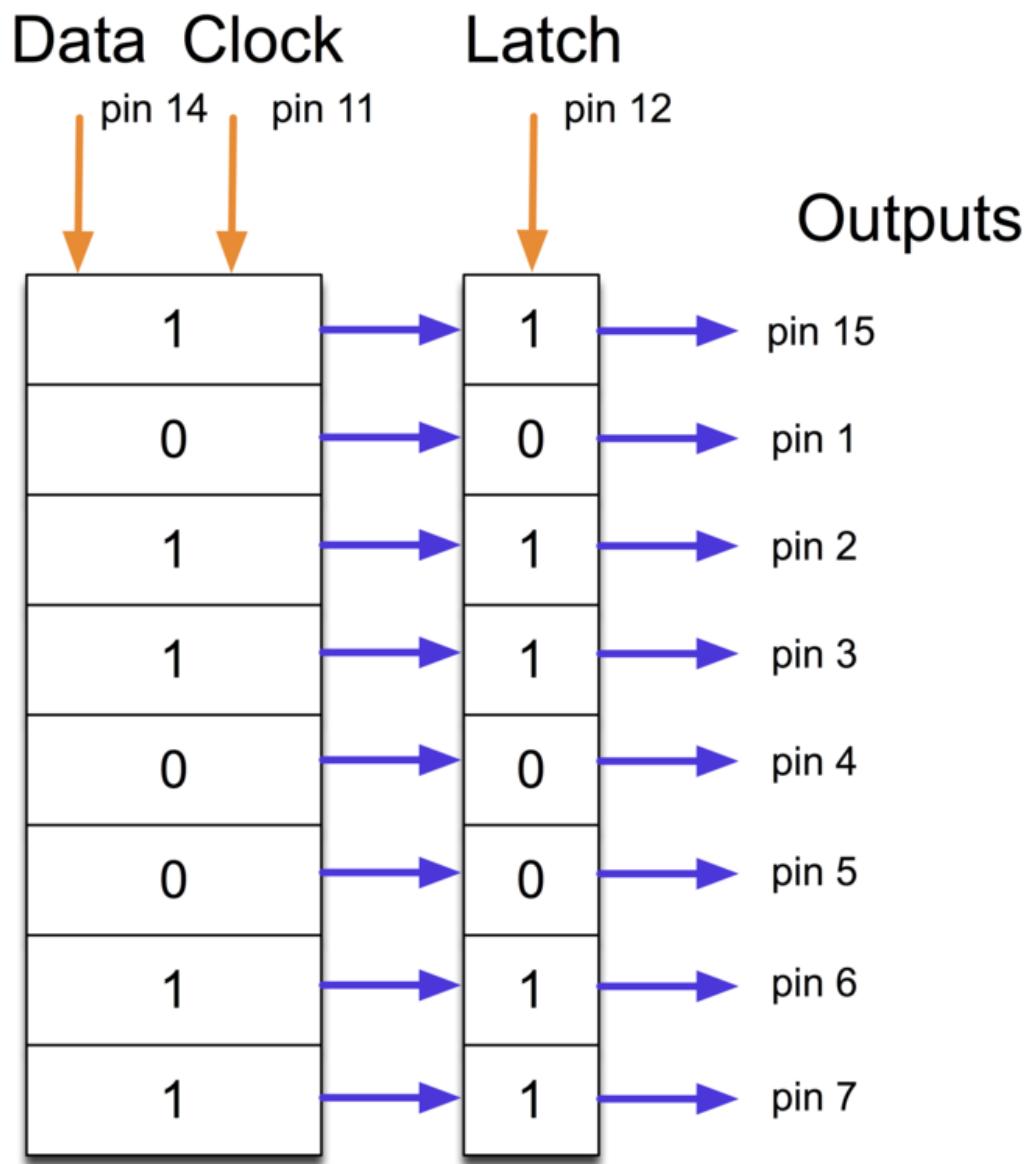
## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) Placa Universal
- (8) LEDs
- (8) 220 ohm resistores
- (1) 74HC595 IC
- (14) M-M alambres

## Introducción de componente

### 74HC595 Registro de desplazamiento:

El registro de desplazamiento es un tipo de chip que tiene lo que puede ser pensado como ocho localidades de memoria, cada uno de los cuales puede ser un 1 o un 0. Para establecer cada uno de estos valores dentro o fuera, que se alimentan de los datos a través del perno de chip de 'datos 'y' Reloj '.



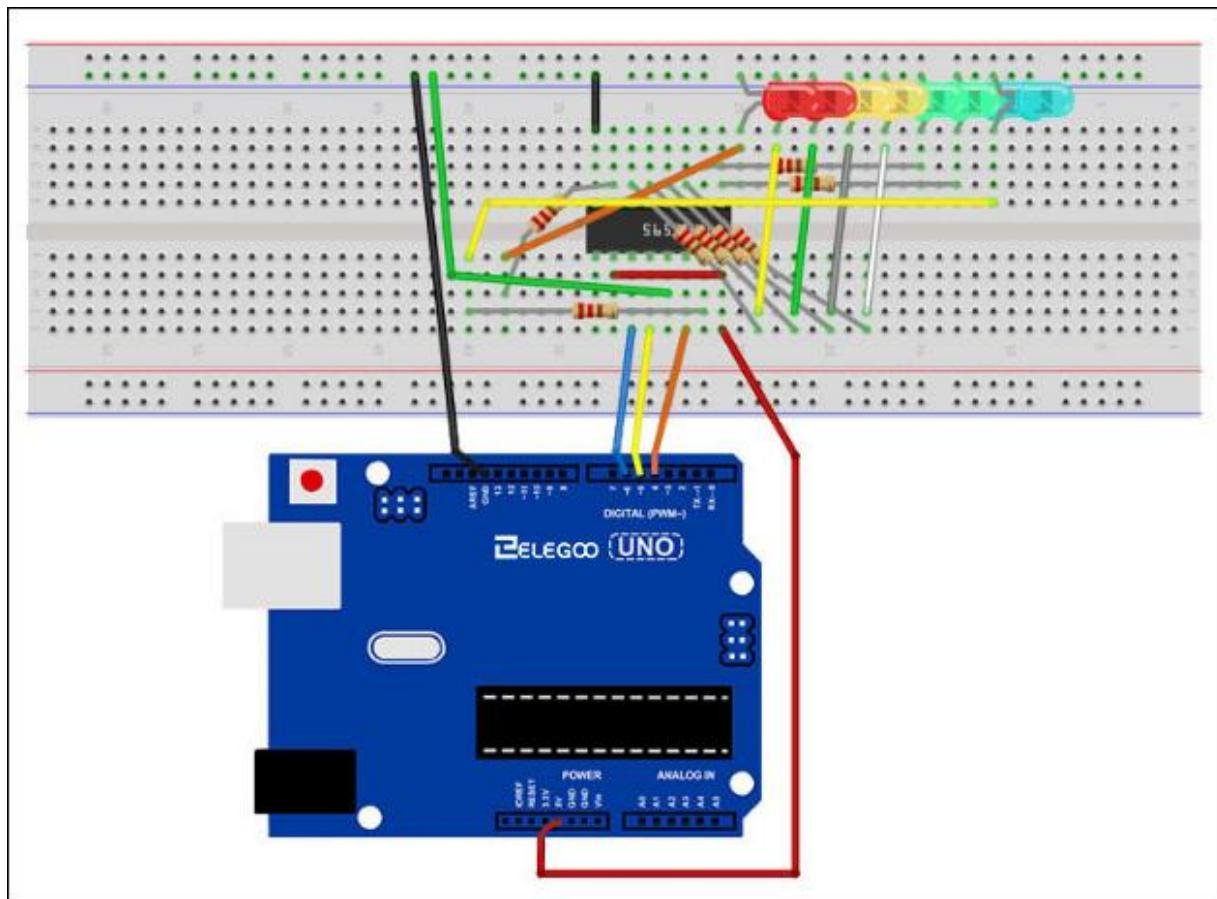
El perno de reloj necesita recibir ocho pulsos. A cada impulso, si el pin de datos es alta, entonces un 1 es empujado en el registro de desplazamiento; de lo contrario, un 0. Cuando se hayan recibido todas las ocho pulsos, permitiendo a los pernos de copias de ' pasador 'esos ocho valores en el registro de cierre. Esto es necesario; de lo contrario, los LED equivocadas sería parpadeo ya que los datos se están cargando en el registro de

desplazamiento.

El chip también tiene un perno de salida de habilitación (OE), que se utiliza para activar o desactivar las salidas a la vez. Puede asociar a un perno de PWM-UNO capaz y utilizar 'escritura analógica' para controlar el brillo de los LEDs. Este pin está activo bajo, por lo que lo ate a GND.

## Conexión

### Diagrama de cableado



Como tenemos ocho LEDs y ocho los resistores para conectar, en realidad hay un buen número de conexiones que se harán.

Es probablemente el más fácil de poner el chip 74HC595 en primer lugar, como casi todo lo demás se conecta a él. Ponerlo de manera que la pequeña muesca en forma de U es hacia la parte superior de placa universal.

El perno 1 del chip es a la izquierda de esta muesca.

- Digital 4 de UNO va al perno # 14 del registro de desplazamiento
- Digital 5 de UNO va al perno # 12 del registro de desplazamiento

- Digital 6 de UNO va al perno # 11 del registro de desplazamiento

Todas menos una de las salidas del IC está en el lado izquierdo del chip. Por lo tanto, para facilitar la conexión, que es donde los LEDs son, también.

Después de que el chip, poner las los resistores en su lugar. Es necesario tener cuidado de que ninguno de los cables de los resistores se tocan entre sí. Debe comprobar esto de nuevo antes de conectar la corriente aUNO. Si le resulta difícil organizar las los resistores sin toca los cables, entonces ayuda a acortar los cables de forma que yacen más cerca de la superficie de placa universal.

A continuación, colocar los LED en la placa universal. todos los cables positivos LED ya deben estar hacia el chip, cualquier lado de la placa universal que se encuentran.

Conecte el puente conduce como se muestra arriba. No se olvide de la que va desde el perno 8 del ICa la columna de GND de la placa universal.

Capacidad de carga hasta el boceto que aparece un poco más tarde y probarlo. Cada LED debe encenderse a su vez hasta que todos los LED están encendidos, y luego todos se apaga y el ciclo se repite.

## Código

Lo primero que hacemos es definir los tres pernos que vamos a utilizar. Estas son las salidas digitales de UNO que se conectarán al perno, reloj y datos estacas del 74HC595.

```
int latchPin = 5;  
int clockPin = 6;  
int dataPin = 4;
```

A continuación, una variable llamada 'leds' se define. Esto se utiliza para mantener el patrón de los LED se encienden actualmente activada o desactivada. Los datos del tipo 'bytes' representa números con ocho bits. Cada bit puede ser encendido o apagado, por lo que este es perfecto para hacer el seguimiento de cuál de nuestros ocho LEDs están encendidos o apagados.

```
byte leds = 0;
```

La función "configuración" sólo establece los tres pernos que estamos utilizando para ser salidas digitales.

```
void setup()
```

```
{
```

```
pinMode(latchPin, OUTPUT);
pinMode(dataPin, OUTPUT);
pinMode(clockPin, OUTPUT);
}
```

La función de 'bucle' se convierte inicialmente todos los LED apagados, dando a la variable 'leds' el valor 0. A continuación, las llamadas de ' actualizar el registro de desplazamiento 'que enviará el patrón de' leds 'en el registro de desplazamiento de modo que todos los LED se apaga . Nos ocuparemos de cómo ' actualizar el registro de desplazamiento 'trabaja más adelante.

La función de bucle hace una pausa de medio segundo y luego comienza a contar de 0 a 7 utilizando el "para" bucle y la variable 'i'. Cada vez, se utiliza la función de 'Configuración de Bit' de Arduino para establecer el bit que controla que el LED en los 'leds' variables. A continuación, llama también ' actualizar el registro de desplazamiento 'por lo que la actualización de leds para reflejar lo que está en los' leds 'variables.

Hay, pues, una demora de medio segundo antes de 'i' se incrementa y el siguiente LED está encendido

```
void loop()
{
leds = 0;
updateShiftRegister();
delay(500);
for (int i = 0; i < 8; i++)
{
bitSet(leds, i);
updateShiftRegister();
delay(500);
}
}
```

La función de ' actualizar el registro de desplazamiento', en primer lugar, coloca el perno a BAJA, a continuación, llama a la función de UNO' desplazar hacia fuera" antes de poner el

"pasador" en alto de nuevo. Esta toma cuatro parámetros. Los dos primeros son los pernos de usar para los datos y reloj, respectivamente.

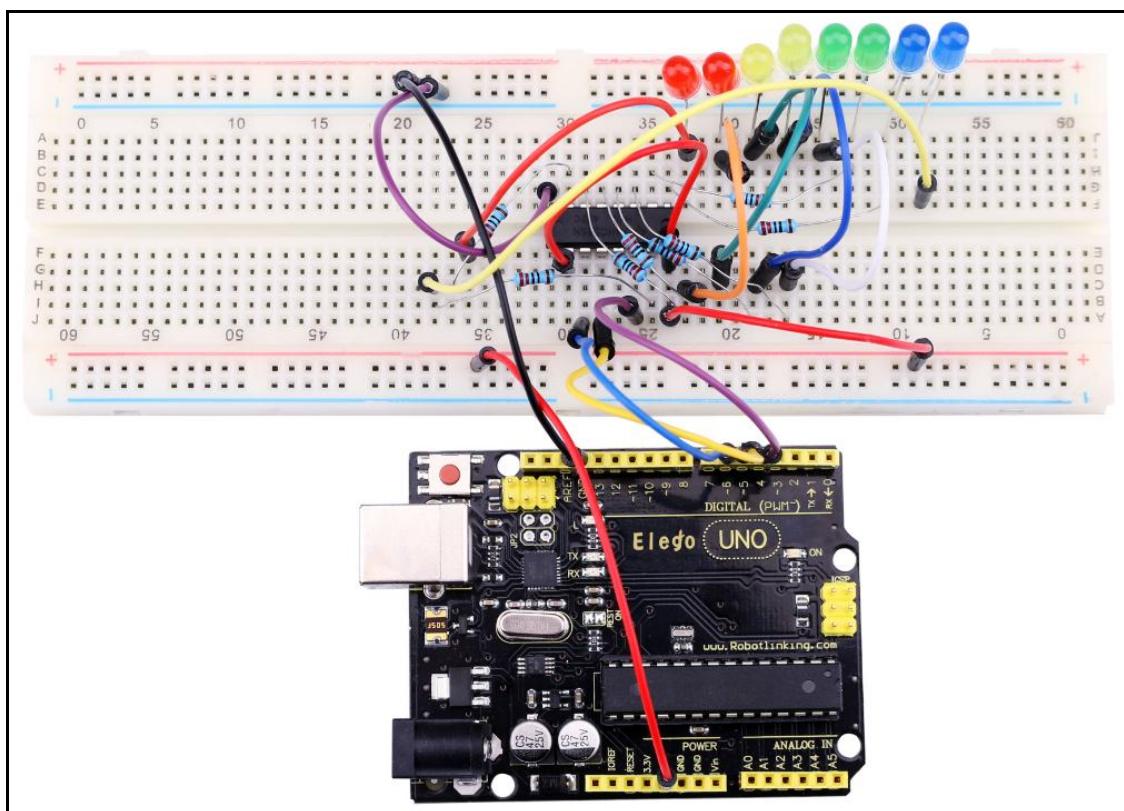
El tercer parámetro especifica en qué extremo de los datos que desea iniciar. Vamos a empezar con el bit más a la derecha, que se conoce como el "El bit menos significativo" (LSB).

El último parámetro es los datos reales para ser desplazado en el registro de desplazamiento que, en este caso, es 'leds'.'

```
void updateShiftRegister()
```

```
{
  digitalWrite(latchPin, LOW);
  shiftOut(dataPin, clockPin, LSBFIRST, leds);
  digitalWrite(latchPin, HIGH);
}
```

Si quería convertir uno de los LED apagados y no en, debería llamar a una función de Arduino similar ( Borrar de bit) con la variable "leds. Esto establece que los bits de 'leds' que es 0 y A continuación, sólo tiene que seguir con una llamada a ' actualizar el registro de desplazamiento "para actualizar los LED reales.



# Lección 7: El Monitor Serial

## Visión general

En esta lección, que se basará en la Lección 6, añadiendo la posibilidad de controlar los LED desde el ordenador utilizando el Monitor Serial de Arduino. El monitor serial es la 'traba' entre la computadora y el UNO. Se le permite enviar y recibir mensajes de texto, útil para la depuración y también el control de la UNO desde un teclado! Por ejemplo, usted será capaz de enviar comandos desde el ordenador para activar los LED.

En esta lección, que va a utilizar exactamente las mismas piezas y un diseño de placa similar a la lección 6. Por lo tanto, si todavía no lo ha hecho, siga lección 6 ahora.

## Pasos

Después de haber subido este boceto en su UNO, haga clic en el botón más a la derecha en la barra de herramientas en el IDE de Arduino. El botón está rodeada por debajo.



La siguiente ventana se abrirá.

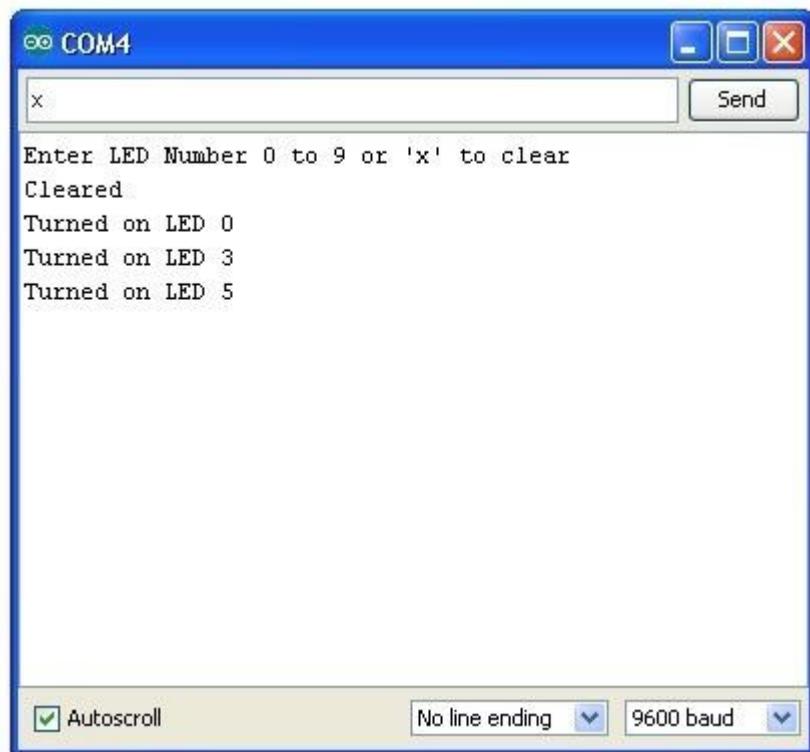


Esta ventana se llama el Monitor Serial y es parte del software IDE de Arduino. Su función es permitir que tanto para enviar mensajes desde el ordenador a una placa de UNO (a través de USB) y también para recibir mensajes de UNO.

El mensaje "Introducir número de LED de 0 a 9 o 'x' para borrar" ha sido enviado por el Arduino. Nos está diciendo lo que manda podemos enviar a la Arduino: o bien enviar la 'x' (para activar todos los LEDs apagados) o el número de los LED que desea activar (donde 0 es la parte inferior del LED, 1 es la siguiente uno para arriba, todo el camino a 7 para el LED superior).

Pruebe a escribir los siguientes comandos en la zona superior del monitor Serial que está a nivel con el botón 'Enviar'. Pulse 'Enviar', después de escribir cada uno de estos caracteres: x 0 3 5

Escribiendo x no tendrá ningún efecto si los LED ya están fuera, pero a medida que se introduce cada número, el LED correspondiente debe encenderse y obtendrá un mensaje de confirmación de la placa de UNO. aparecerá el Monitor Serial como se muestra a continuación.



Tipo X de nuevo y pulse 'Enviar' para apagar todos los LED.

## Código

Como era de esperar, el boceto se basa en el boceto utilizado en la lección 6. Por lo tanto, nos limitaremos a cubrir los nuevos bits aquí. Usted encontrará que es útil referirse al boceto completo en el IDE de Arduino.

En la función 'configuración', hay tres nuevas líneas al final:

```
void setup()
{
pinMode(latchPin, OUTPUT);
pinMode(dataPin, OUTPUT);
pinMode(clockPin, OUTPUT);
updateShiftRegister();
Serial.begin(9600);
while (! Serial); // Wait until Serial is ready - Leonardo
Serial.println("Enter LED Number 0 to 7 or 'x' to clear");
}
```

En primer lugar, tenemos el comando 'Serial.comenzar (9600)'. Así se inicia la comunicación en serie, de modo que la UNO puede enviar comandos a través de la conexión USB. El valor 9600 es llamado el "velocidad de transmisión" de la conexión. Esta es la rapidez con que los datos se va a enviar. Usted puede cambiar esto a un valor más alto, pero también tendrá que cambiar el Monitor Serial de Arduino en el mismo valor. Discutiremos esto más adelante; por ahora, dejarlo en 9600.

La línea que comienza con "mientras que" asegura que hay algo en el otro extremo de la conexión USB para el Arduino para hablar antes de que comience el envío de mensajes. De lo contrario, el mensaje puede ser enviado, pero no se muestra. Esta línea es en realidad sólo es necesario si está utilizando un Leonardo Arduino porque la placa de UNO de Arduino restablece automáticamente la placa de Arduino cuando se abre el monitor Serial, mientras que esto no sucede con el Leonardo.

La última de las nuevas líneas en 'Configuración' envía el mensaje de que vemos en la parte superior del Monitor Serial.

La función de 'bucle' es donde sucede toda la acción:

```
void loop()
{
if (Serial.available())
{
char ch = Serial.read();
if (ch >= '0' && ch <= '7')
{
int led = ch - '0';
bitSet(leds, led);
updateShiftRegister();
Serial.print("Turned on LED ");
Serial.println(led);
}
if (ch == 'x')
{
```

```
leds = 0;  
updateShiftRegister();  
Serial.println("Cleared");  
}  
}  
}
```

Todo lo que sucede dentro del bucle está contenida dentro de una sentencia 'si'. Así que a menos que la llamada a la función de Arduino incorporado de 'Serial.disponible ()' es 'verdadera' , entonces nada más va a pasar.

Serial.disponible () devuelve "verdadera" si los datos se han enviado a UNO y si hay datos listos para ser procesados. Los mensajes entrantes se llevan a cabo en lo que se llama un búfer y Serial.disponible () devuelve verdadera si ese búfer no está vacío.

Si se ha recibido un mensaje, luego se pasa a la siguiente línea de código:

```
char ch = Serial.read();
```

Esto se lee el siguiente carácter de la memoria intermedia, y lo elimina de la memoria intermedia. También lo asigna a la variable 'ch'. El 'ch' variable es del tipo 'char', que significa "carácter". Como su nombre indica, se tiene un solo carácter.

If you have followed the instructions in the prompt at the top of the Serial Monitor, then this character will either be a single digit number between 0 and 7 or the letter 'x'.

Si ha seguido las instrucciones del mensaje en la parte superior del monitor Serial, entonces este personaje, ya sea será un número de un solo dígito entre 0 y 7 o la letra "x".

La sentencia "si" en la siguiente línea comprueba si se trata de un solo dígito por ver si 'ch' es mayor o igual que el carácter '0' y menor o igual al carácter '7'. Se ve un poco caracteres extraños que comparan de esta manera, pero es perfectamente aceptable.

Cada carácter se representa mediante un número único, llamado el valor de ASCII. Esto significa que cuando se comparan caracteres utilizando <= y> =, que es en realidad los valores ASCII que se están comparando.

Si pasa la prueba, entonces llegamos a la siguiente línea:

```
int led = ch - '0';
```

Ahora estamos llevando a cabo la aritmética en los caracteres! Estamos restando el dígito '0' de lo que sea se ha introducido dígitos. Por lo tanto, si ha escrito '0', entonces '0' - '0' será

igual a 0. Si escribió '7' y luego '7' - '0' será igual al número 7, ya que es en realidad los valores ASCII que se utilizan en la resta.

Dado que sabemos que el número de los LED que queremos activar, sólo tenemos que establecer que los bits en los 'leds' variables y actualizar el registro de desplazamiento.

```
bitSet(leds, led);
```

```
updateShiftRegister();
```

Las siguientes dos líneas de escribir de nuevo un mensaje de confirmación a la Serial Monitor.

```
Serial.print("Turned on LED ");
```

```
Serial.println(led);
```

La primera línea utiliza Serial.impresión en lugar de Serial.println. La diferencia entre los dos es que Serial.impresión no se inicia una nueva línea después de imprimir lo que está en su parámetro. Utilizamos esta en la primera línea, porque estamos imprimiendo el mensaje en dos partes: el bit general 'Encender LED 'y luego el número del LED.

El número de los LED se lleva a cabo en un 'int' variable en lugar de ser una cadena de texto. Serial. impresión puede tomar una cadena de texto entre comillas dobles, un 'int' o casi cualquier tipo de variable.

Después de la sentencia de "si" que maneja el caso, cuando se ha tratado de un solo dígito, hay una segunda sentencia de "si" que comprueba para ver si 'ch' es la letra 'x'.

```
if (ch == 'X')  
{  
    leds = 0;  
    updateShiftRegister();  
    Serial.println("Cleared");  
}
```

Si es así, entonces se borra todos los LED y envía un mensaje de confirmación.

# Lección 8: Fotocélula

## Visión general

En esta lección, usted aprenderá cómo medir la intensidad de la luz utilizando una entrada analógica. Usted va a construir en el circuito y usar el nivel de luz para controlar el número de LEDs para ser encendido

La fotocélula es en la parte inferior de la placa, donde el bote estaba por encima.

## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) Placa Universal
- (8) LEDs
- (8) 220 ohm resistores
- (1) 1k ohm resistores
- (1) 74HC595 IC
- (1) Fotocélula
- (14) M-M alambres

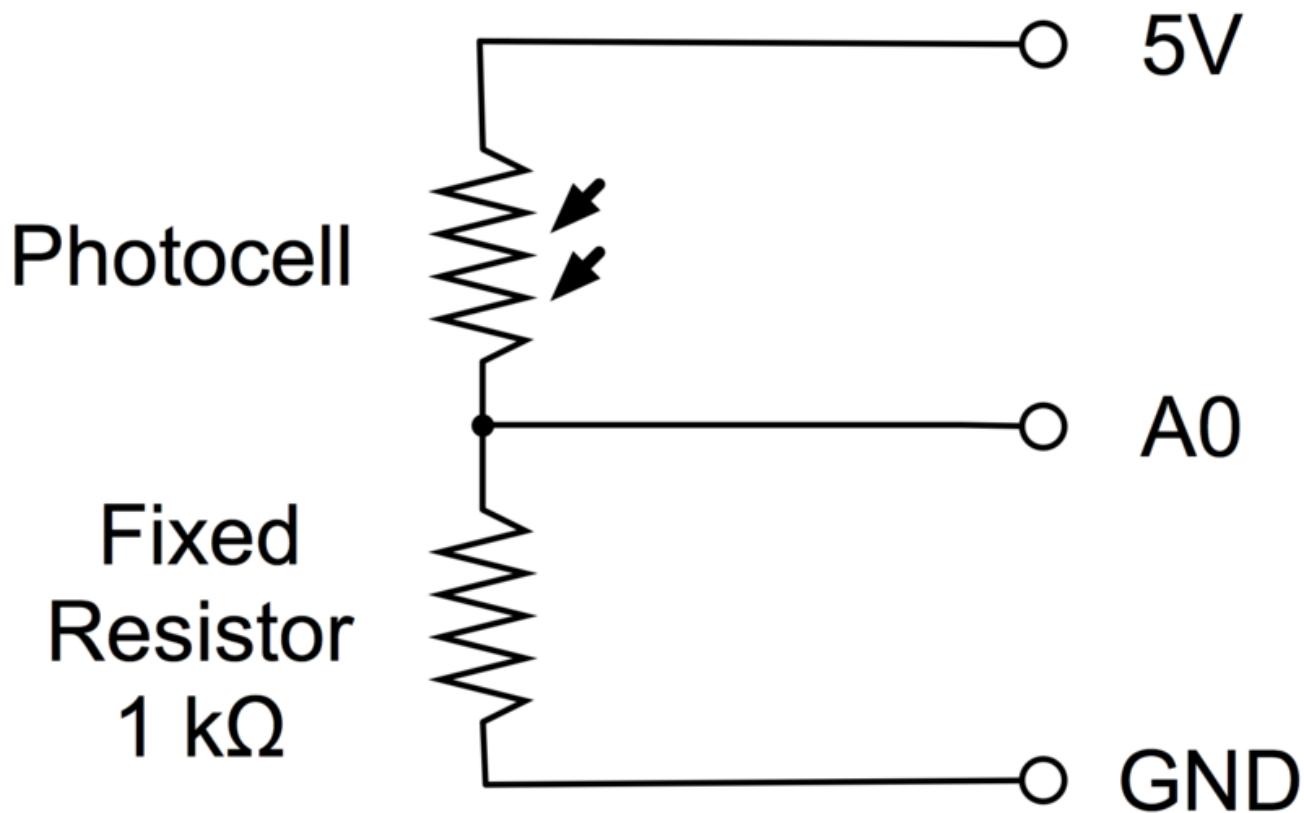
## Introducción de Componente

### Fotocélula:

La fotocélula utilizado es de un tipo llamado un resistor dependiente de la luz (LDR). Como su nombre indica, este componente actúa como el resistor, excepto que el resistor cambia en respuesta a la cantidad de luz que cae sobre ellas.

Éste tiene una resistencia de unos 50 kW en la oscuridad cerca de  $500 \Omega$  y en la luz brillante. Para convertir este valor del resistor variable en algo que podemos medir en la entrada analógica de la placa de UNO R3, que necesita ser convertida en una tensión.

La forma más sencilla de hacerlo es combinarlo con el resistor fijo.



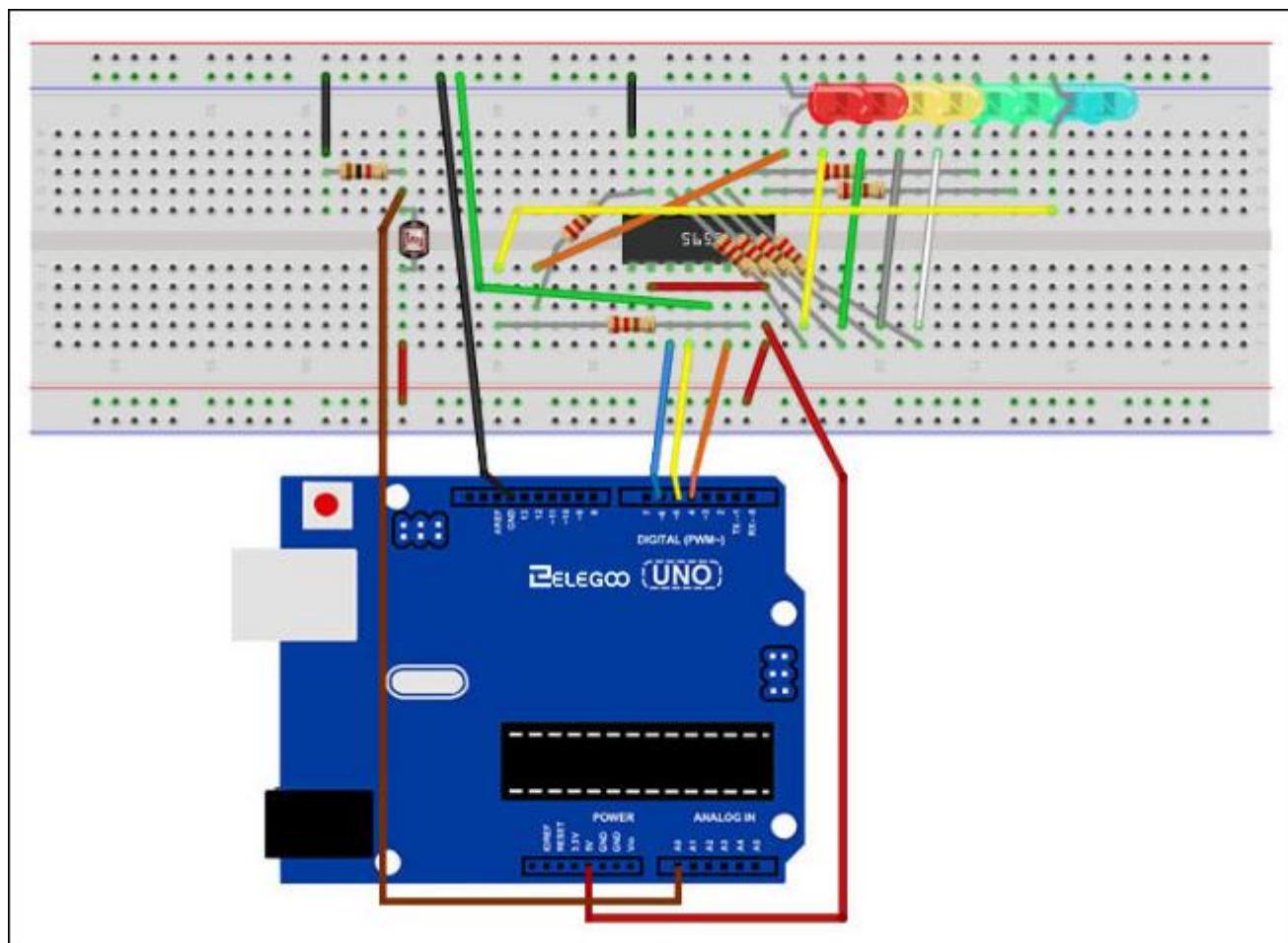
El resistor y la fotocélula en conjunto se comportan como una olla. Cuando la luz es muy brillante, entonces el resistor de la fotocélula es muy baja en comparación con el valor de resistor fijo, y así es como si el bote se convirtió en el máximo.

Cuando la célula fotoeléctrica es la luz apagada, la resistencia se vuelve mayor que la resistencia fija  $1\text{ k}\Omega$  y es como si el bote estaba siendo girada hacia GND.

Capacidad de carga del boceto que figura en la siguiente sección y tratar de cubrir la fotocélula con el dedo, y lo sostiene cerca de una fuente de luz

## Conexión

### Diagrama de cableado



## Código

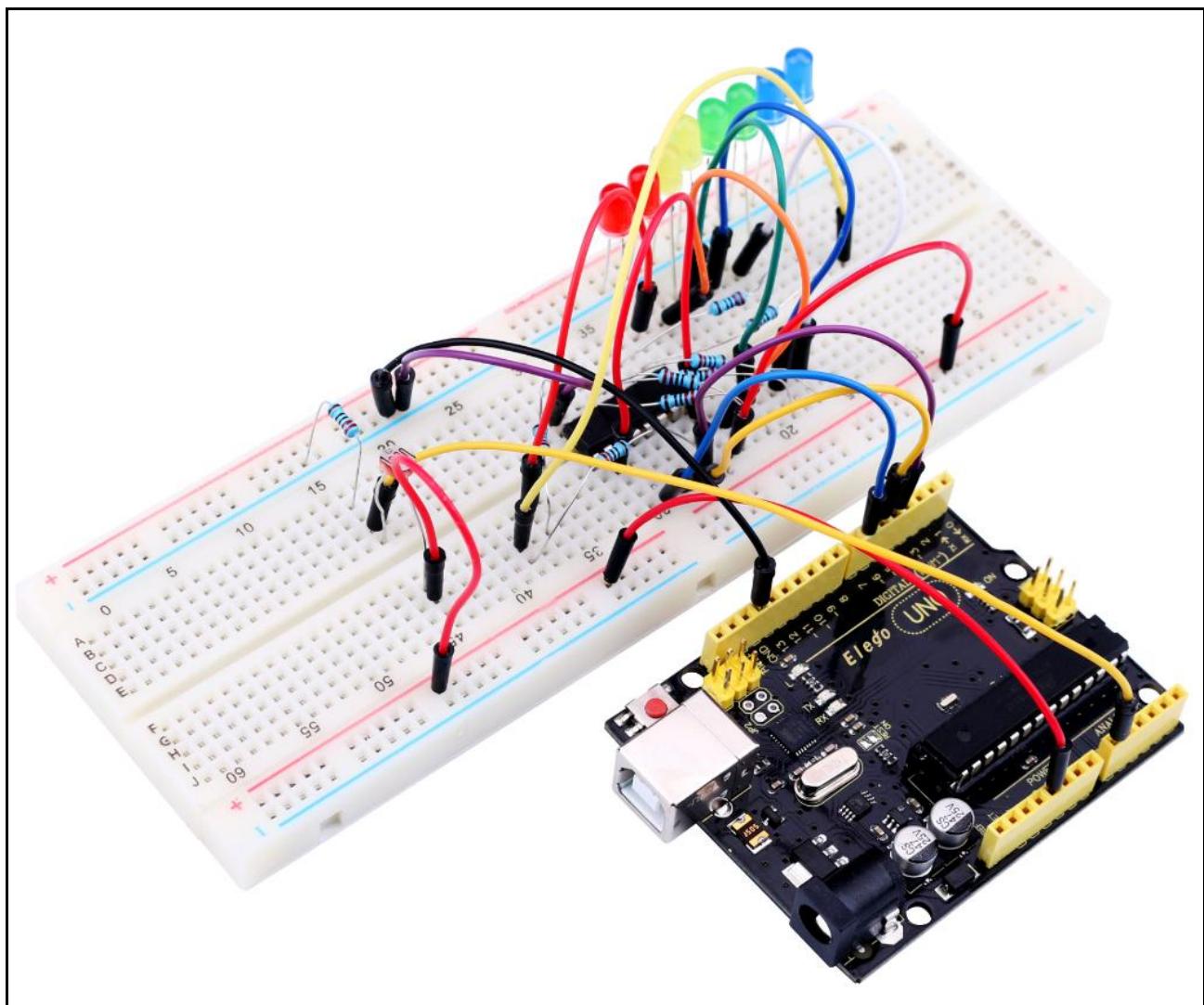
La primera cosa a tener en cuenta es que hemos cambiado el nombre del perno analógico a ser "Perno de la luz 'en lugar de' Perno de olla' puesto que ya no tenemos una olla conectado.

El único otro cambio sustancial en el boceto es la línea que calcula cuántos de los LED para la luz:

```
int numLEDSlit = reading / 57; // all LEDs lit at 1k
```

Esta vez, se divide la lectura directa por 57 en lugar de 114. En otras palabras, lo dividimos por la mitad de lo que lo hicimos con la olla para dividirlo en nueve zonas, de ningún LED de los ocho lit. Este factor adicional es dar cuenta del resistor de 1 kΩ fijo. Esto significa que cuando la célula fotoeléctrica tiene una resistencia de 1 kΩ (la misma que el resistor fijo), la lectura en bruto será  $1023/2 = 511$ . Esto equivale a todos los LEDs se encienden y entonces

un poco (numLEDSLit) habrá 9.



# Lección 9: Haciendo sonidos

## Visión general

En esta lección, usted aprenderá cómo generar un sonido con un zumbador activo.

## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) Zumbador activo
- (2) F-M alambres

## Introducción de Componente

### ZUMBADOR:

Zumbadores electrónicos son CC-alimentado y equipado con un circuito integrado. Son ampliamente utilizados en computadoras, impresoras, fotocopiadoras, alarmas, juguetes electrónicos, dispositivos electrónicos para automóviles, teléfonos, temporizadores y otros productos electrónicos para dispositivos de voz. Zumbadores se pueden categorizar como las activas y pasivas. A su vez los pernos de dos zumbadores boca arriba. El uno con una placa de circuito verde es un zumbador pasivo, mientras que el otro cerrado con una cinta negra es uno activo.

La diferencia entre los dos es que un zumbador activo tiene un oscilante fuente incorporado, por lo que va a generar un sonido cuando electrificado. Un zumbador pasivo no tiene una fuente tal para que no se pío si se utilizan señales de corriente continua; En su lugar, es necesario utilizar ondas cuadradas cuya frecuencia está comprendida entre 2K y 5K para iniciarla. El zumbador activo es a menudo más caro que la pasiva debido a las múltiples incorporado circuitos oscilantes.

## Conexión Esquemático

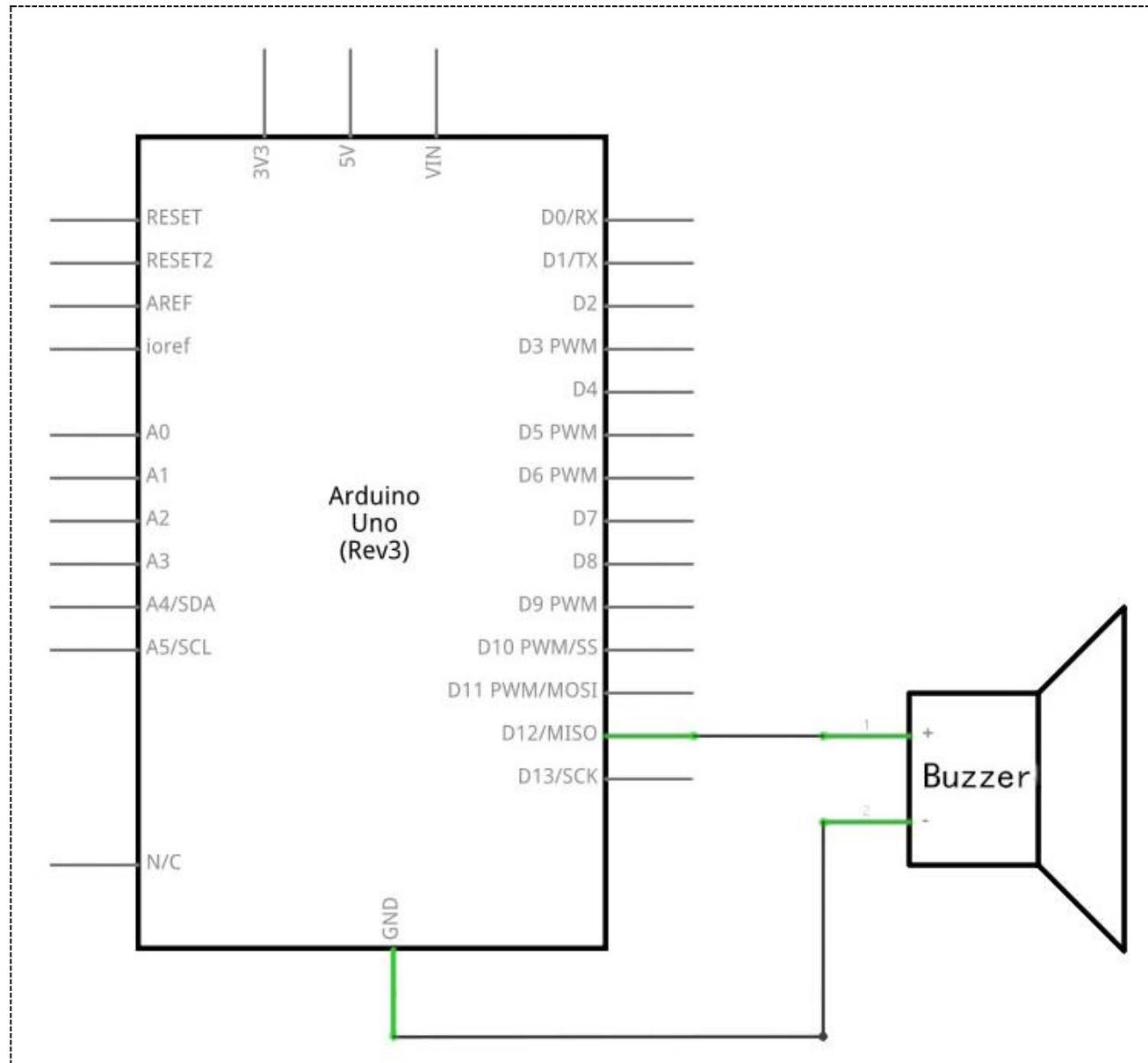
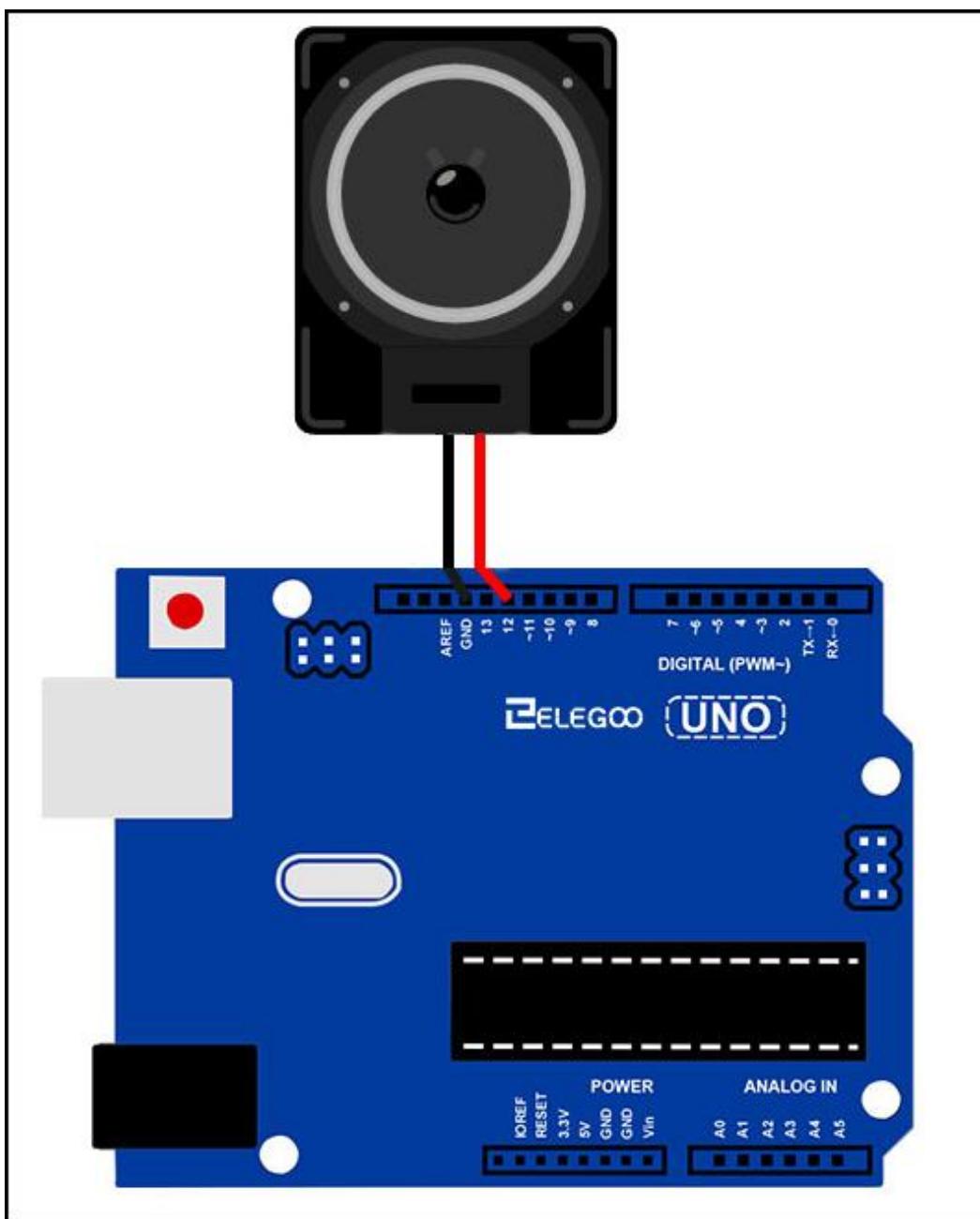
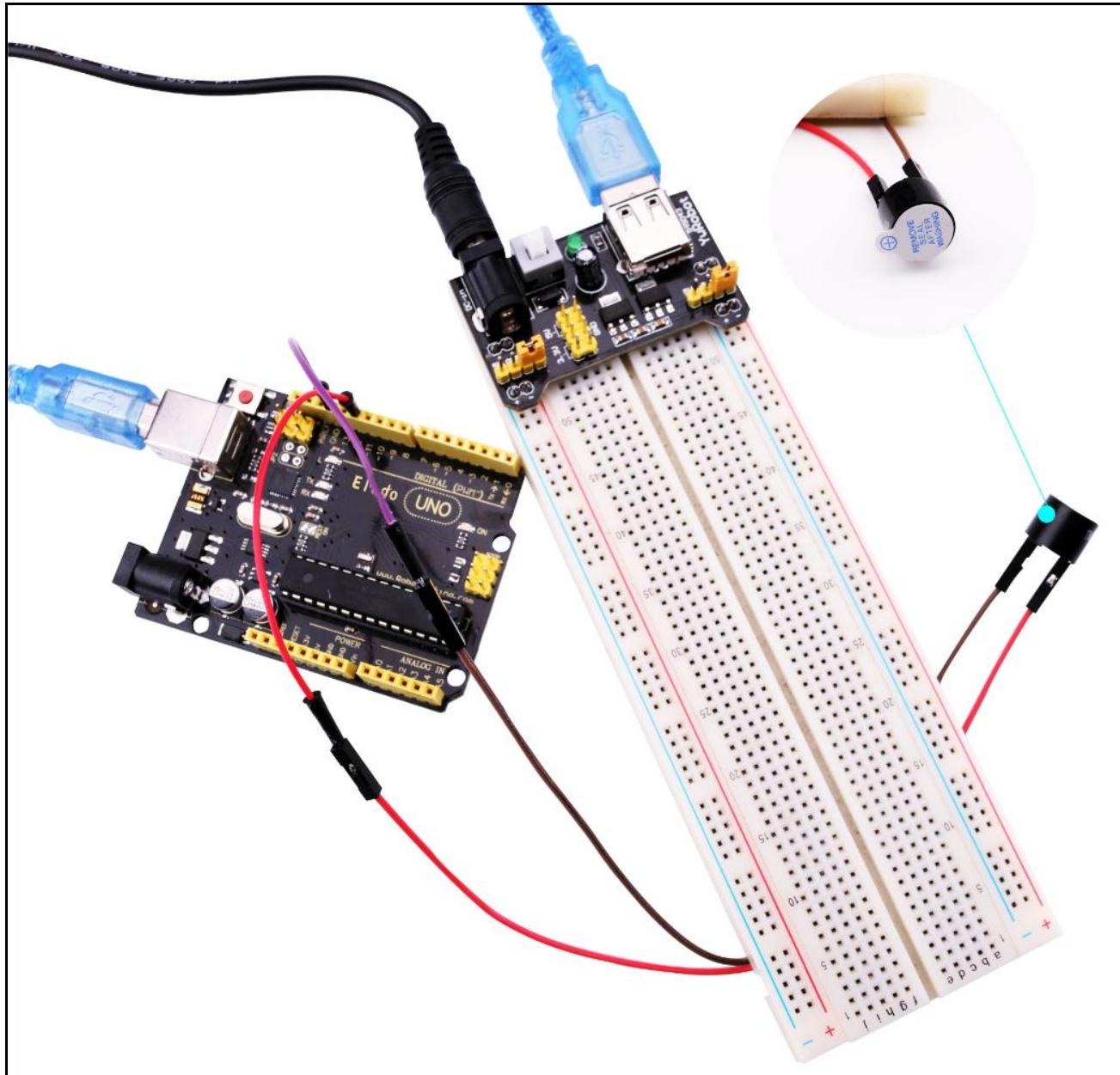


Diagrama de cableado



## Código

Ver el archivo de código.



# Lección 10: Zumbador Pasivo

## Visión general

En esta lección, usted aprenderá cómo utilizar un zumbador pasivo.

El propósito del experimento es el de generar ocho sonidos diferentes, cada sonido que dura 0.5 segundos: desde Alto Do (523Hz), Re (587Hz), Mi (659Hz), Fa (698Hz), So (784Hz), La (880Hz), Si (988Hz) para Do agudos(1047Hz).

## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) Zumbador Pasivo
- (2) F-M alambres

## Introducción de Componente

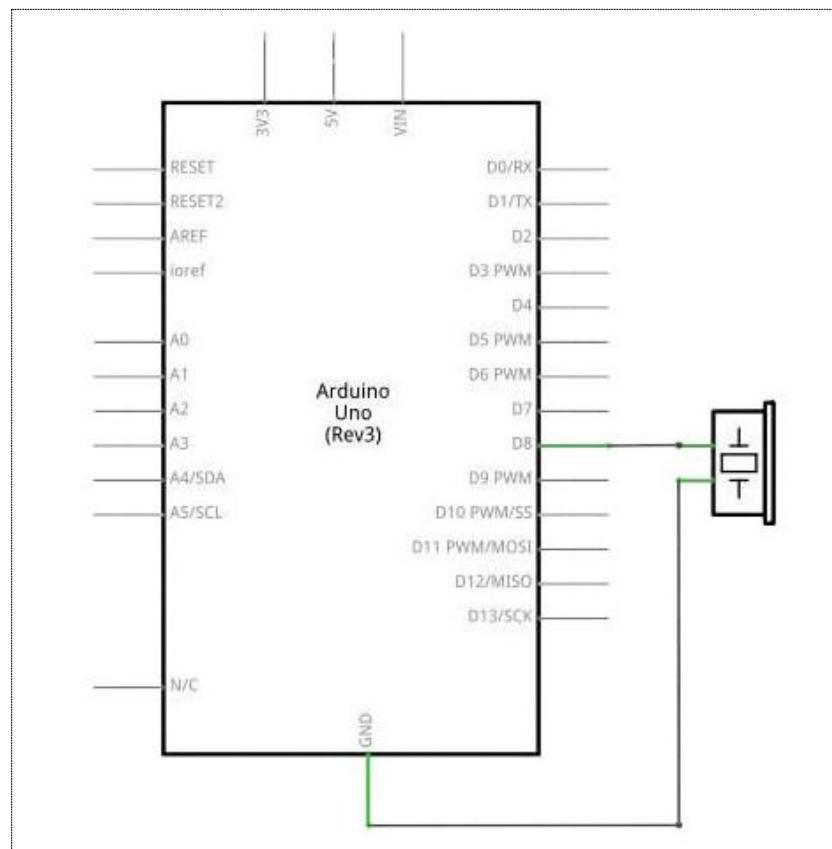
Zumbador Pasivo :

El principio de funcionamiento de pasivo zumbador que utilice audio de generación de PWM para hacer que el aire vibre. Apropiadamente cambiado, siempre que la frecuencia de vibración, puede generar diferentes sonidos. Por ejemplo, el envío de un pulso de 523Hz, que puede generar Alto Do, el pulso de 587Hz, que puede generar gama media Re, el pulso de 659Hz, se puede producir de gama media Mi. Por el zumbador, puede reproducir una canción.

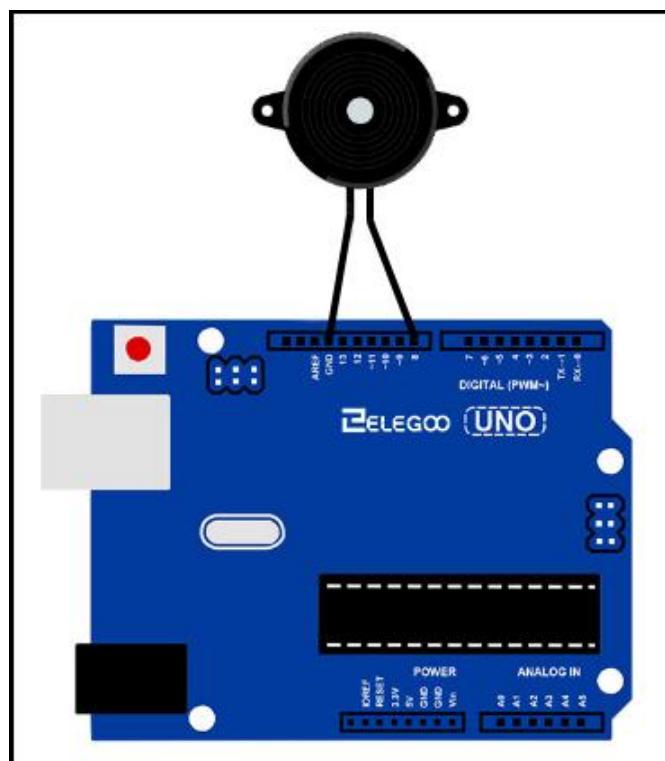
Debemos tener cuidado de no utilizar Escritura analógica de la placa deUNO R3() para generar un pulso con el zumbador, debido a que la salida de impulsos de la escritura analógica () es fijo (500 Hz).

## Conexión

### Esquemático



### Diagrama de cableado



---

Cableado del zumbador conectado a la placa de UNO R3, el rojo (positivo) al perno 8, alambre negro (negativo) al GND

## Código

**Antes de poder ejecutar esto, asegúrese de que ha instalado la biblioteca de <tonos> y volver a instalarla, si es necesario. De lo contrario, su código no funcionará.**

# Lección 11: Interruptor de bola

## Visión general

En esta lección, usted aprenderá cómo utilizar un interruptor de bola con el fin de detectar pequeño ángulo de inclinación.

## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) Interruptor de bola
- (2) F-M alambres

## Introducción de Componente

Sensor de inclinación:

sensores de inclinación permiten detectar la orientación o inclinación. Son pequeños, de bajo costo, baja potencia y fácil de usar. Si se utiliza correctamente, no se gastará. Su simplicidad hace populares para los juguetes, aparatos y electrodomésticos. A veces, se les conoce como "interruptores de mercurio", "interruptores basculantes" o "sensores de bolas rodando" por razones obvias.

Por lo general, se componen de una cavidad de algún tipo (cilíndrica es popular, aunque no siempre) con una masa sin conductor en el interior, como una gota de mercurio o bola rodando. Un extremo de la cavidad tiene dos elementos conductores (polos). Cuando se orienta el sensor para que ese fin es a la baja, los rollos de masa sobre la polos y los corta ellos, actuando como un interruptor de tiro.

Aunque no es tan precisa o flexible como un acelerómetro completo, interruptores basculantes pueden detectar movimiento o la orientación. Otra ventaja es que las grandes pueden cambiar el poder por sí mismos. Los acelerómetros, por otro lado, digital de salida o tensión analógica que deben entonces ser analizados utilizando circuitería adicional.

## Conexión

### Esquemático

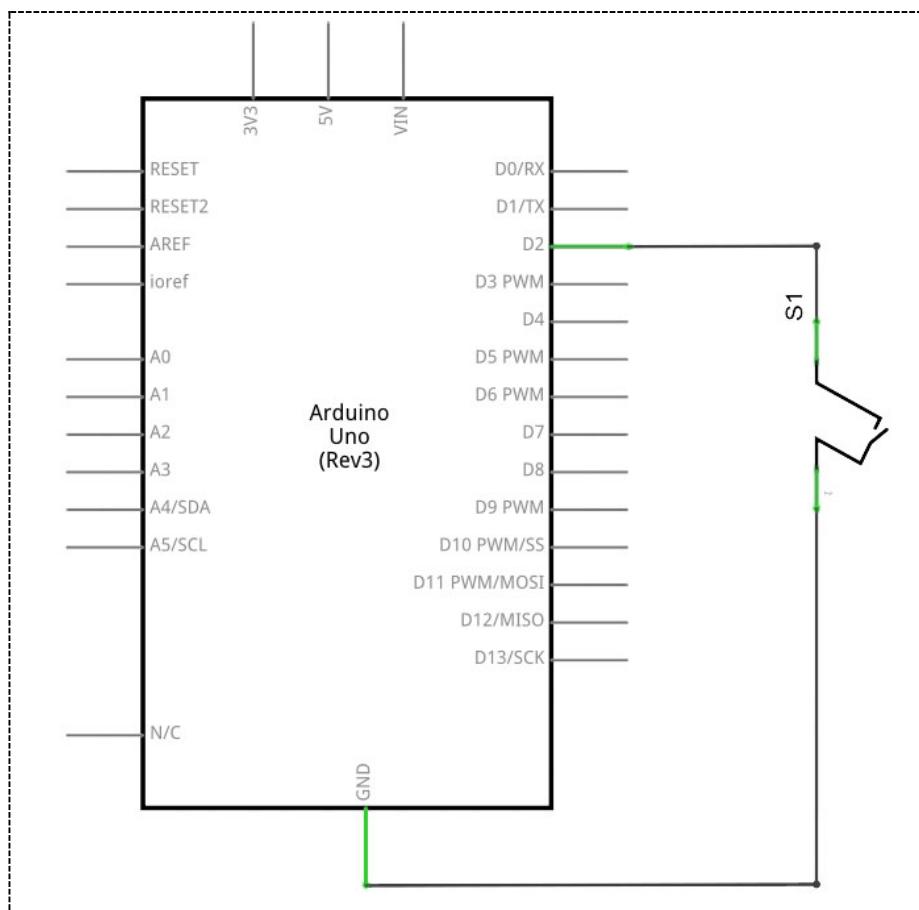
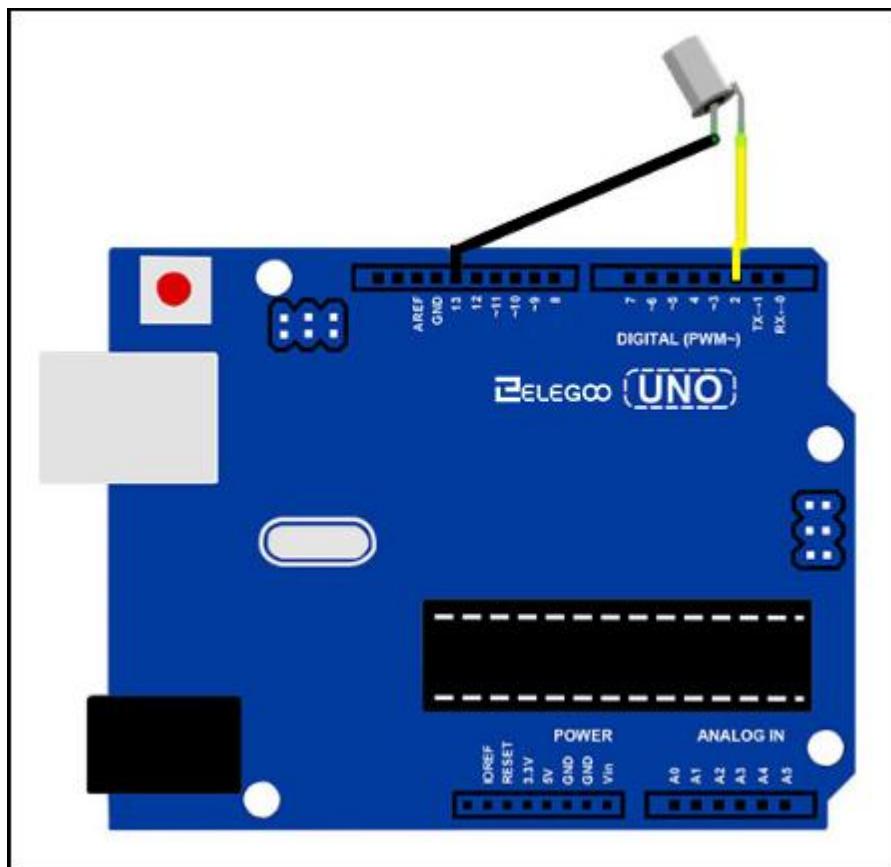
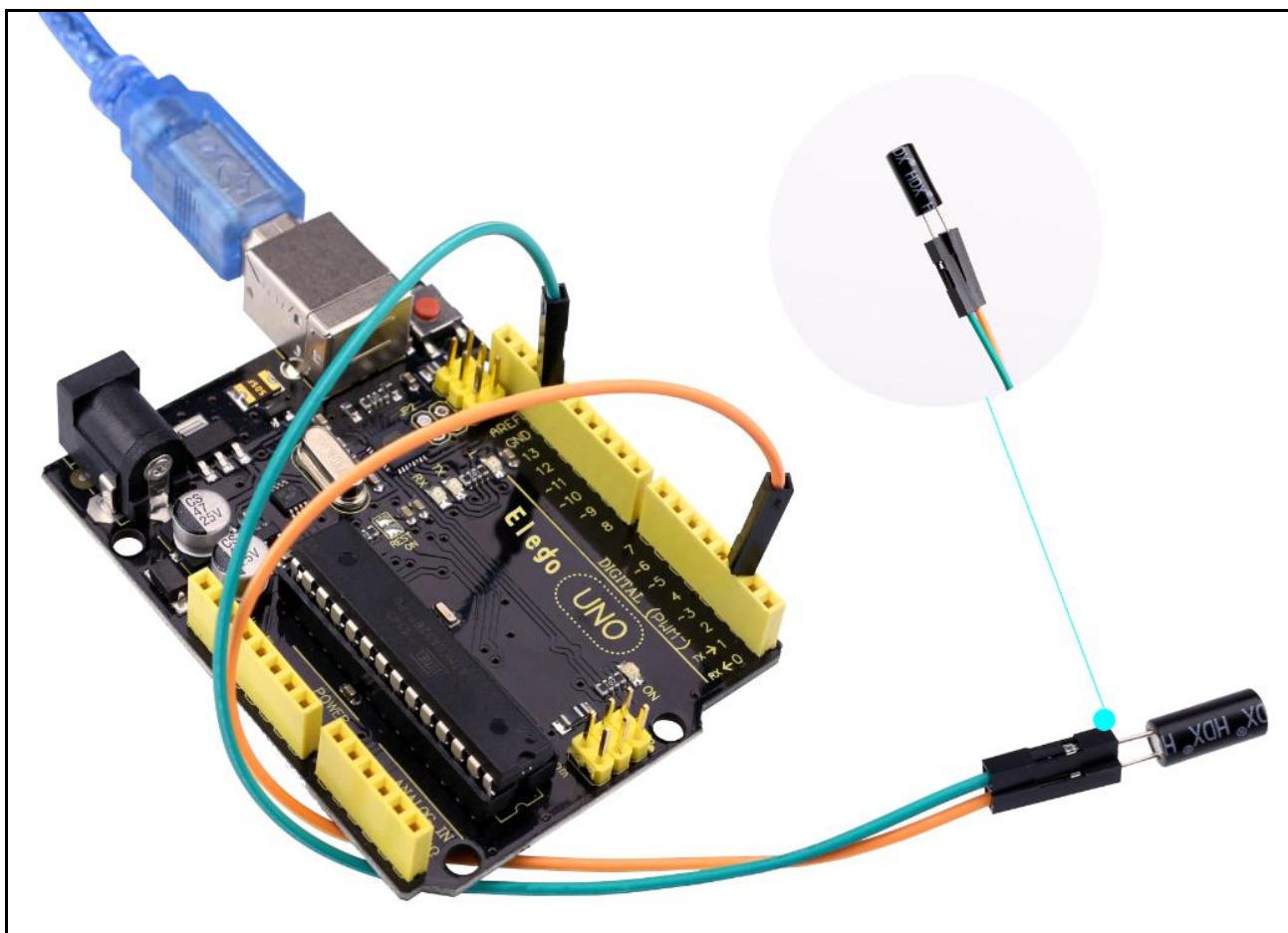


Diagrama de cableado



## Código

[Ver el archivo de código](#)



# Lección 12: Relé

## Visión general

En esta lección, usted aprenderá cómo utilizar un relé.

## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) Placa Universal
- (1) PN2222
- (1) 1N4007
- (1) 220 ohm resistor
- (1) 6v DC motor
- (1) Relé
- (1) Módulo de fuente de alimentación de placa universal
- (1) Adaptador de fuente de alimentación de 9v
- (8) M-M alambres

## Introducción de Componente

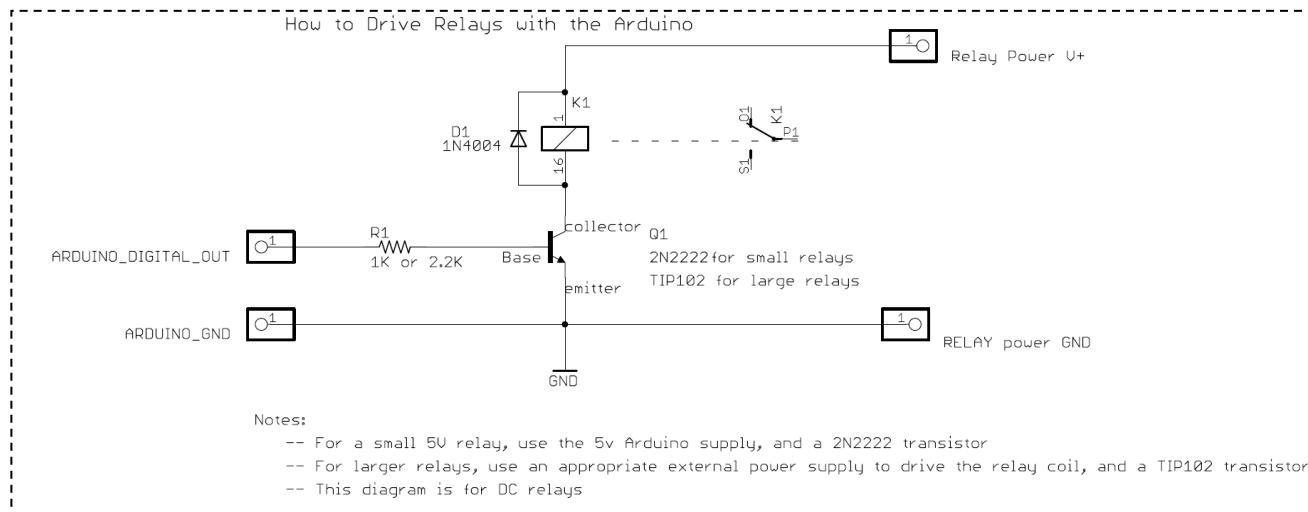
### Relé:

Un relé es un interruptor accionado eléctricamente. Muchos relés utilizan un electroimán para operar mecánicamente un interruptor, pero otros principios de funcionamiento también se utilizan como relés de estado sólido. Los relés se utilizan cuando es necesario para controlar un circuito por una señal de baja potencia (con aislamiento eléctrico completo entre el control y los circuitos controlados), o cuando varios circuitos deben ser controladas por una señal. Los primeros relés se utilizan en los circuitos telegráficos de larga distancia como amplificadores. Repitieron la señal que viene de un circuito y retransmitida en otro circuito. Los relés se utilizan ampliamente en las centrales telefónicas y los primeros ordenadores para realizar operaciones lógicas.

Un tipo de relé que puede manejar la alta potencia necesaria para controlar directamente un motor eléctrico o de otras cargas se llama un contactor. Relés de estado sólido de control de circuitos de potencia sin partes móviles, en lugar de utilizar un dispositivo semiconductor para realizar la conmutación. Relés con características de funcionamiento calibrados y en

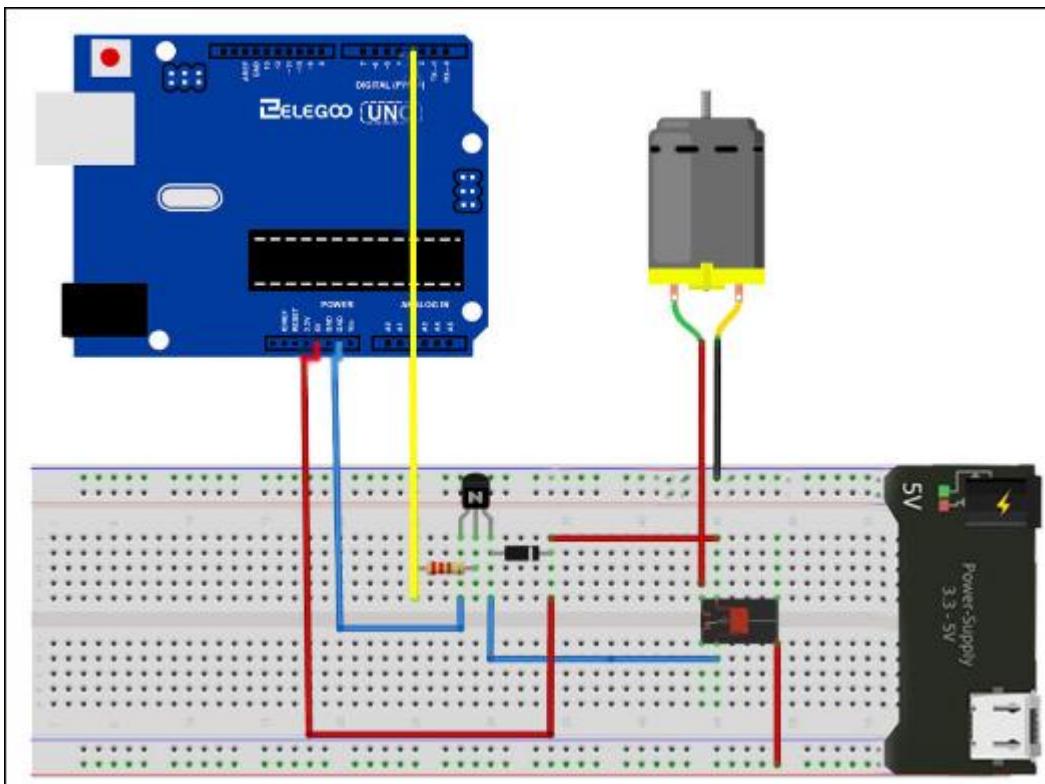
ocasiones varias bobinas de operación se utilizan para proteger los circuitos eléctricos de sobrecarga o fallas. En los sistemas eléctricos modernos, estas funciones son realizadas por los instrumentos digitales llamados "relés de protección".

A continuación se muestra el boceto de cómo conducir relé con Arduino (descarga desde el arduino.cc)



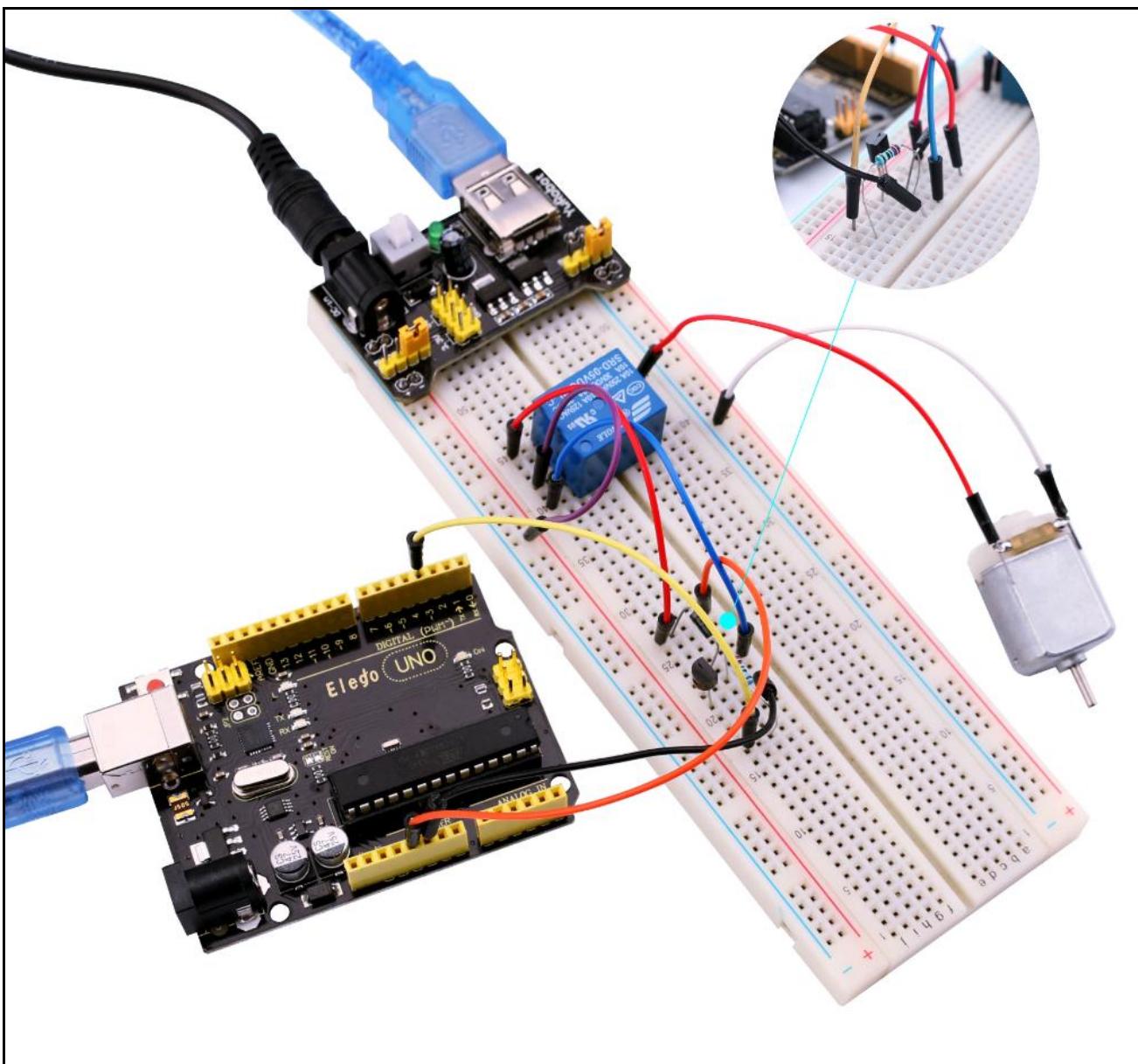
## Conexión

### Diagrama de cableado



## Código

Ver el archivo de código.



# Lección 13: 74HC595 y Visualizador de Segmentos

## Visión general

Después de aprender lección 6 y lección 7, utilizaremos el registro de desplazamiento 74HC595 para controlar el visualizador de segmentos. La pantalla mostrará el segmento número entre 9-0.

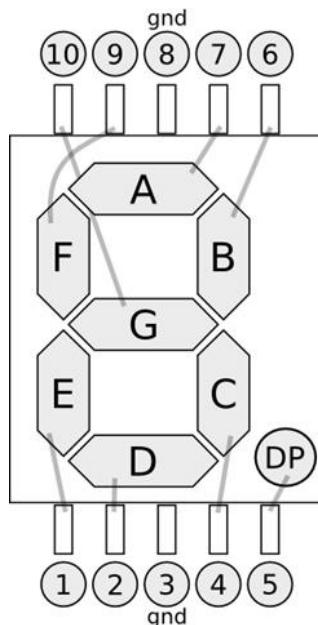
## Componentes Requeridos

- (1) Elegoo UNO R3
- (2) Placa Universal
- (1) 74HC595
- (1) Visualizador de Segmentos
- (8) 220 ohm resistor
- (20) M-M alambres

## Introducción de Componente

### Visualizador de siete segmentos

A continuación se muestra el diagrama de perno de de siete segmentos.

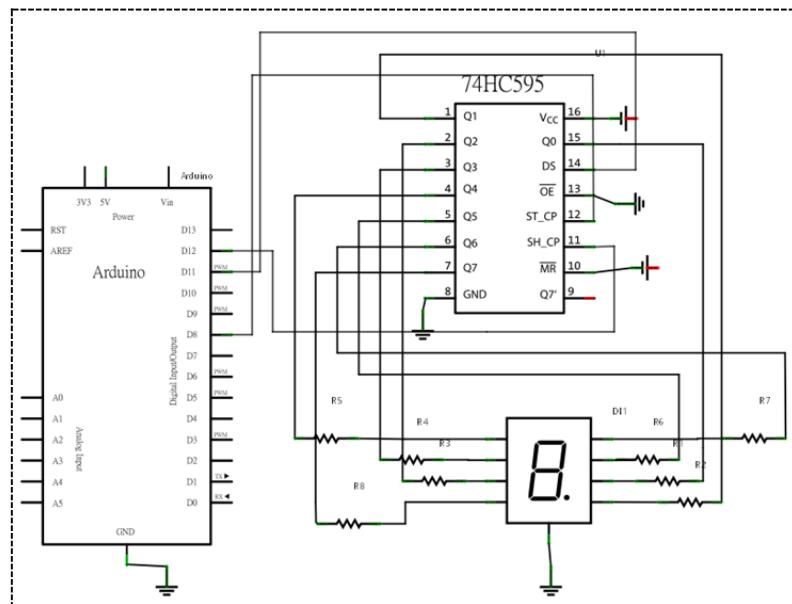


0-9 de diez dígitos se corresponden con cada segmento son los siguientes (la siguiente tabla se aplica cátodo común dispositivo de visualización de siete segmentos, si está utilizando un ánodo común, la mesa debe ser reemplazado cada 1 0 0 todos deben sustituido por 1):

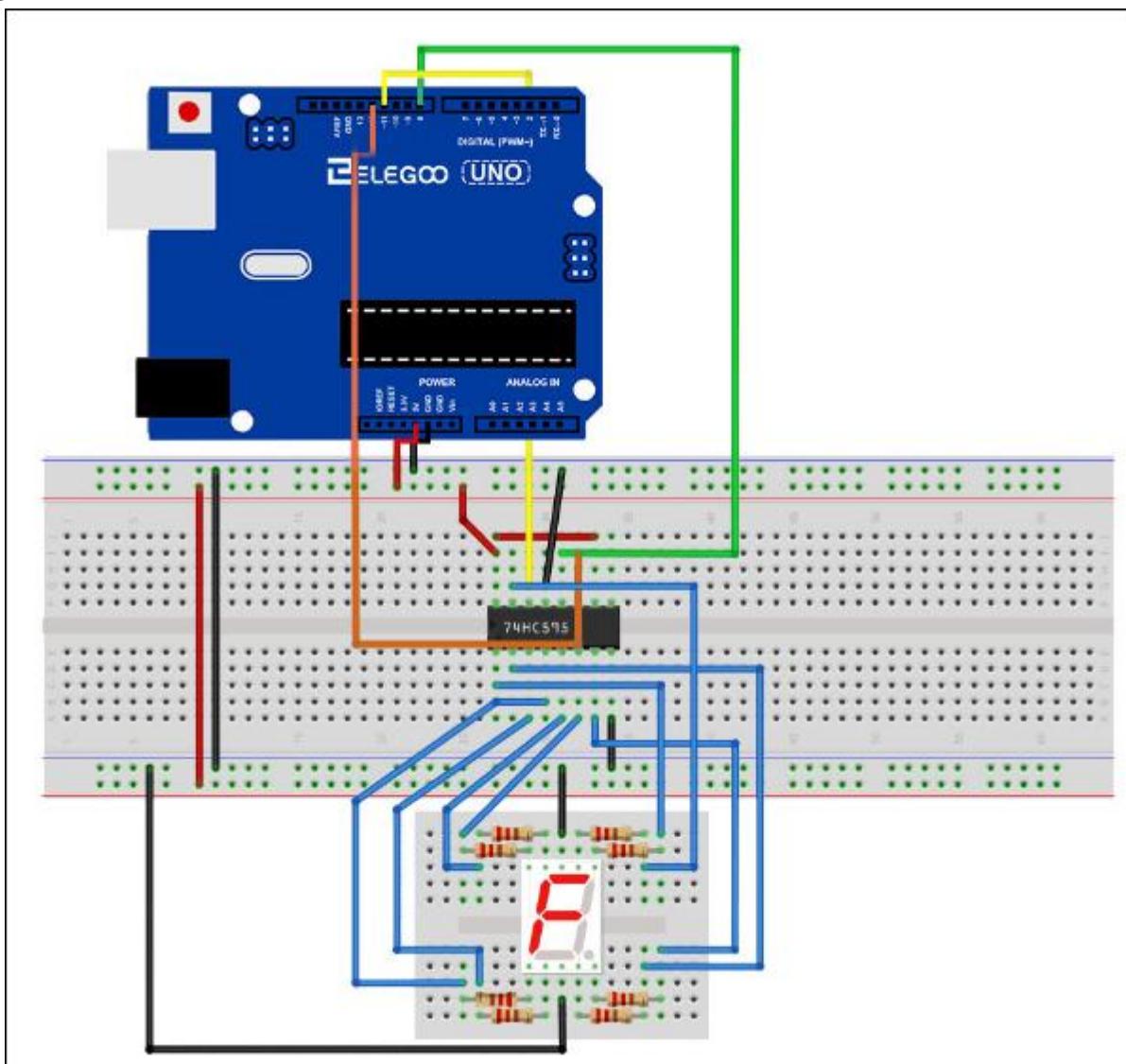
Visualizador digital	dp	a	b	c	d	e	f	g
0	0	1	1	1	1	1	1	0
1	0	0	1	1	0	0	0	0
2	0	1	1	0	1	1	0	1
3	0	1	1	1	1	0	0	1
4	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1
6	0	1	0	1	1	1	1	1
7	0	1	1	1	0	0	0	0
8	0	1	1	1	1	1	1	1
9	0	1	1	1	1	0	1	1

## Conexión

### Esquemático



## Diagrama de cableado



La siguiente tabla muestra la tabla de correspondencia de perno 74HC595 del visualizador de siete segmentos:

74HC595 perno	Siete muestra el perno de control notable(trazo)
Q0	7 (A)
Q1	6 (B)
Q2	4 (C)
Q3	2 (D)
Q4	1 (E)
Q5	9 (F)
Q6	10 (G)

Q7

5 (DP)

Primer paso: Conectar 74HC595

En primer lugar, conecte el cableado a la potencia y el suelo:

- Vcc (perno 16) y MR (perno 10) conectado a 5V
- GND (perno 8) y OE (perno 13) al suelo

Conecta DS, ST\_CP y SH\_CP pernos:

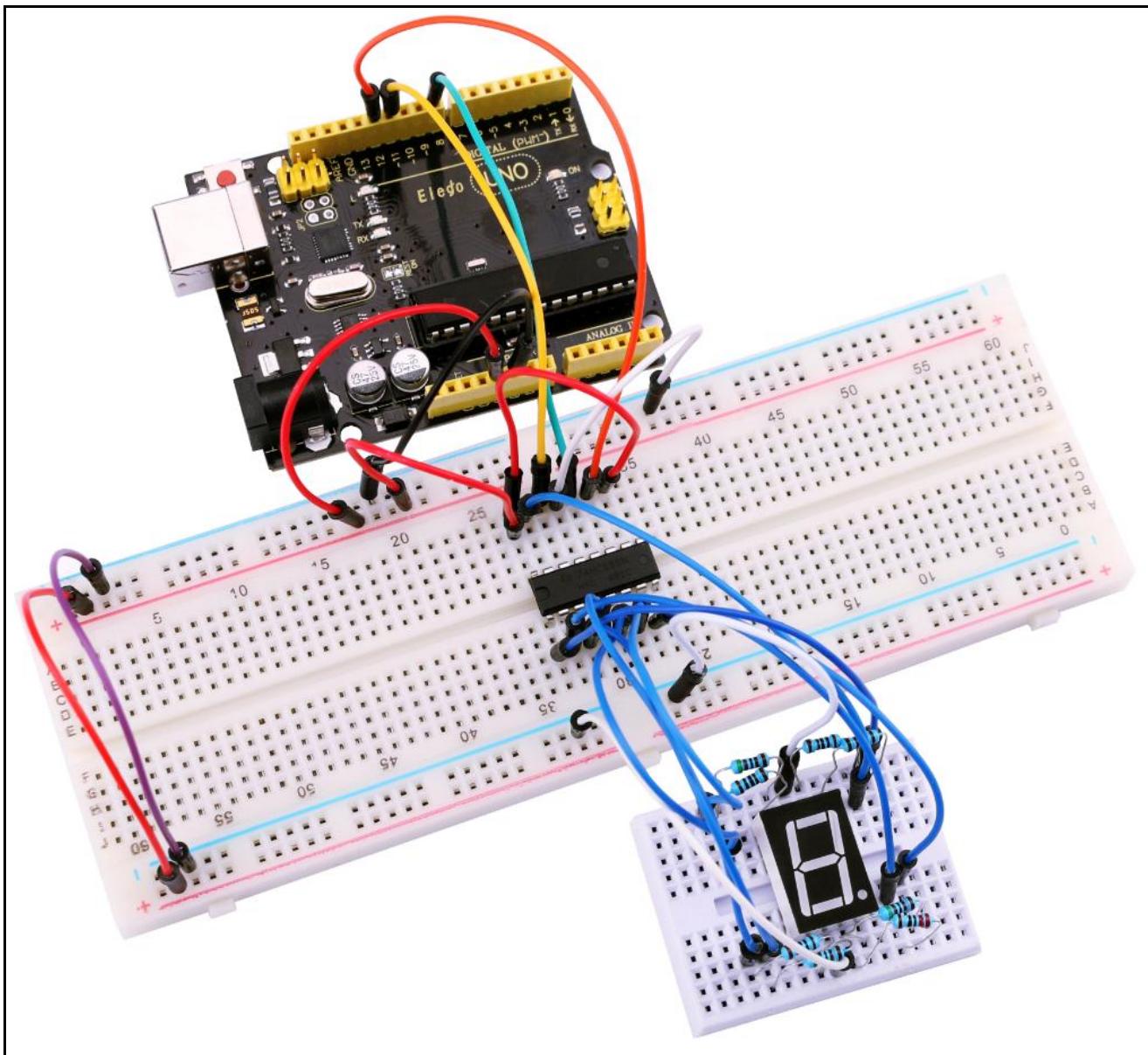
- DS (perno 14) conectado a perno 11 de placa de UNO R3 (línea azul)
- ST\_CP (perno 12, pasador) conectado a perno 8 de placa de UNO R3 (línea verde)
- SH\_CP (perno 11, perno de reloj) conectado a perno 12 de placa de UNO R3 (amarillo)

Segundo paso: Conectar el visualizador de siete segmentos

- La visualizador de siete segmentos pernos 3, 8 a la placa deUNO R3 de GND(En este ejemplo se utiliza el cátodo común, si se utiliza el ánodo común, por favor conecte el perno 3, 8 perno a placa de UNO R3 + 5V)
- De acuerdo con la tabla anterior, conectar el 74HC595 Q0 ~ Q7 a visualizador de siete segmentos correspondiente perno de (A ~ G y DP), y luego cada pie en el resistor de 220 ohmios en serie.

## Código

[Ver el archivo de código](#)



# Lección 14: Visualizador de 7 segmentos de 4 dígitos

## Visión general

En esta lección, usted aprenderá cómo utilizar un visualizador de 7 segmentos de 4 dígitos. Cuando se utiliza visualizador de 7 segmentos de 4 dígitos y está ánodo común, el perno de ánodo común se conecta a la fuente de energía; si se trata de cátodo común, el perno de cátodo común se conecta a la GND. Cuando se utiliza visualizador de 7 segmentos de 4 dígitos, el ánodo común o cátodo común perno de se utiliza para controlar los dígitos se muestra. A pesar de que sólo hay un trabajo dígitos, el principio de la persistencia de la visión le permite ver todos los números que aparecen debido a que cada velocidad del escaneo es tan rápido que apenas se nota los intervalos.

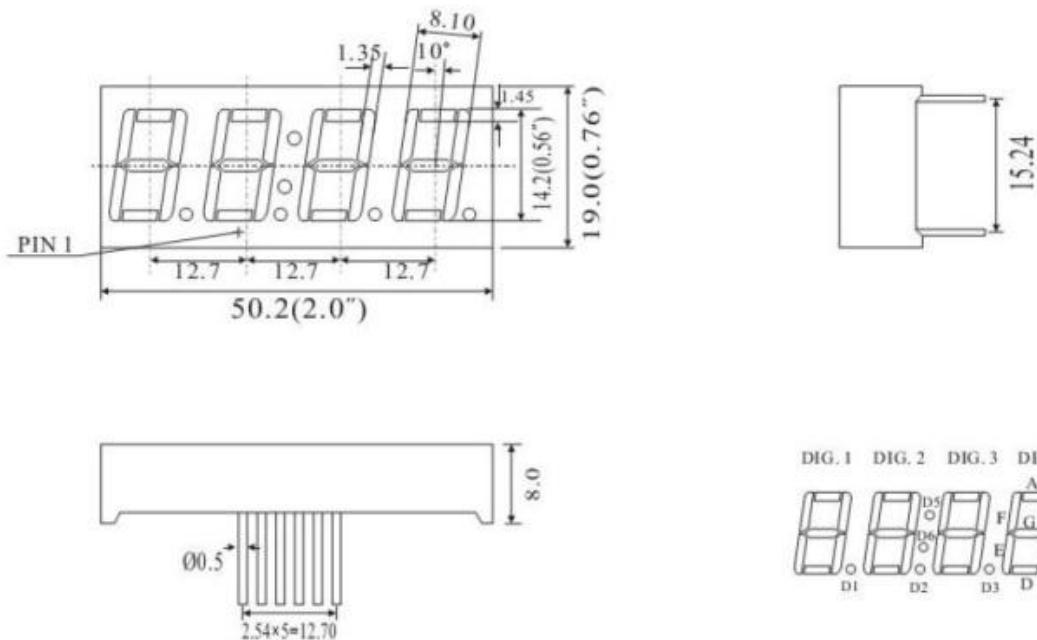
## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) Placa Universal
- (1) 74HC595
- (1) Visualizador de 7 segmentos de 4 dígitos
- (4) 220 ohm resistor
- (20) M-M alambres

## Introducción de Componente

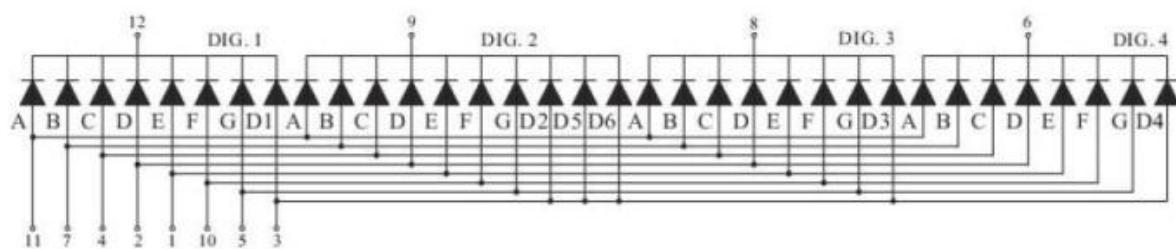
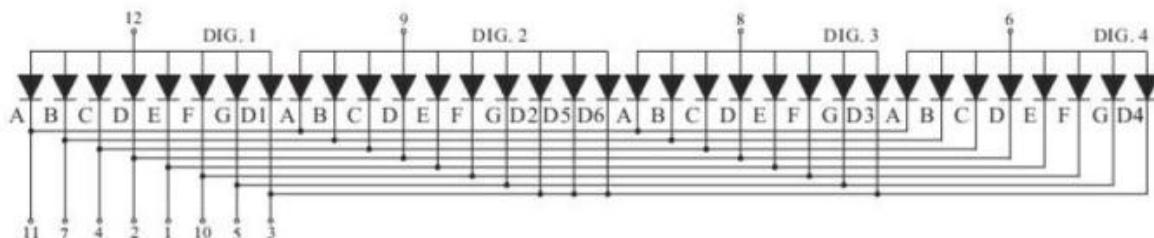
### Visualizador de 7 segmentos de 4 dígitos:

## Package Dimensions

**CPS05643AB**

UNIT: MM(INCH) TOLERANCE:  $\pm 0.25(0.01")$

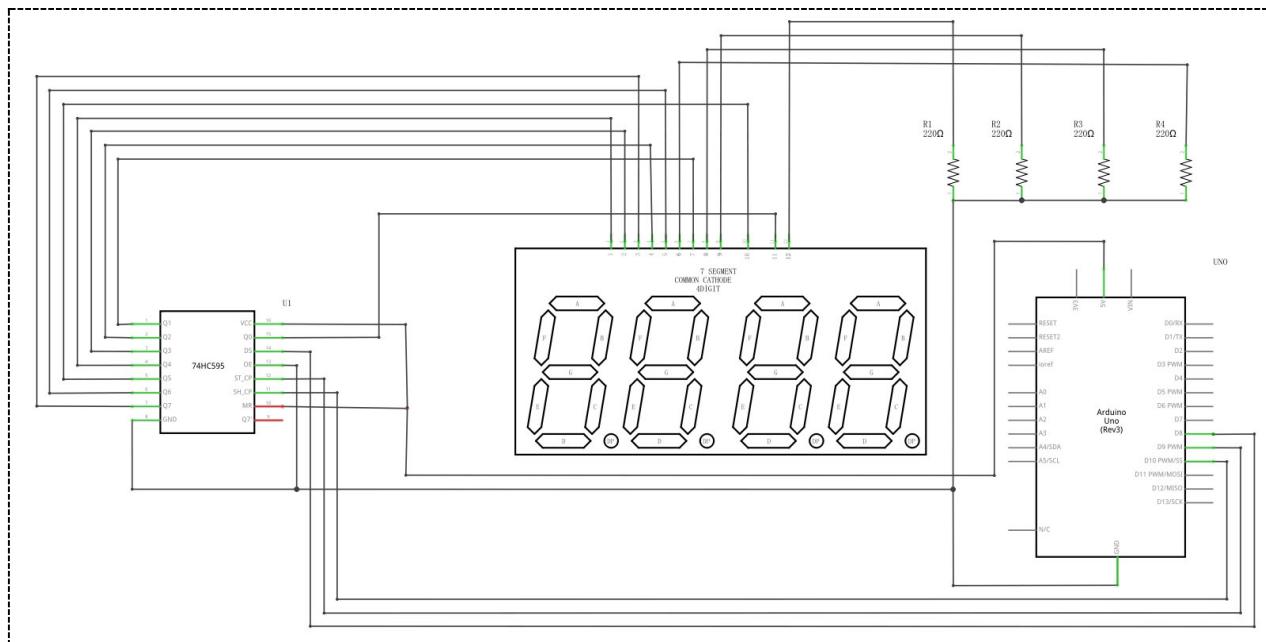
## Internal Circuit Diagram

**5643A****5643B**

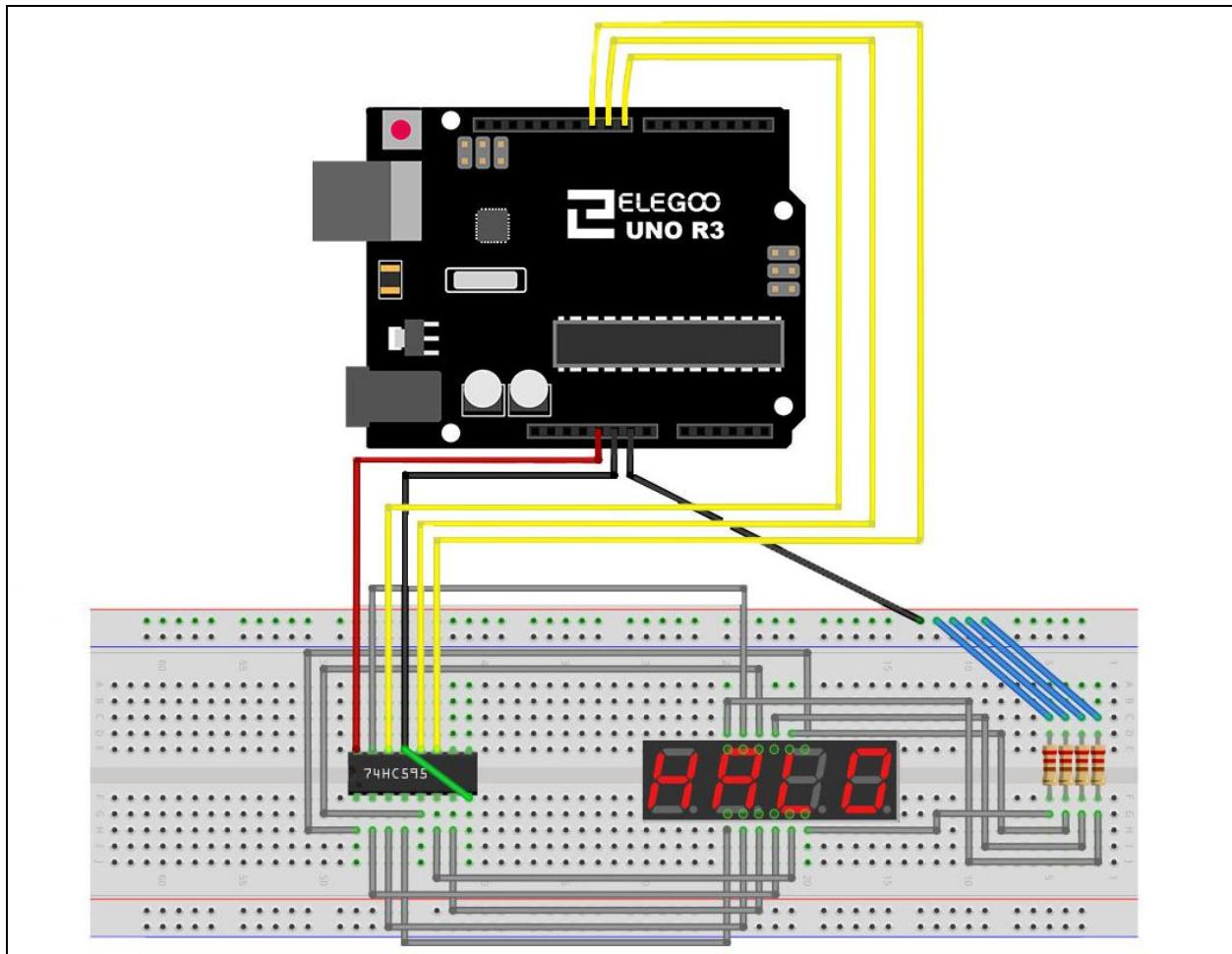
## Four Digits Displays Series

## Conexión

### Esquemático

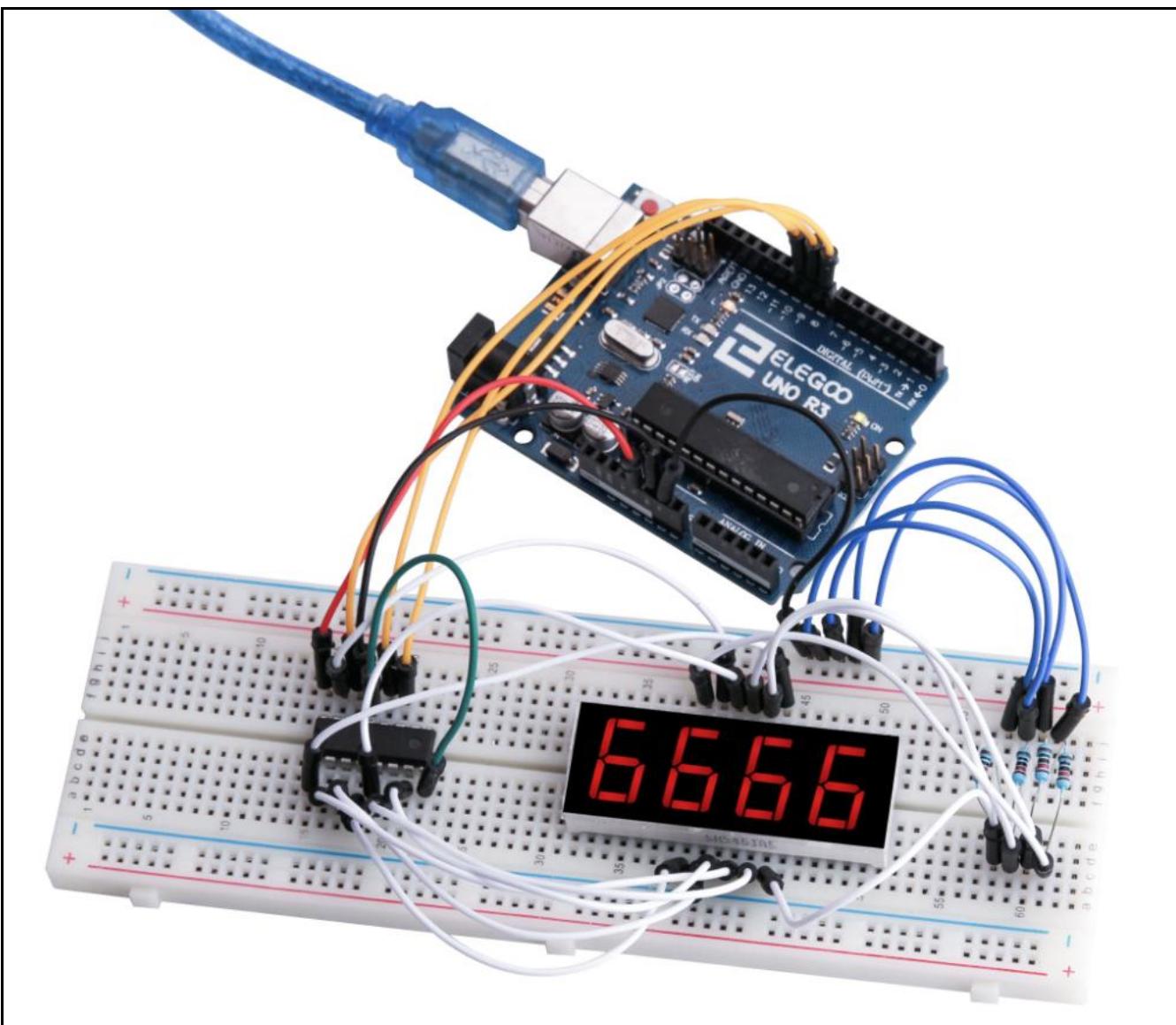


### Diagrama de cableado



## Código

Ver el archivo de código.



# Lección 15: Servo

## Visión general

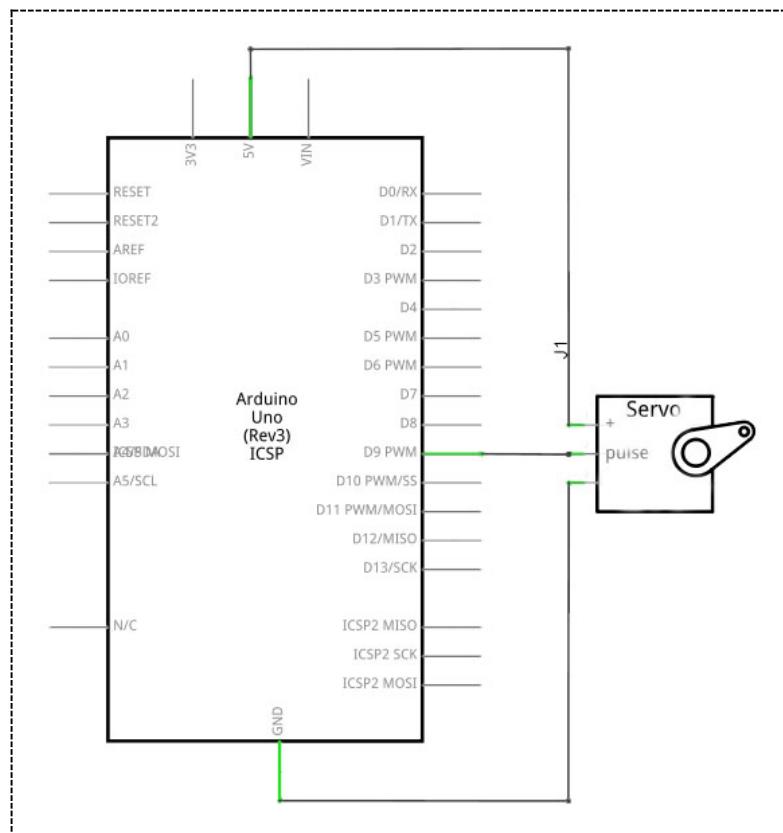
Servo es un tipo de motor reductor que sólo puede girar 180 grados. Está controlado mediante el envío de impulsos eléctricos de su placa de UNO R3. Estos pulsos dicen que el servo qué posición debe pasar a. Un servo tiene tres cables, el cable marrón es GND , el rojo es VCC, y el naranja es la línea de señal.

## Componentes Requeridos

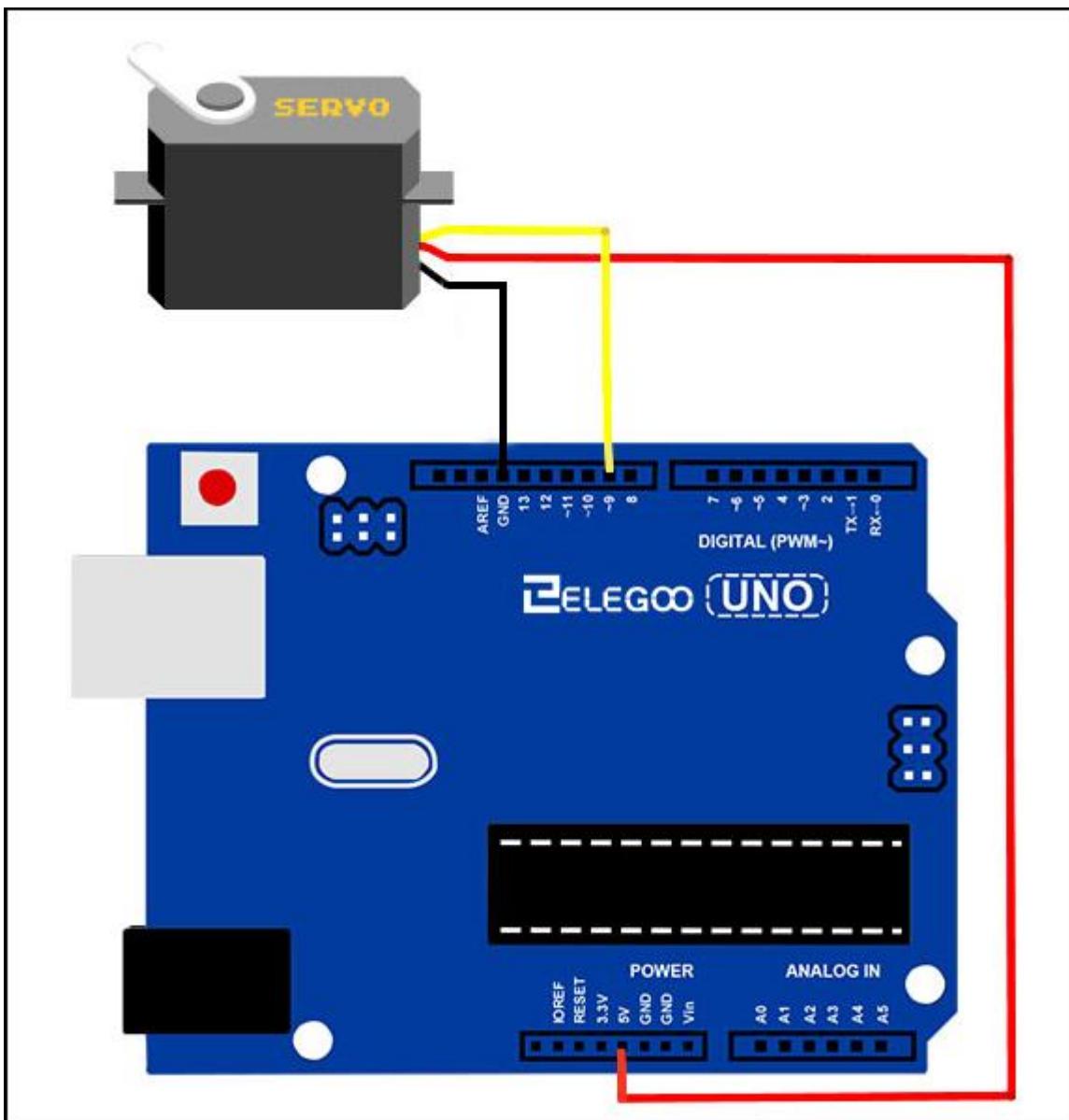
- (1) Elegoo UNO R3
- (1) Servo
- (2) F-M alambres

## Conexión

### Esquemático

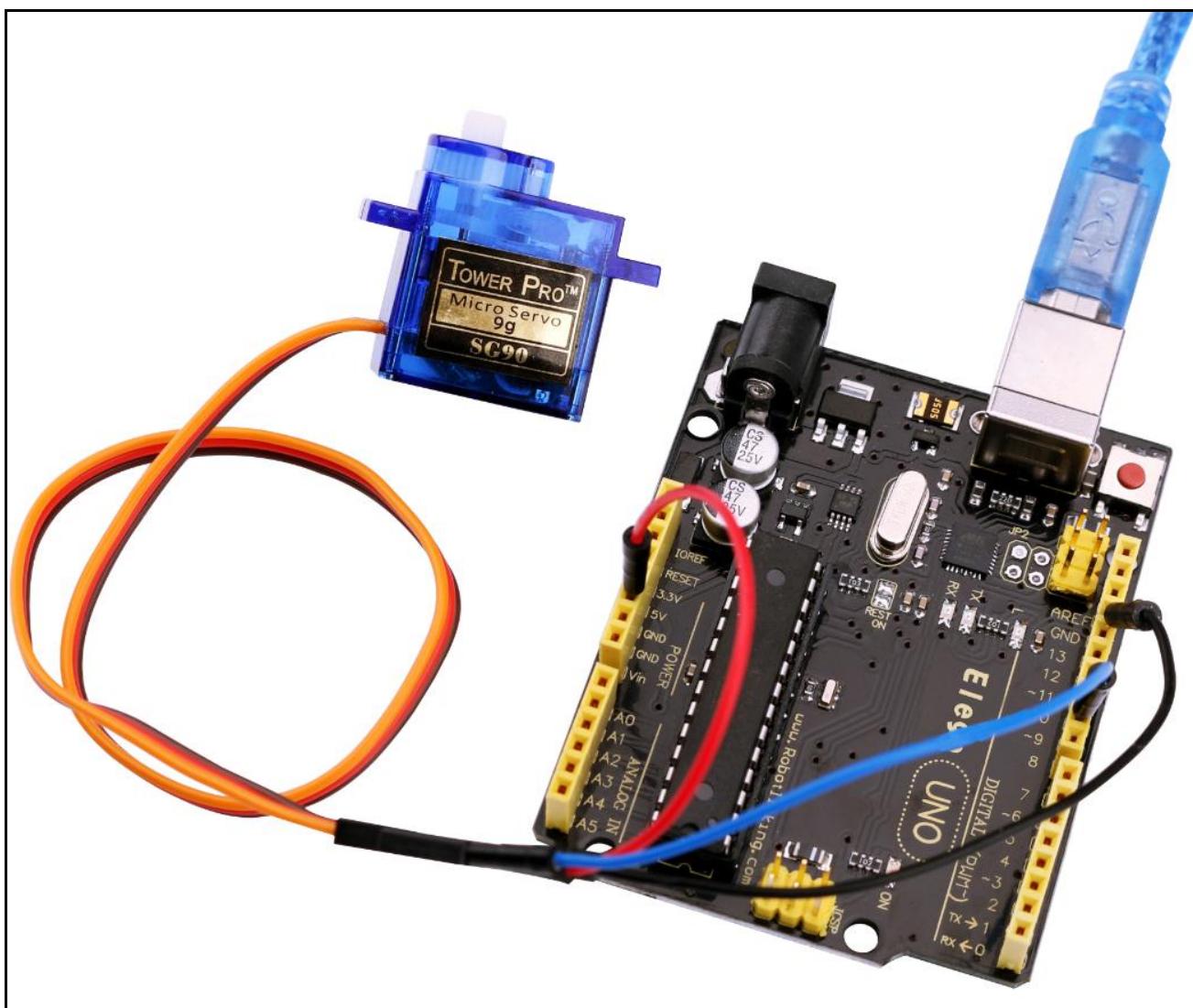


## Diagrama de cableado



## Código

[Ver el archivo de código.](#)



# Lección 16: Visualizador LCD

## Visión general

En esta lección, usted aprenderá cómo conectar y utilizar un visualizador LCD alfanumérico. La pantalla tiene una luz de fondo LED y puede mostrar dos filas con un máximo de 16 caracteres en cada fila. Se puede ver los rectángulos para cada personaje en el visualizador y los píxeles que componen cada carácter. El visualizador es sólo blanco en azul y está destinado a mostrar el texto.

En esta lección, vamos a ejecutar el programa de ejemplo para la biblioteca de Arduino LCD, pero en la siguiente Lección, obtendremos nuestro visualizador para mostrar el nivel de temperatura y la luz, el uso de sensores.

## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) LCD1602 visualizador
- (1) Potenciómetro
- (1) Placa Universal
- (16) F-M alambres

## Introducción de Componente

LCD1602:

**V<sub>SS</sub>**: Un perno que conecta a tierra

**V<sub>DD</sub>**: Un perno que se conecta a una fuente de alimentación de + 5V

**VO**: Un perno que ajustar el contraste de LCD 1602

**RS**: Un perno de selección de registro que controla el lugar de la memoria de LCD que está escribiendo datos. Puede seleccionar el registro de datos, que contiene lo que sucede en la pantalla, o un registro de instrucción, que es donde el controlador de la pantalla LCD se ve para obtener instrucciones sobre qué hacer a continuación.

**R/W**: Un perno de lectura / escritura que selecciona el modo de lectura o de escritura

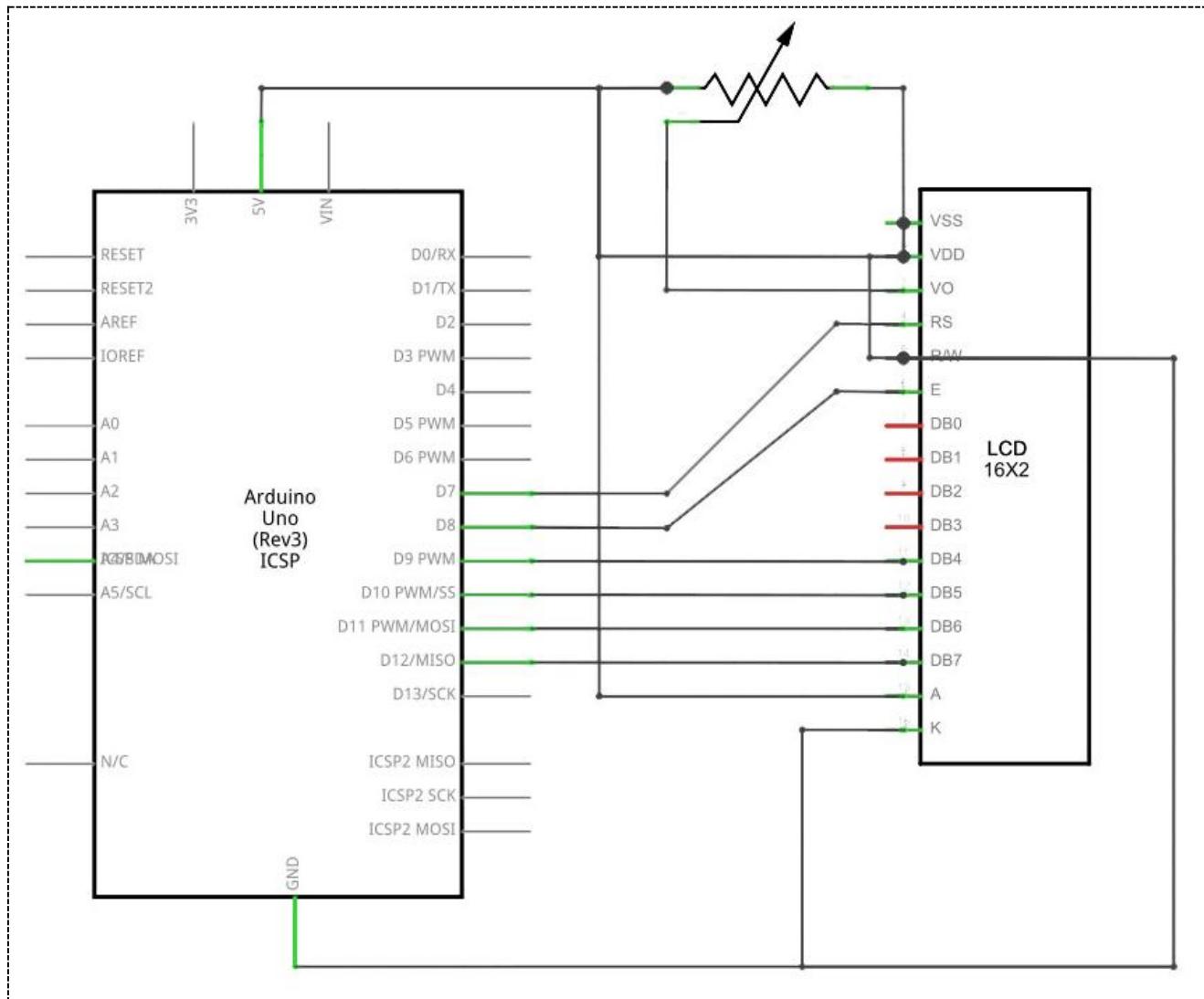
**E:** Un perno que permite que, cuando se suministra con energía de bajo nivel, hace que el módulo de LCD para ejecutar las instrucciones pertinentes.

**D0-D7:** Pernos que leen y escriben datos

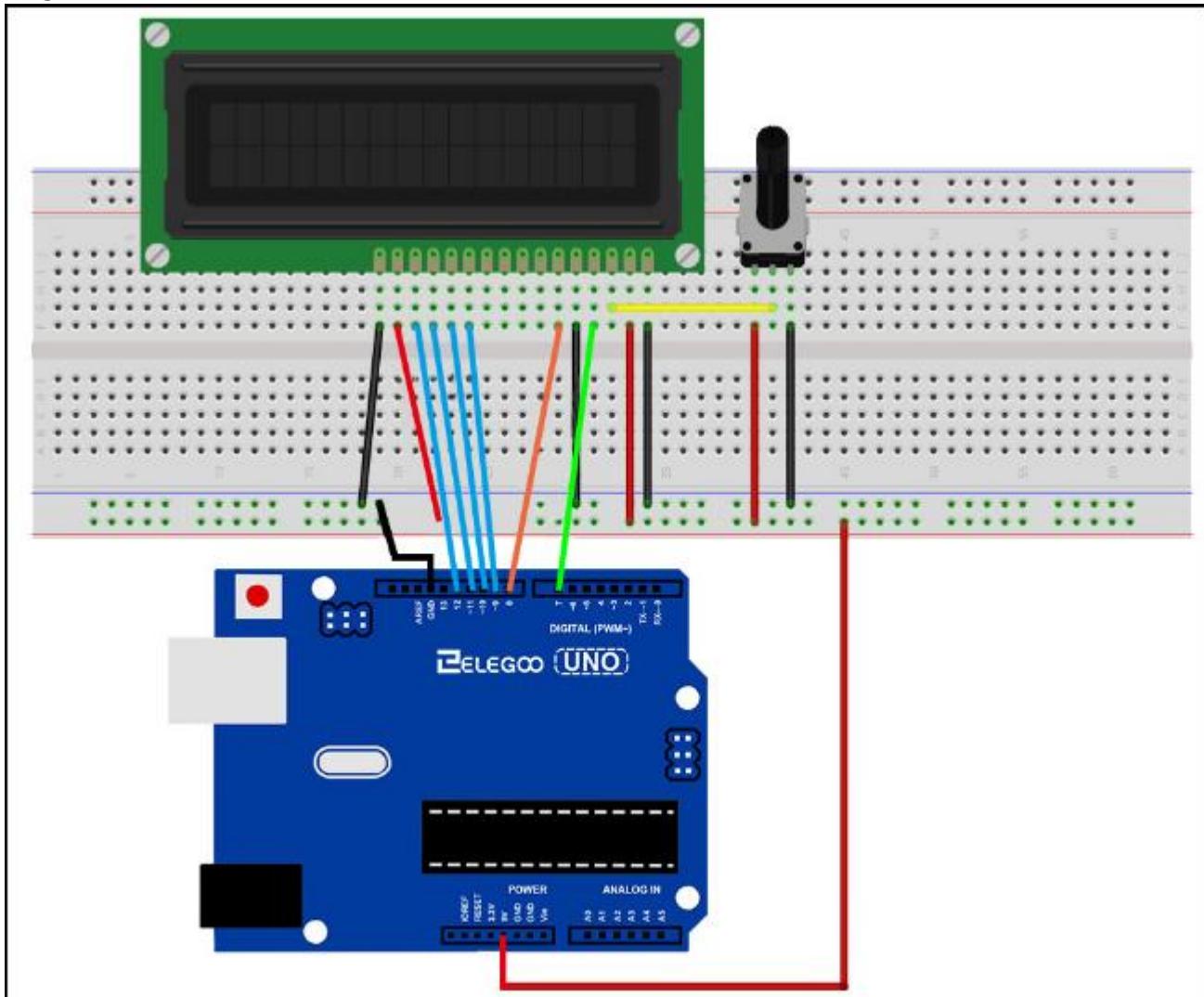
**A y K:** Pernos que controlan la luz de fondo LED

## Conexión

### Esquemático



## Diagrama de cableado



El visualizador LCD necesita seis pernos de Arduino, todo listo para ser salidas digitales. También necesita conexiones 5V y GND.

Hay un número de conexiones a realizar. La alineación del visualizador con la parte superior de la placa universal ayuda a identificar sus patas sin demasiada cuenta, sobre todo si la placa universal tiene sus filas numeradas con la fila 1 como la fila superior de la placa. No se olvide, el largo cable amarillo que une el control deslizante de la olla al perno 3 del visualizador. La "olla" se utiliza para controlar el contraste del visualizador.

Usted puede encontrar que su visualizador se suministra sin pernos del cabezal que se le atribuye. Si es así, siga las instrucciones de la sección siguiente

## Código

Ahora que tiene la configuración física, todo lo que necesitamos ahora es el código.

Antes de poder ejecutar esto, asegúrese de que ha instalado la biblioteca <Cristal Líquido> o volver a instalarlo de nuevo, si es necesario. De lo contrario, su código no funcionará

En este ejemplo se utiliza pernos diferentes a los que utilizamos, por lo que encontrar la línea de código a continuación:

Cristal Líquido lcd(12, 11, 5, 4, 3, 2);

y el cambio a:

Cristal Líquido lcd(7, 8, 9, 10, 11, 12);

Cargue el código para la placa de Arduino y debería ver el mensaje "hola, mundo 'que se muestra, seguido de un número que cuenta a partir de cero.

La primera cosa de la nota en el boceto es la línea:

#incluir < Cristal Líquido.h>

Esto le dice a Arduino que deseamos utilizar la biblioteca de cristal líquido.

Lo siguiente es la línea que tenemos que modificar. Esto define qué pernos del Arduino se van a conectar a cuyos pernos del visualizador.

Cristal Líquido lcd(7, 8, 9, 10, 11, 12);

Los argumentos a esto son los siguientes:

Nombre de Perno de Visualizar Nombre de Perno de Visualizar Perno de Arduino(en este ejemplo) RS 4 7 E 6 8 D4 11 9 D5 12 10 D6 13 11 D7 14 12

Después de cargar el código, asegúrese de que la luz de fondo se iluminó, y ajustar el potenciómetro de todo el camino hasta que vea el mensaje de texto.

En la función 'configuración', tenemos dos comandos:

lcd. empezar (16, 2);

lcd. imprimir ("hola,mundo!");

El primera es la la biblioteca de cristal líquido el número de columnas y filas de la pantalla tiene. La segunda línea muestra el mensaje que vemos en la primera línea de la pantalla.

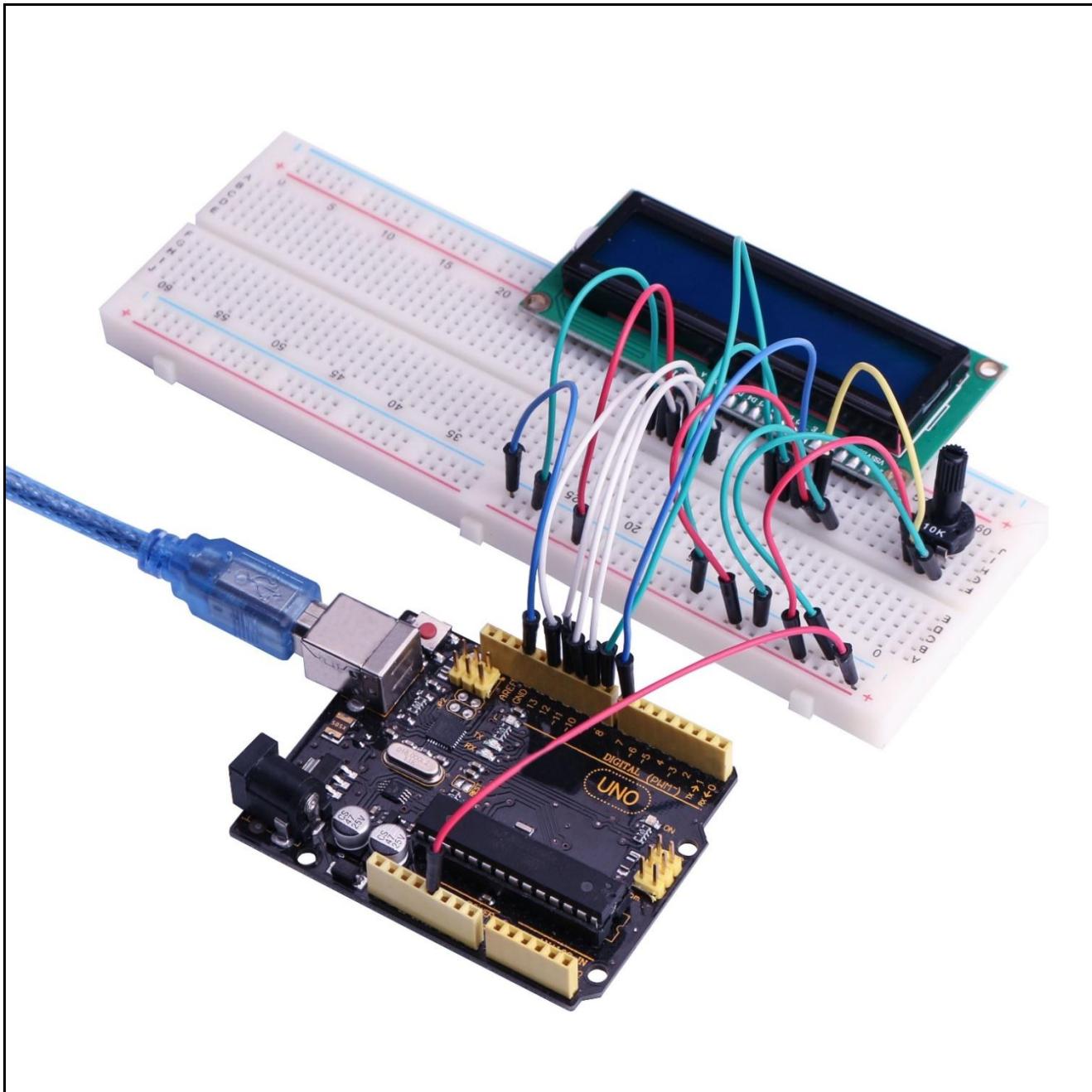
En la función de 'bucle', que aso tenemos dos comandos:

lcd. ajuste de cursor (0, 1);

lcd.imprimir (millis()/1000);

La primera conjuntos de la posición del cursor (en la que aparecerá el siguiente texto) a la columna 0 y la fila 1. Ambos números de fila y columna comienzan en 0 en lugar de 1.

La segunda línea muestra el número de milisegundos desde que se restableció el Arduino.



## Lección 17: Termómetro

## **Visión general**

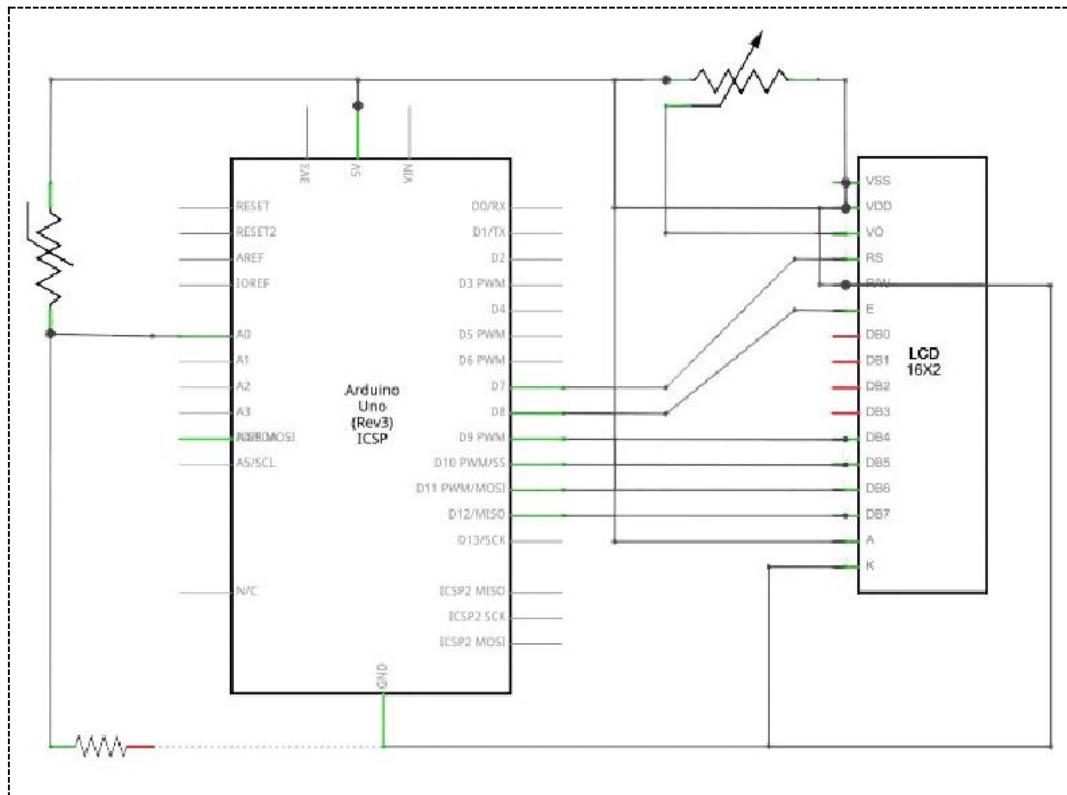
En esta lección, que va a utilizar un visualizador LCD para mostrar la temperatura.

## Componentes Requeridos

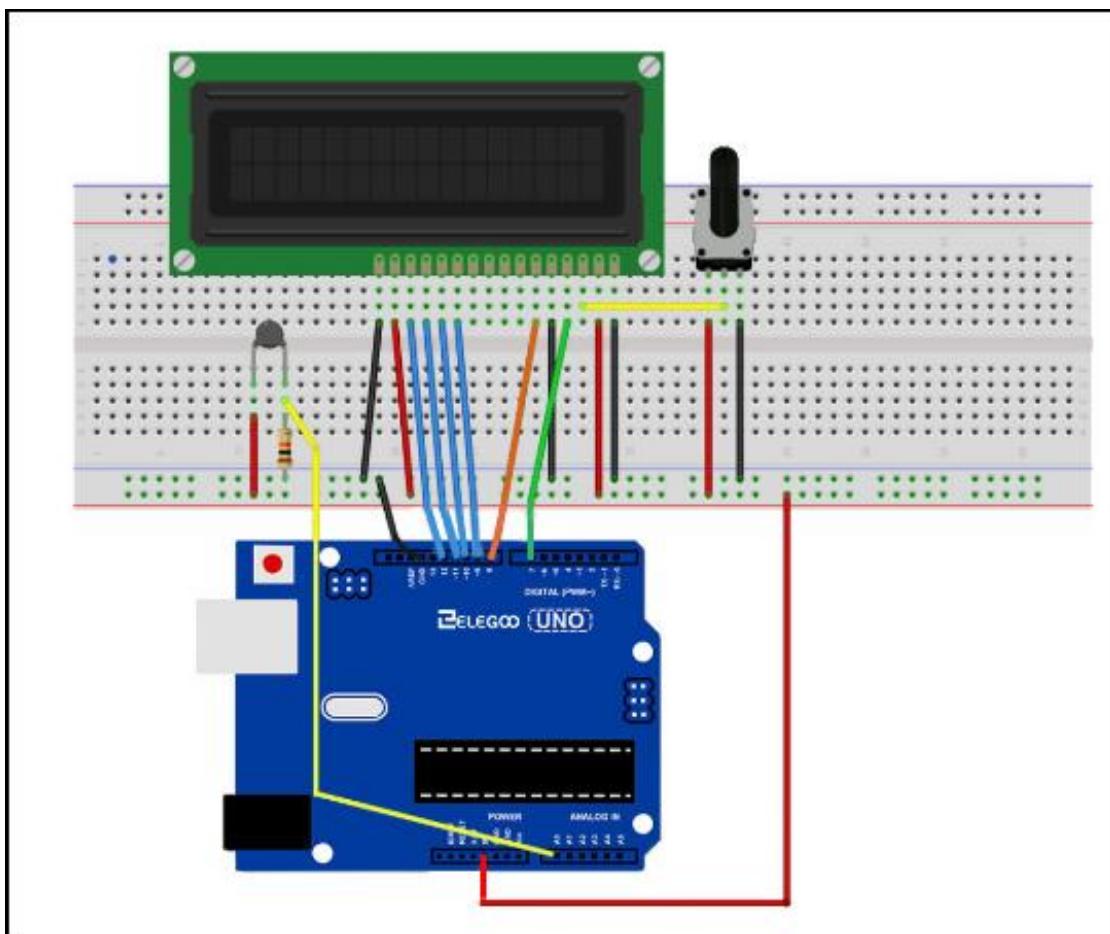
- (1) Elegoo UNO R3
  - (1) LCD1602 visualizador
  - (1) 10k ohm resistor
  - (1) Termistor
  - (1) Potenciómetro
  - (1) Placa Universal
  - (16) F-M alambres

## Conexión

## **Esquemático**



## Diagrama de cableado



La disposición de placa universal se basa en la disposición de la lección 16, por lo que va a simplificar las cosas si todavía tiene esto en la placa universal.

Hay unos pocos cables de puente cerca de la olla que se han movido ligeramente en esta disposición

La fotocélula, el resistor 1 kΩ y el termistor son todas las nuevas incorporaciones a la placa.

## Código

**Antes de poder ejecutar esto, asegúrese de que ha instalado el <Cristal Líquido> biblioteca o volver a instalarlo, si es necesario. De lo contrario, su código no funcionará.**

El boceto para esto se basa en la de Lección 14. Carga para arriba sobre la placa Arduino y usted debe encontrar que el calentamiento del sensor de temperatura poniendo el dedo sobre ella aumentará la lectura de la temperatura.

Me parece útil para poner una línea de comentario sobre el comando 'lcd'.

```
//      BSE    D4  D5  D6  D7
```

## Cristal Líquido Lcd(7, 8, 9, 10, 11, 12);

Esto resultará más fácil si usted decide cambiar qué pernos se utiliza.

En la función de "bucle" en la actualidad hay dos cosas interesantes que ocurren. En primer lugar tenemos que convertir el análogo del sensor de temperatura en una temperatura real, y en segundo lugar tenemos que encontrar la manera de mostrarlos.

En primer lugar, vamos a ver el cálculo de la temperatura.

```
int lectura de temperatura = leído (Perno de temperatura );
```

```
voltios de temperatura de flotador = lectura de temperatura * 5.0 / 1024.0;
```

```
temperatura de flotador C = (voltios de temperatura - 0.5) * 100.0;
```

```
temperatura de flotador F = temperatura C * 9.0 / 5.0 + 32.0;
```

La lectura en bruto del sensor de temperatura primero se multiplica por 5 y luego se divide por 1024 para darnos la tensión (entre 0 y 5) en la entrada analógica del 'Perno de temperatura'.

Para convertir el voltaje procedente de la TMP36 en una temperatura en grados C, usted tiene que restar 0.5 V a partir de la medición y luego se multiplica por 100.

Para convertir esto en una temperatura en grados Fahrenheit, y luego tienes que multiplicar por 9/5 y luego añadir 32.

Viendo que cambian las lecturas en el visualizador LCD puede ser complicado. El principal problema es que la lectura puede no ser siempre el mismo número de dígitos. Por lo tanto, si la temperatura cambia 101,50-99,00 entonces el dígito adicional de la edad de lectura está en peligro de siendo a la izquierda en el visualizador.

Para evitar esto, escribir toda la línea del visualizador LCD cada vez alrededor del bucle.

```
// -----
```

```
Lcd. imprimir ("Temperatura      F      ");
```

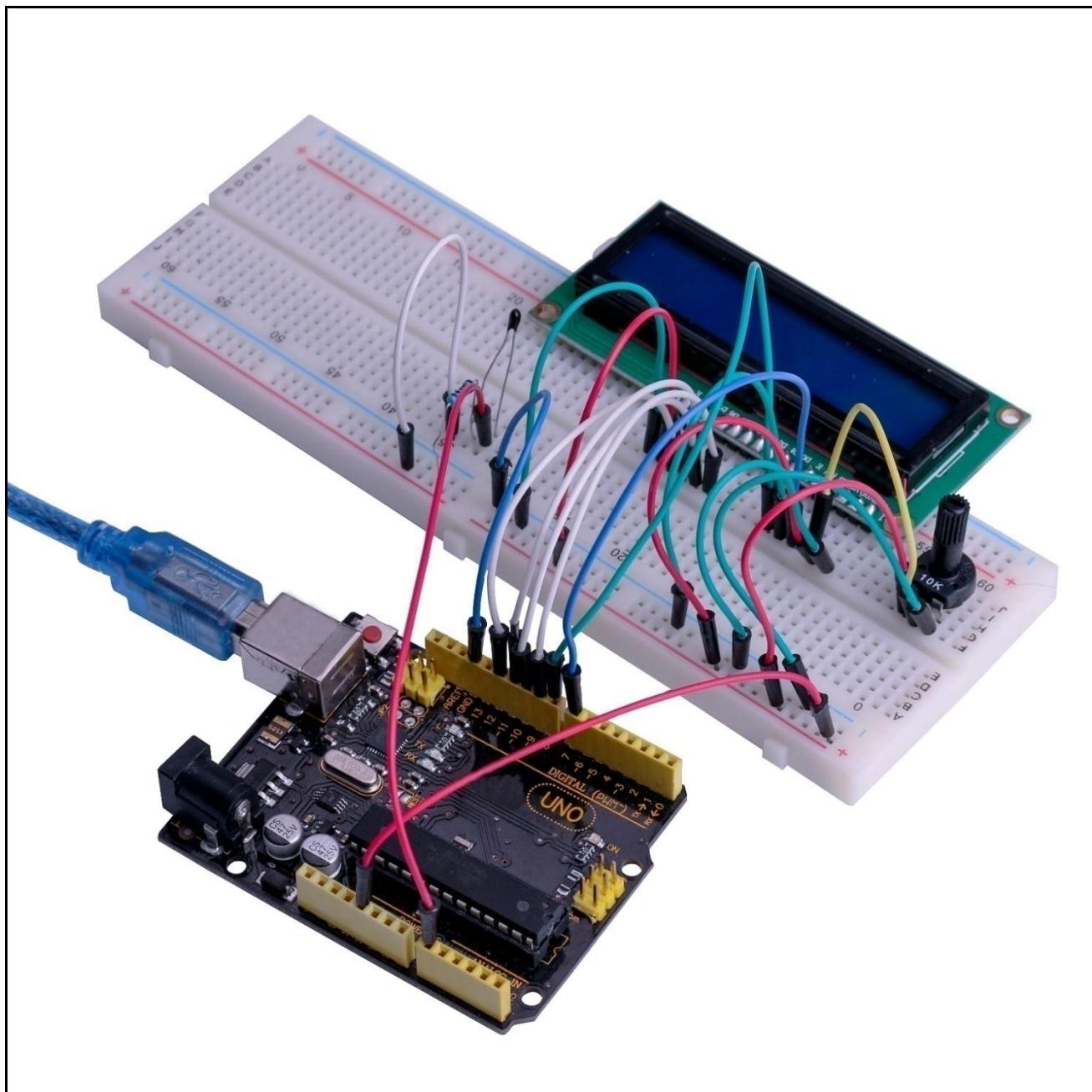
```
Lcd. ajuste de cursor (6, 0);
```

```
Lcd. imprimir (temperatura F);
```

El bastante extraño comentario sirve para recordarle las 16 columnas del visualizador. A continuación, puede imprimir una cadena de esa longitud con espacios donde la lectura real tope.

Para llenar los espacios en blanco, ajuste la posición del cursor de donde debería aparecer la lectura y luego imprimirlo.

Exactamente el mismo método se utiliza para mostrar el nivel de luz. No hay unidades para el nivel de luz, que acabamos de mostrar la lectura prima a partir de la lectura analógica.



# Lección 18: Módulo de Sensor Ultrasónico

## Visión general

Sensor de ultrasonidos son ideales para todo tipo de proyectos que necesitan mediciones de distancia, evitando obstáculos como ejemplos.

El HC-SR04 son de bajo costo y fácil de usar ya que vamos a utilizar una biblioteca diseñada específicamente para estos sensores.

## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) Módulo de Sensor Ultrasónico
- (4) F-M alambres

## Introducción de Componente

### Sensor Ultrasónico

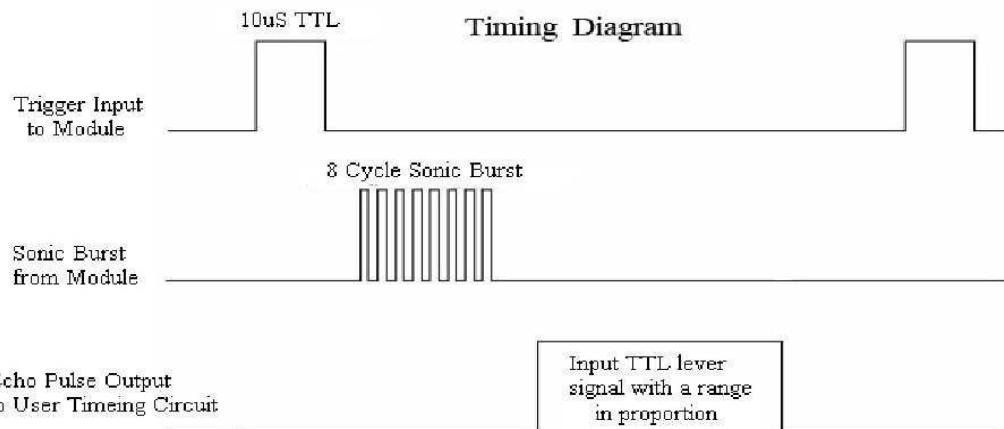
Módulo de alcance ultrasónico HC - SR04 ofrece 2 cm - 400 cm función de medición sin contacto, la precisión que oscila puede llegar a 3 mm Los módulos incluye transmisores ultrasónicos, el receptor y el circuito de control del principio básico de trabajo .:

- (1) Utilizando disparador de IO por lo menos durante 10us señal de alto nivel,
- (2) El módulo automáticamente envía ocho de 40 kHz y detectar si hay una señal de impulso de nuevo.
- (3) Si la señal de vuelta, a través de alto nivel, el tiempo de duración de alto rendimiento IO es el tiempo desde el envío de giro del neumático ultrasónica.

Distancia de prueba = (tiempo de alto nivel × velocidad del sonido (340M/S) /2

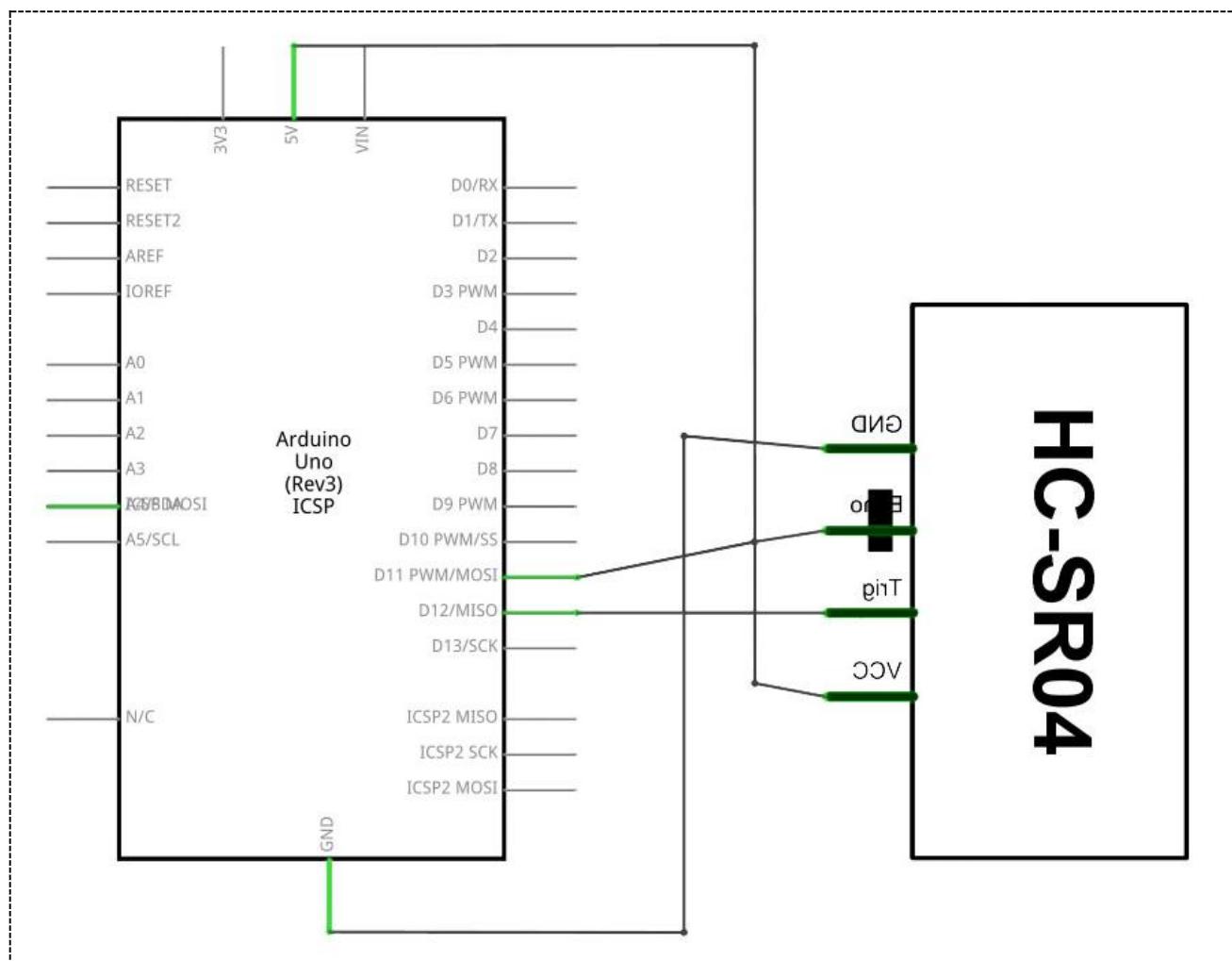
El diagrama de temporización se muestra a continuación. Sólo tiene que suministrar un pulso corto 10uS a la entrada de disparo para iniciar el rango, y luego el módulo va a enviar una ráfaga de 8 ciclo de ultrasonidos a 40 kHz y elevar su eco. El eco es un objeto de distancia que es ancho de pulso y el rango en proporción Usted puede calcular el rango a través del intervalo de tiempo entre el envío y la recepción de la señal de disparo de la señal de eco.

Fórmula:  $\mu\text{s} / 58 = \text{centímetros}$  o  $\mu\text{s} / 148 = \text{pulgada}$ ; o: el rango = tiempo de alto nivel \* velocidad (340M/S) / 2; se aconseja utilizar ciclo de medición más de 60 ms, con el fin de evitar que la señal de disparo de la señal de eco.

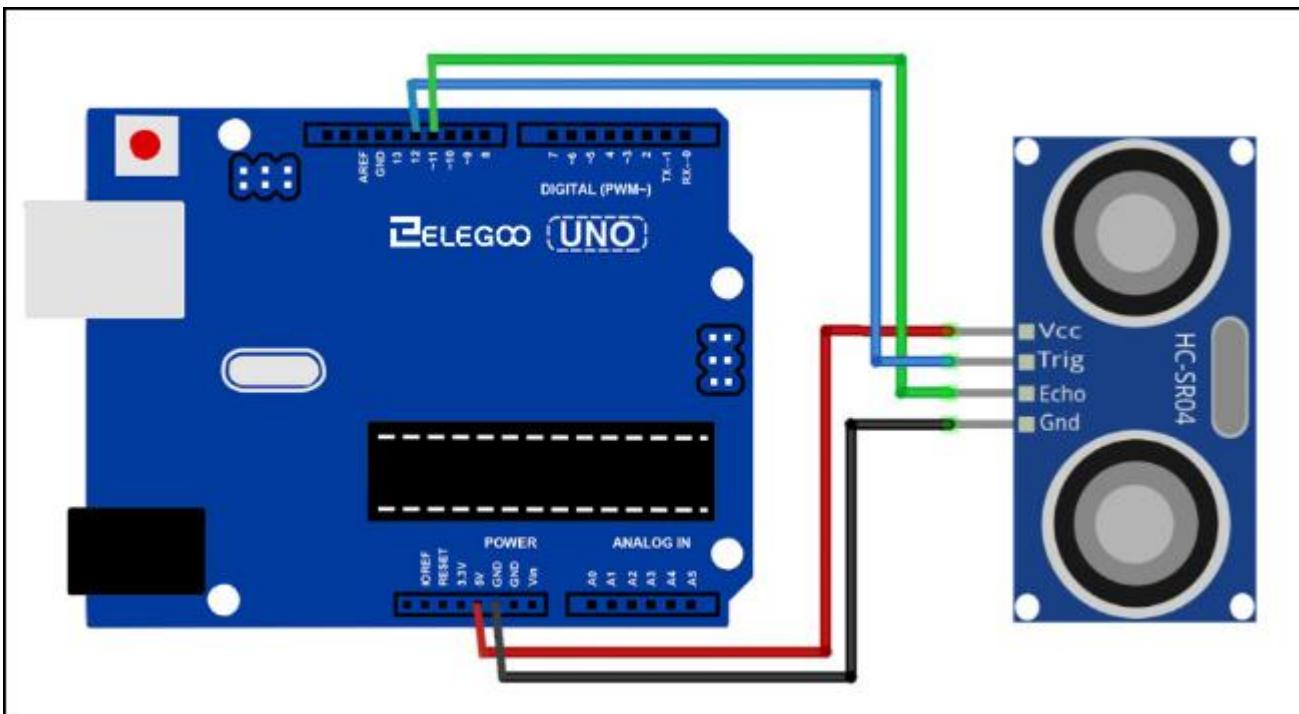


## Conexión

### Esquemático



## Diagrama de cableado



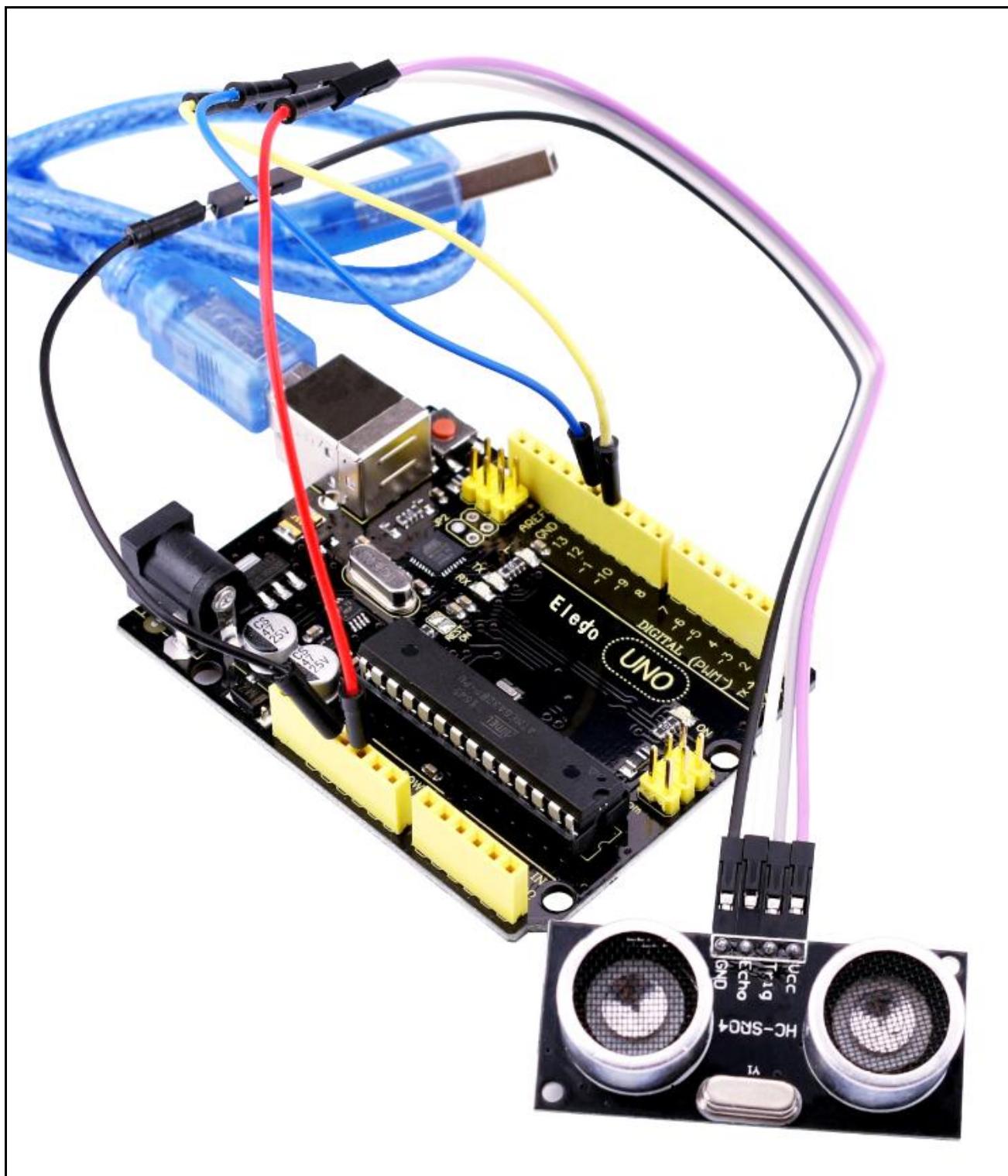
## Código

El uso de una biblioteca diseñada para estos sensores hará que nuestro código corto y simple.

Incluimos la biblioteca a principios de nuestro código, y luego mediante el uso de comandos simples que podemos controlar el comportamiento del sensor.

**Ahora que tenemos la configuración física, todo lo que necesitamos ahora es el código.**

**Antes de poder ejecutar esto, usted tiene que asegurarse de que tiene instalar la biblioteca <Biblioteca HC-SR04>. O es necesario instalar de nuevo. Si no lo hace, su código no funcionará.**



# Lección 19: DHT11 Sensor de Temperatura y Humedad

## Visión general

En este tutorial vamos a aprender cómo usar DHT11 Sensor de Temperatura y Humedad.

Es lo suficientemente preciso para la mayoría de los proyectos que necesitan para realizar un seguimiento de las lecturas de humedad y temperatura.

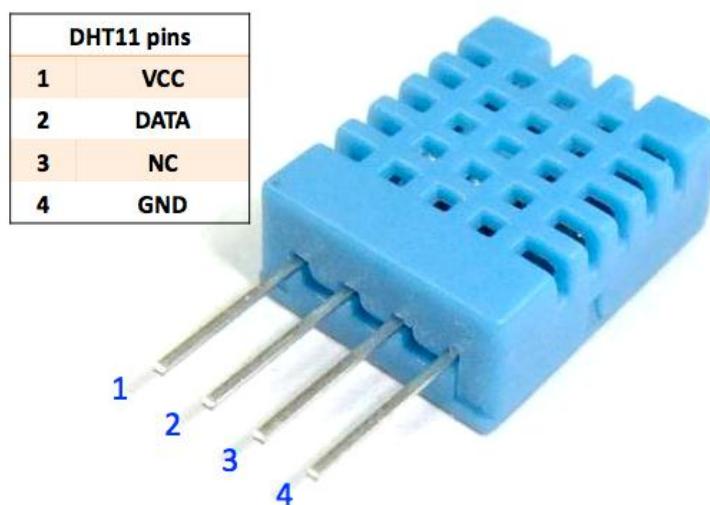
Una vez más vamos a utilizar una biblioteca diseñada específicamente para estos sensores que hará que nuestro código corto y fácil de escribir.

## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) DHT11 módulo
- (3) F-M alambres

## Introducción de Componente

### Sensor de Temperatura y Humedad:



DHT11 Sensor digital de Temperatura y Humedad es un sensor compuesto contiene una salida de señal digital calibrado de la temperatura y la humedad. La aplicación de una tecnología dedicada colección módulos digitales y la tecnología de detección de la temperatura y la humedad, para asegurar que el producto tiene una alta fiabilidad y una excelente estabilidad a largo plazo. El sensor incluye un sentido del resistor de los componentes

húmedos y un dispositivos de medición de temperatura NTC, y conectado con un alto rendimiento microcontrolador de 8 bits.

Aplicaciones: HVAC, deshumidificador, las pruebas y la inspección de equipos, bienes de consumo, automoción de control, automático, registradores de datos, estaciones meteorológicas, electrodomésticos, regulador de la humedad, médica y otra medición de la humedad y control.

## **parámetros del producto**

Humedad relativa:

Resolución: 16Bit

Repetibilidad:  $\pm 1\%$  RH

Exactitud: En  $25^{\circ}\text{C}$   $\pm 5\%$  RH

Intercambiabilidad: totalmente intercambiables

Tiempo de respuesta: 1 / e (63%) de  $25^{\circ}\text{C}$  6s

1m / s aire 6s

Histéresis:  $<\pm 0.3\%$  RH

Estabilidad a largo plazo:  $<\pm 0.5\%$  RH / yr in

Temperatura:

Resolución: 16Bit

Repetibilidad:  $\pm 0.2^{\circ}\text{C}$

Rango: At  $25^{\circ}\text{C}$   $\pm 2^{\circ}\text{C}$

Tiempo de respuesta: 1 / e (63%) 10s

Características electricas

Fuente de alimentación: DC 3.5~5.5V

Corriente de suministro: medición de 0.3 mA espera de  $60\mu\text{A}$

Período de muestreo: más de 2 segundos

Descripción de perno:

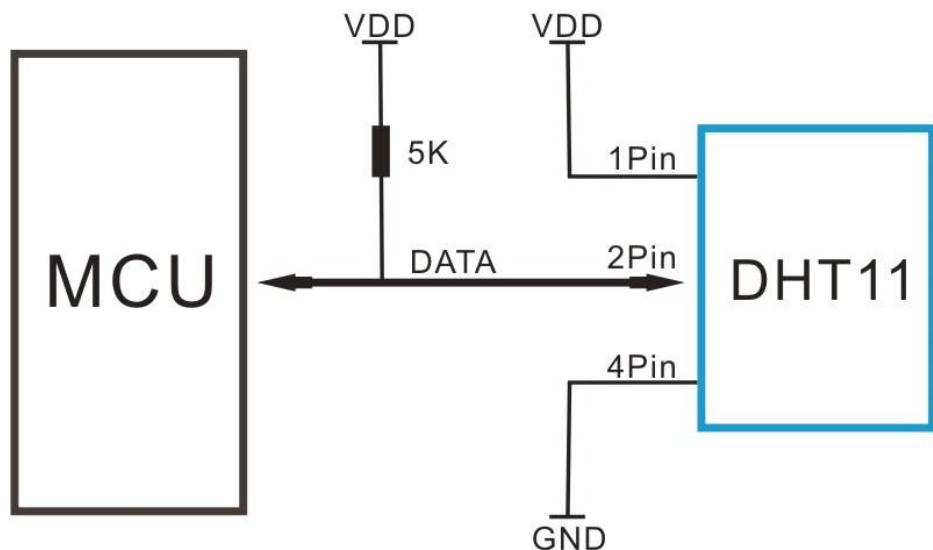
1, Fuente de alimentación VDD 3.5~5.5V DC

2 DATO dato serial, un único bus

3, NC, perno vacío

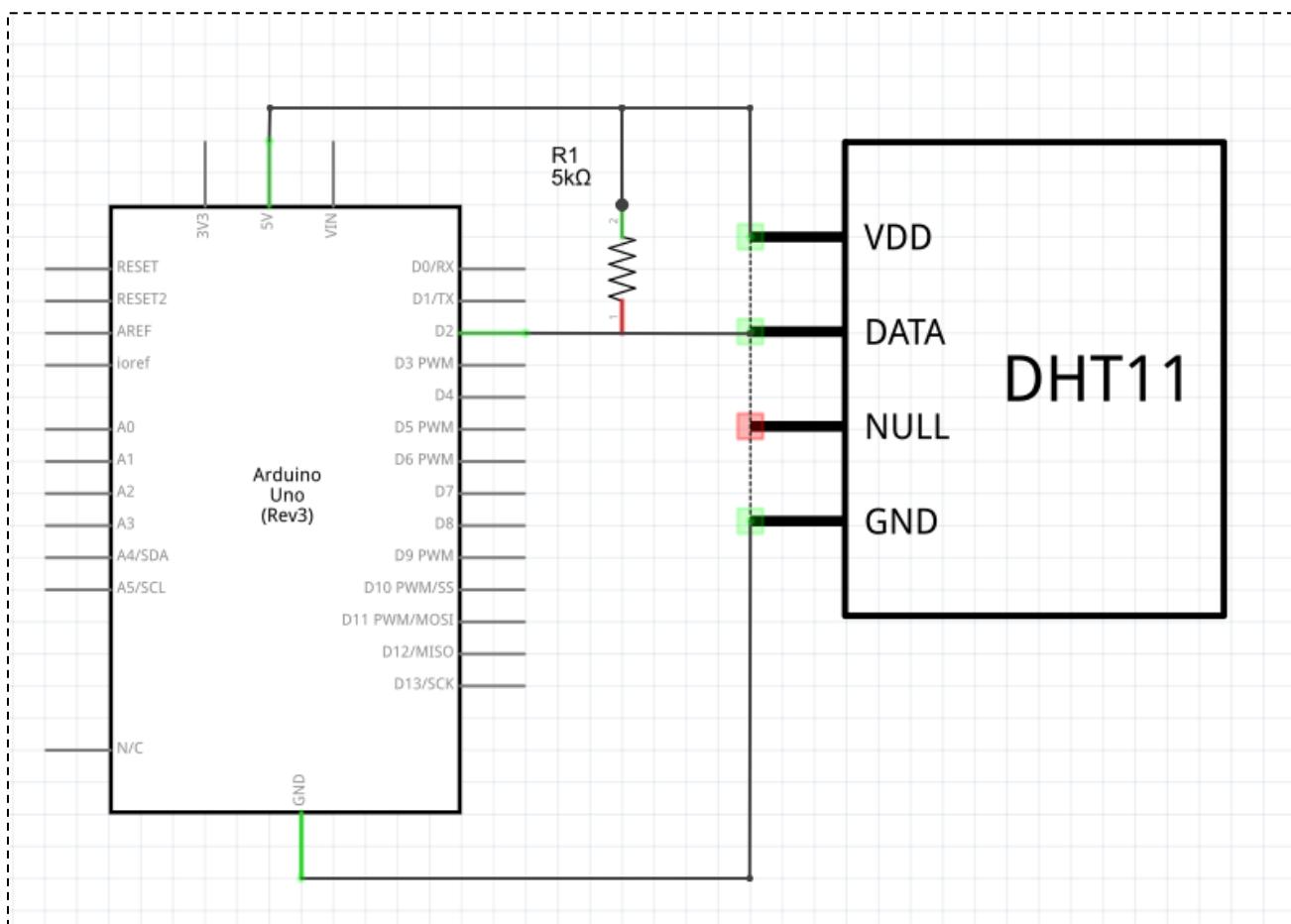
4, Tierra GND, el poder negativo

## **Aplicación típica**

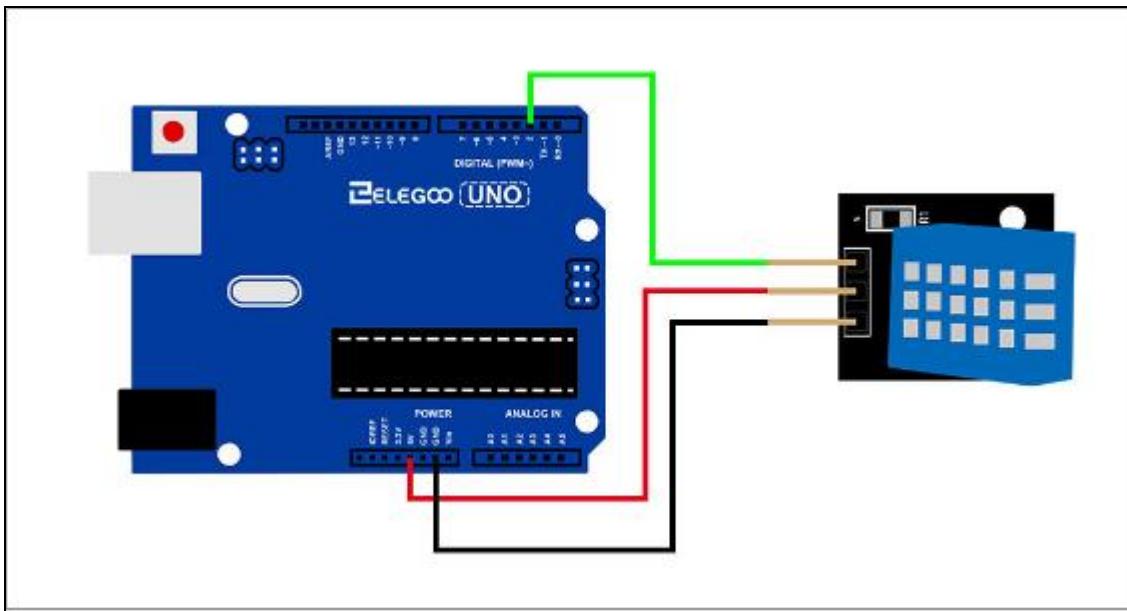


## Conexión

### Esquemático



## Diagrama de cableado



Como se puede ver sólo tenemos 3 conexiones al sensor, ya que uno de los pernos no se usa. La conexión son: Tensión, Tierra y Señal que se puede conectar a cualquier perno analógico en nuestra UNO.

## Código

Ahora que tenemos la configuración física, todo lo que necesitamos ahora es el código.

Antes de poder ejecutar esto, usted tiene que asegurarse de que tiene instalar la biblioteca <simple DHT>. O es necesario instalar de nuevo. Si no lo hace, su código no funcionará.

Cargue el programa a continuación, abra el monitor, podemos ver los datos como sigue:  
(Muestra la temperatura del medio ambiente, podemos ver que es de 21 grados)

```
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Read DHT11 failed=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
```

# Lección 20: Módulo de palanca de mando analógica

## Visión general

Palancas analógicas son una gran manera de añadir un poco de control en sus proyectos.

En este tutorial vamos a aprender a utilizar el módulo de palanca analógica.

## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) Módulo de palanca
- (5) F-M alambres

## Introducción de Componente

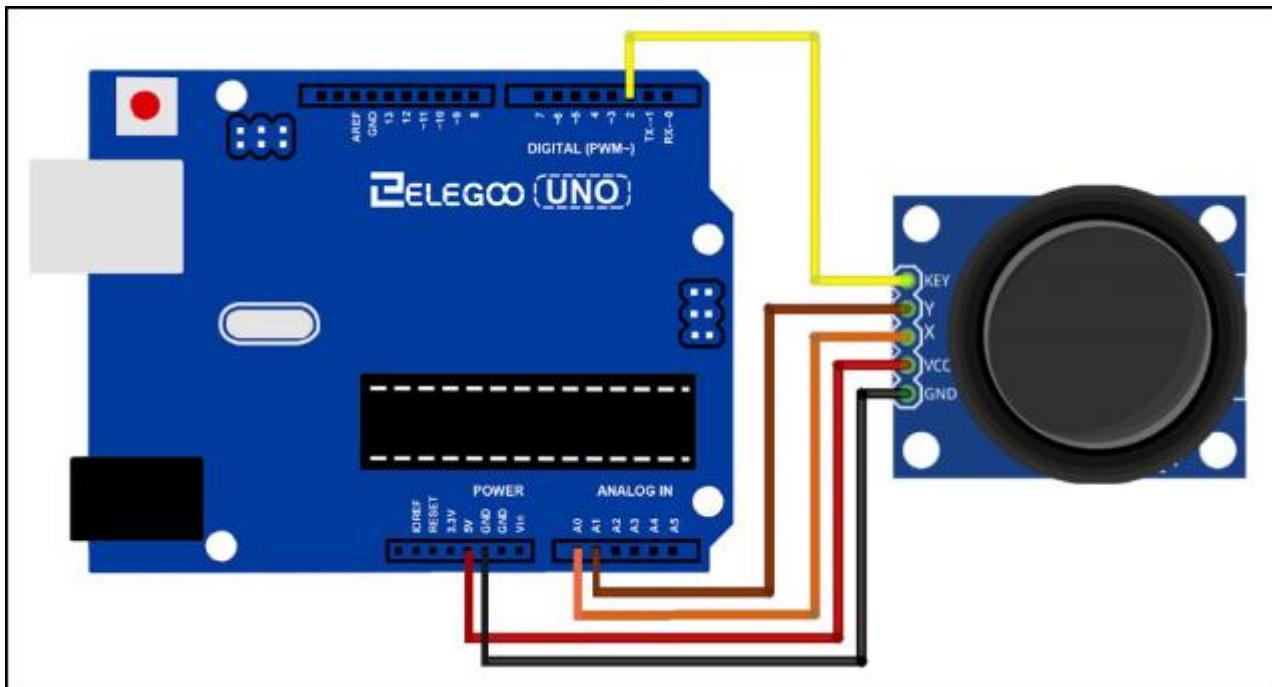
### Palanca

El módulo tiene 5 pernos Vcc, Tierra, X, Y, tecla. Tenga en cuenta que las etiquetas de los suyos pueden ser ligeramente diferentes, dependiendo de dónde sacó el módulo. El palo de oro es analógica y debe proporcionar lecturas más precisas que un simple 'direccional'. Palancas uso tacto algunas formas de botones o interruptores mecánicos. Además, puede pulsar la palanca hacia abajo (más bien duro en el mío) para activar un botón de empuje 'pulse para seleccionar'.

Tenemos que usar pernos de Arduino analógicos para leer los datos de los pernos X / Y, y un perno digital para leer el botón. El perno de tecla está conectada a tierra, cuando se presiona la palanca de mando hacia abajo, y está flotando en contrario. Para obtener lecturas estables de la Tecla / Perno de seleccionar, que necesita ser conectado a Vcc a través de un resistor de tracción. El construido en resistencias en los pernos digitales de Arduino se puede utilizar. Para ver un tutorial sobre cómo activar el resistor de tracción de pernos de Arduino, configurados como entradas

## Conexión

### Diagrama de cableado



Necesitamos 5 conexiones con la palanca de mando.

La conexión son: Tecla, Y, X, Voltaje y Tierra.

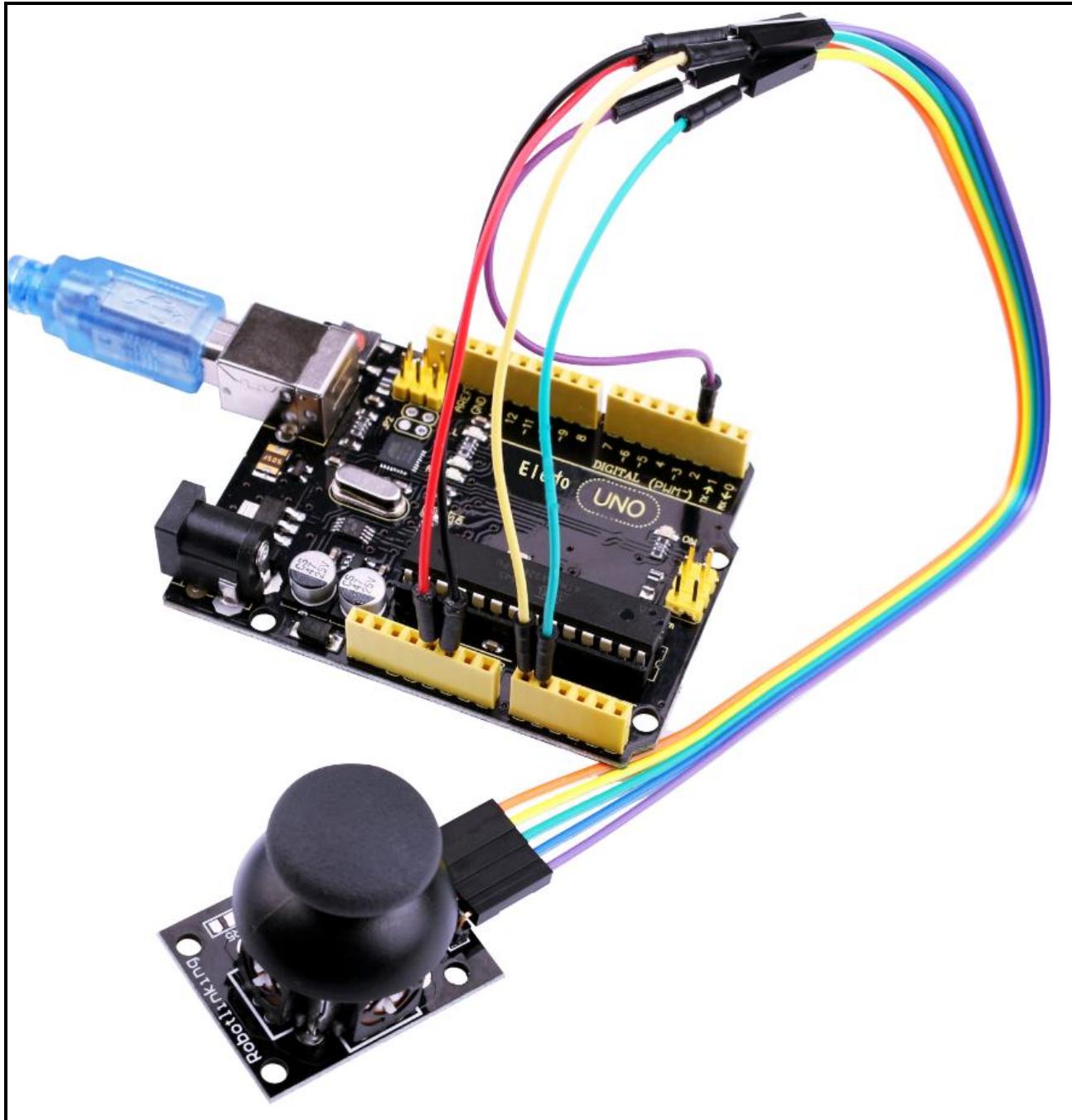
"Y y X" son analógicas y "Tecla" es digital. Si usted no necesita el interruptor entonces se puede usar sólo 4 pernos.

## Código

Palancas analógicas son básicamente Potenciómetros para que devuelven valores analógicos.

Cuando la palanca de mando está en la posición de reposo o en el medio, que debe devolver un valor de alrededor de 512.

El rango de valores van desde 0 hasta 1024



# Lección 21: Módulo de Receptor IR

## Visión general

El uso de un control remoto IR es una gran manera de tener el control inalámbrico de su proyecto.

Controles remotos infrarrojos son simples y fáciles de usar. En este tutorial vamos a conectar el receptor IR a la UNO, y luego usar una biblioteca que fue diseñado para este sensor en particular.

En nuestro boceto tendremos todos los códigos IR hexadecimales que están disponibles en este control remoto, también vamos a detectar si se ha reconocido el código y también si estamos manteniendo pulsada una tecla.

## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) Módulo del receptor IR
- (1) Remoto IR
- (3) F-M alambres

## Introducción de Componente

### SENSOR DE RECEPTOR IR:

Los detectores de IR son pequeños microchips con una fotocélula que están sintonizados para escuchar a la luz infrarroja. Casi siempre se utiliza para la detección de control remoto - cada televisor y reproductor de DVD tiene uno de estos en la parte delantera para escuchar la señal IR del Clickador. En el interior del mando a distancia es un LED IR coincidente, que emite pulsos IR para decirle al televisor para activar, desactivar o cambiar de canal. Luz infrarroja no es visible para el ojo humano, lo que significa que se necesita un poco más de trabajo para probar una configuración.

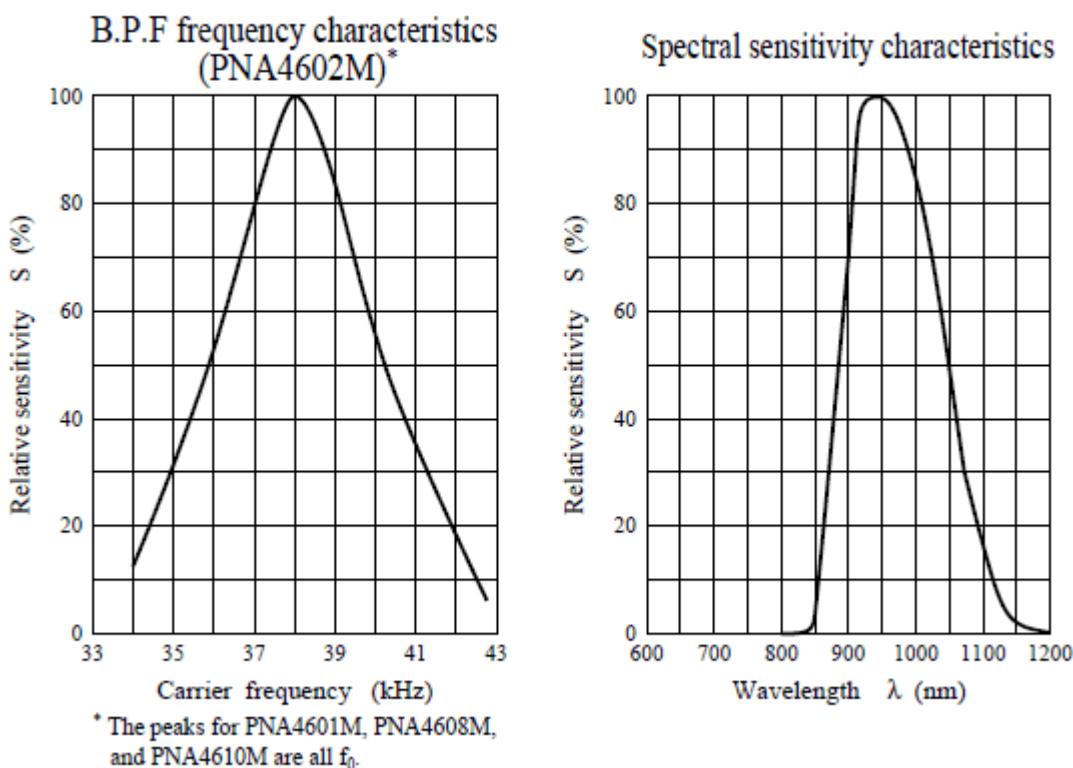
Hay unos pocos diferencia entre estos y dicen que un CdS fotocélulas:

Detectores IR están especialmente filtrar la luz infrarroja, que no son buenos en la detección

de la luz visible. Por otro lado, fotocélulas son buenos en la detección de la luz visible de color amarillo / verde, no es bueno en la luz IR

- Los detectores de IR tienen un demodulador el interior que busca para modulada IR a 38 KHz. Sólo brillante no se detectará un LED IR, que tiene que ser PWM parpadeando a 38KHz. Las fotocélulas no tienen ningún tipo de demodulador y pueden detectar cualquier frecuencia (incluyendo DC) dentro de la velocidad de respuesta de la célula fotoeléctrica (que es de aproximadamente 1 kHz)
- Detectores IR son digitales: - bien detectan la señal de 38Khz y la salida bajo (0V) o que no detectan ninguna salida y alta (5V). Fotocélulas actúan como resistores, el resistor cambia en función de la cantidad de luz que se exponen.

### **Lo que se puede medir**



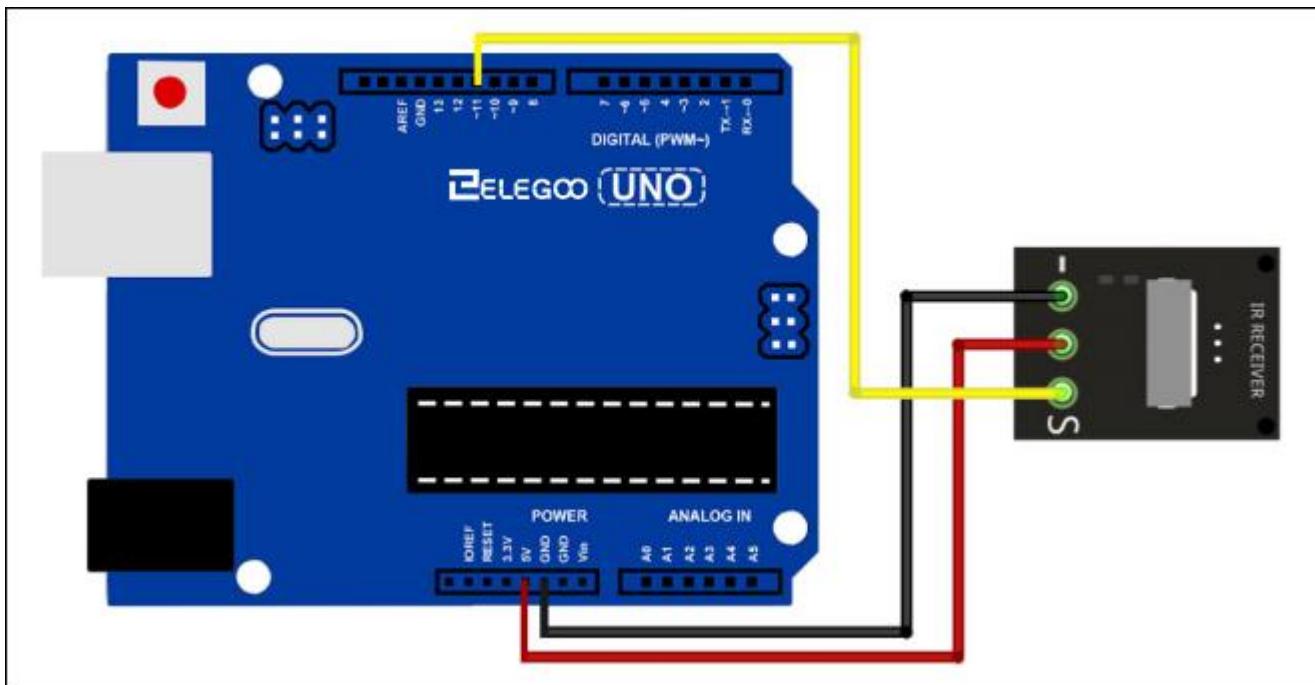
Como se puede ver en estos gráficos hoja de datos, la detección de la frecuencia se presenta a 38 KHz y el pico de color del LED es de 940 nm. Se puede usar de aproximadamente 35 KHz a 41 KHz pero la sensibilidad va a dejar para que no detectará así de lejos. Del mismo modo, puede utilizar 850 a 1100 nm LEDs pero no va a funcionar tan bien

como 900 a 1000 nm, así que asegúrese de obtener los LED coincidente! Compruebe la hoja de datos para el IR LED para verificar la longitud de onda.

Intenta conseguir 940 nm - recuerda que 940 nm no es la luz visible (su infrarroja roja)!

## Conexión

### Diagrama de cableado



Que hay 3 conexiones con el receptor IR.

Las conexiones son: Señal, Voltaje y Tierra

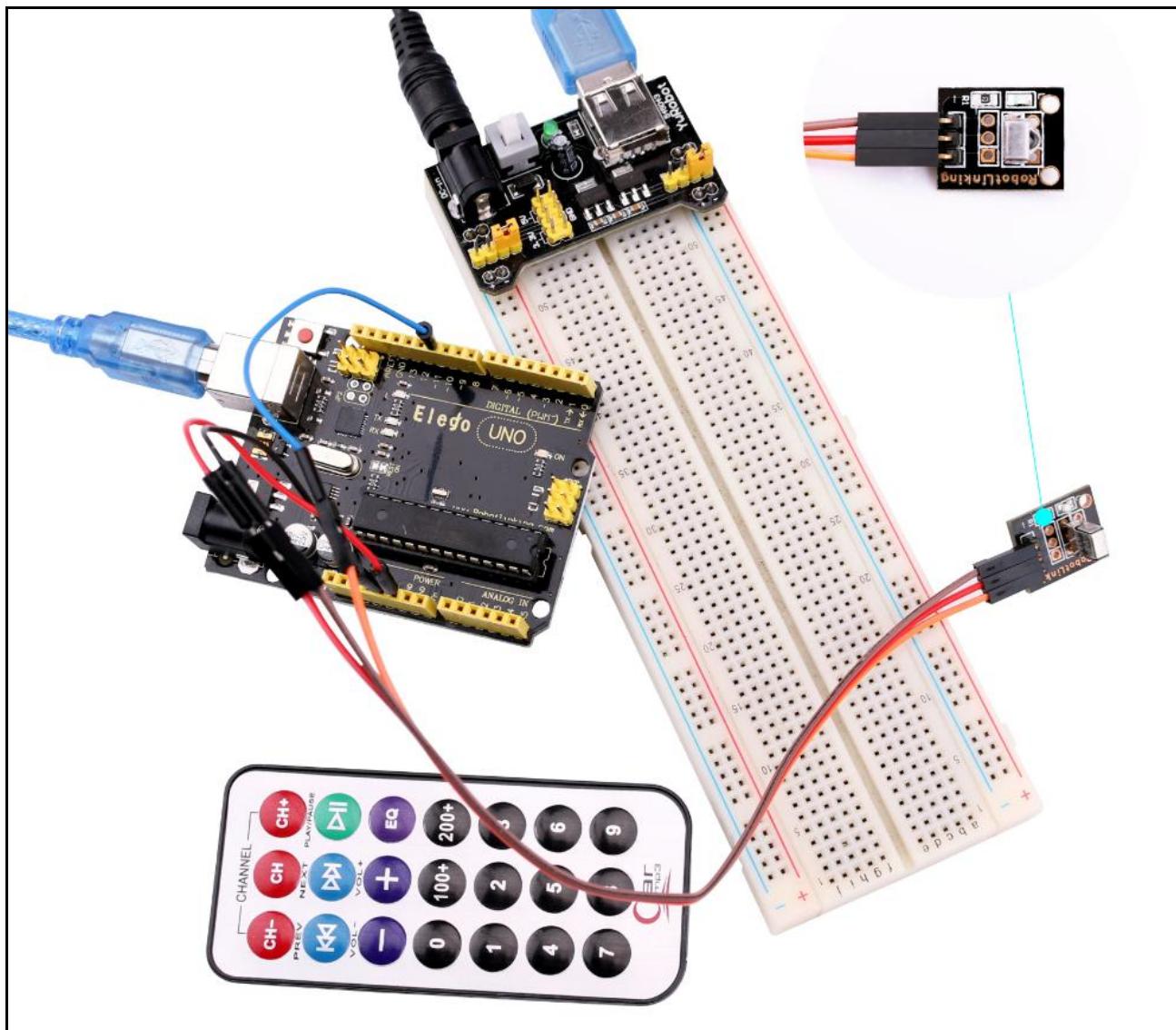
El "-" es la tierra, "S" es la señal, y el perno central es de Voltaje 5 V.

## Código

**Antes de poder ejecutar esto, asegúrese de que ha instalado la biblioteca de <remoto IR> o volver a instalarlo, si es necesario. De lo contrario, su código no funcionará.**

A continuación vamos a mover el <robot remoto IR> fuera de la carpeta de la biblioteca, lo hacemos porque que los conflictos de la biblioteca con la que va a utilizar. Usted puede simplemente arrastrar de nuevo dentro de la carpeta de la biblioteca, una vez que haya terminado la programación de su microcontrolador.

Una vez que haya instalado la biblioteca, sólo seguir adelante y reiniciar el software IDE.



# Lección 22: DC Motores

## Visión general

En esta lección, usted aprenderá cómo controlar un pequeño motor de corriente continua utilizando una UNO R3 y un transistor.

## Componentes Requeridos

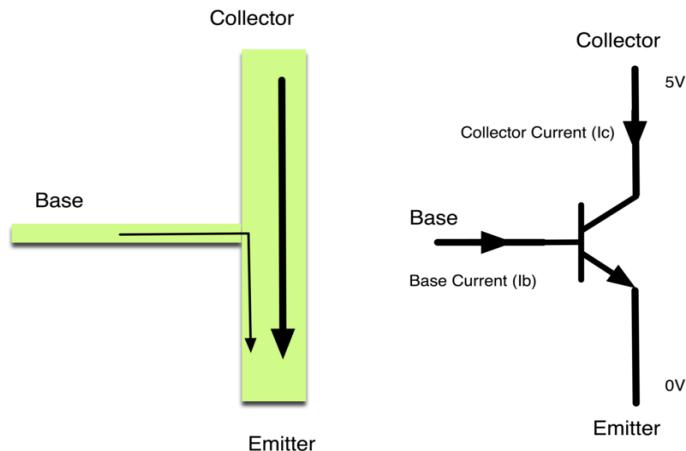
- (1) Elegoo UNO R3
- (1) Placa Universal
- (1) 6v DC motor
- (1) PN2222
- (1) 1N4007
- (1) 220 ohm resistor
- (3) M-M alambres

## Introducción de Componente

### TRANSISTORES:

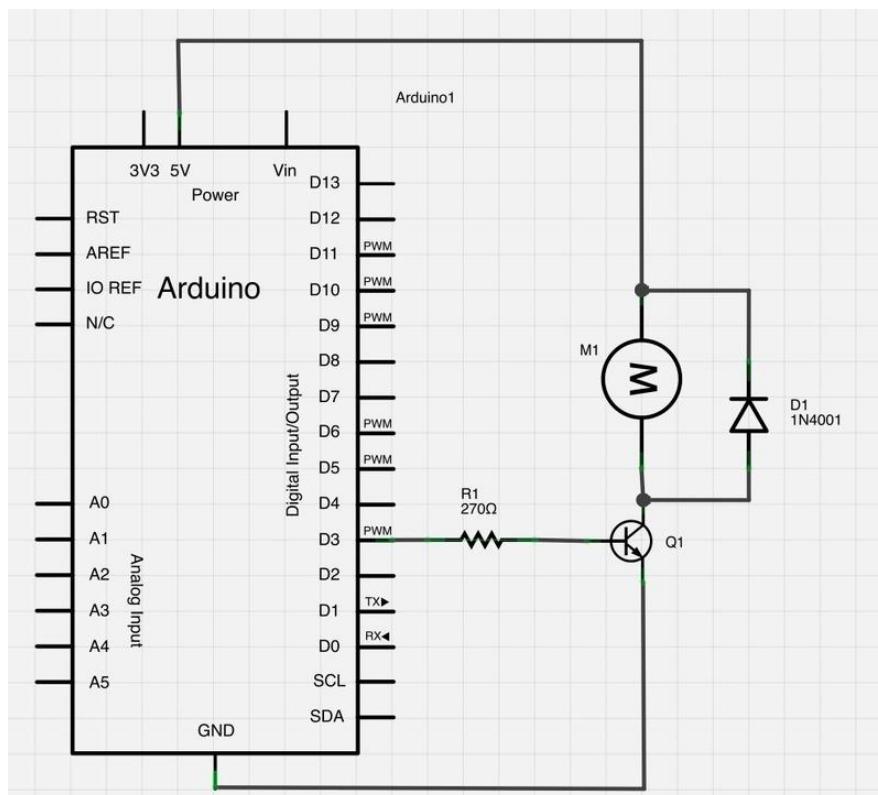
El pequeño motor es probable que utilice más energía que la placa de UNO R3 de salida digital puede manejar directamente . Si tratamos de conectar el motor directamente al perno de placa de UNO R3, hay una buena probabilidad de que podría dañar la placa de UNO R3.

Un pequeño transistor como el PN2222 se puede utilizar como un interruptor que utiliza sólo un poco corriente de salida digital de placa de UNO R3 para controlar la corriente mucho más grande del motor.



El transistor tiene tres terminales. La mayor parte de la electricidad que fluye desde el Colector al emisor, pero esto sólo sucederá si una pequeña cantidad está fluyendo en la conexión de base. Esta pequeña corriente es suministrada por la salida digital de placa de UNO R3.

A continuación se llama un diagrama esquemático. Como un diseño de placa, que es una forma de mostrar cómo las partes de un proyecto electrónico están conectados entre sí.



El perno D3 de la placa de UNO R3 está conectado al resistor. Al igual que cuando se utiliza un LED, esto limita la corriente que fluye en el transistor a través de la base.

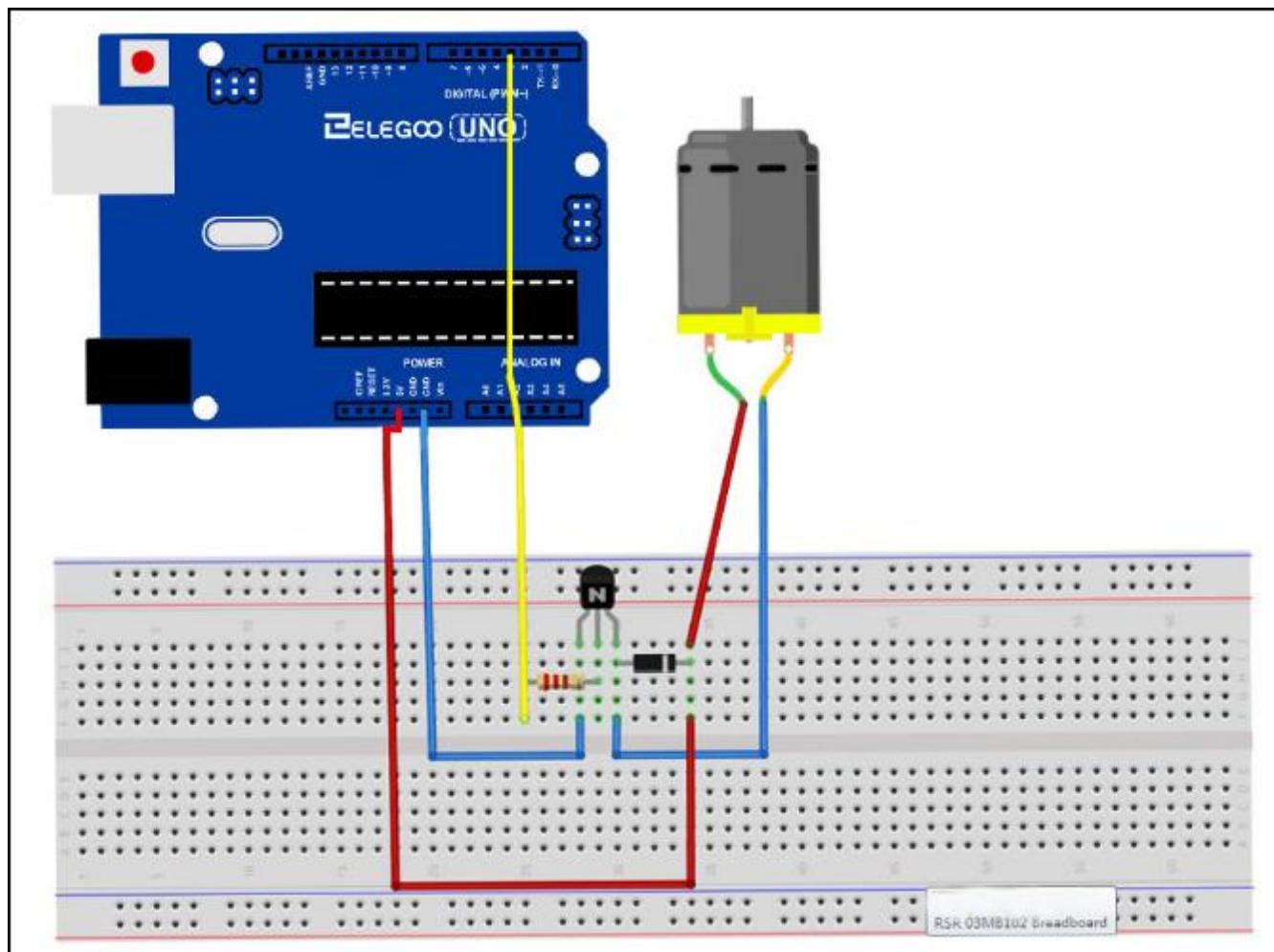
Hay un diodo conectado a través de las conexiones del motor. Diodos sólo permiten que la electricidad fluya en una dirección (la dirección de su flecha).

Cuando se desconecta la alimentación a un motor, se obtiene un pico negativo de la tensión, que puede dañar la placa de UNO R3 o el transistor. El diodo protege contra esto, por un cortocircuito en cualquier corriente inversa desde el motor.

A través de este experimento, que ha aprendido a conducir LCD1602. Ahora usted puede crear sus propios mensajes para mostrar! También puede tratar de dejar que sus números de visualizador de LCD1602.

## Conexión

### Diagrama de cableado



Que va a utilizar una salida de placa de UNO R3 analógica (PWM) para controlar la velocidad del motor mediante el envío de un número entre 0 y 255 del Monitor Serial.

Cuando juntas la placa universal, hay dos cosas a tener en cuenta.

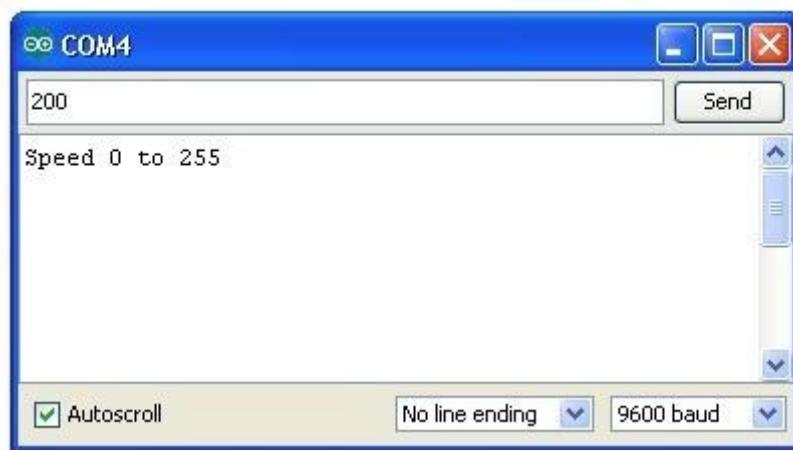
En primer lugar, asegúrese de que el lado plano del transistor está en el lado derecho de la placa universal.

En segundo lugar, el fin de rayas del diodo debe ser hacia la línea de potencia de + 5V - véase la imagen de abajo!

El motor que viene con los kits no extrae más de 250mA, pero si usted tiene un motor diferente, que fácilmente podría dibujar 1000mA, más de un puerto USB puede manejar! Si no está seguro de consumo de corriente de un motor, alimentar la placa de UNO R3 de un adaptador de pared, no sólo USB.

El transistor actúa como un interruptor, el control de la potencia al motor, perno 3 de placa de UNO R3 se utiliza para activar el transistor encendido y apagado y se le da el nombre de 'Perno de motor' en el boceto.

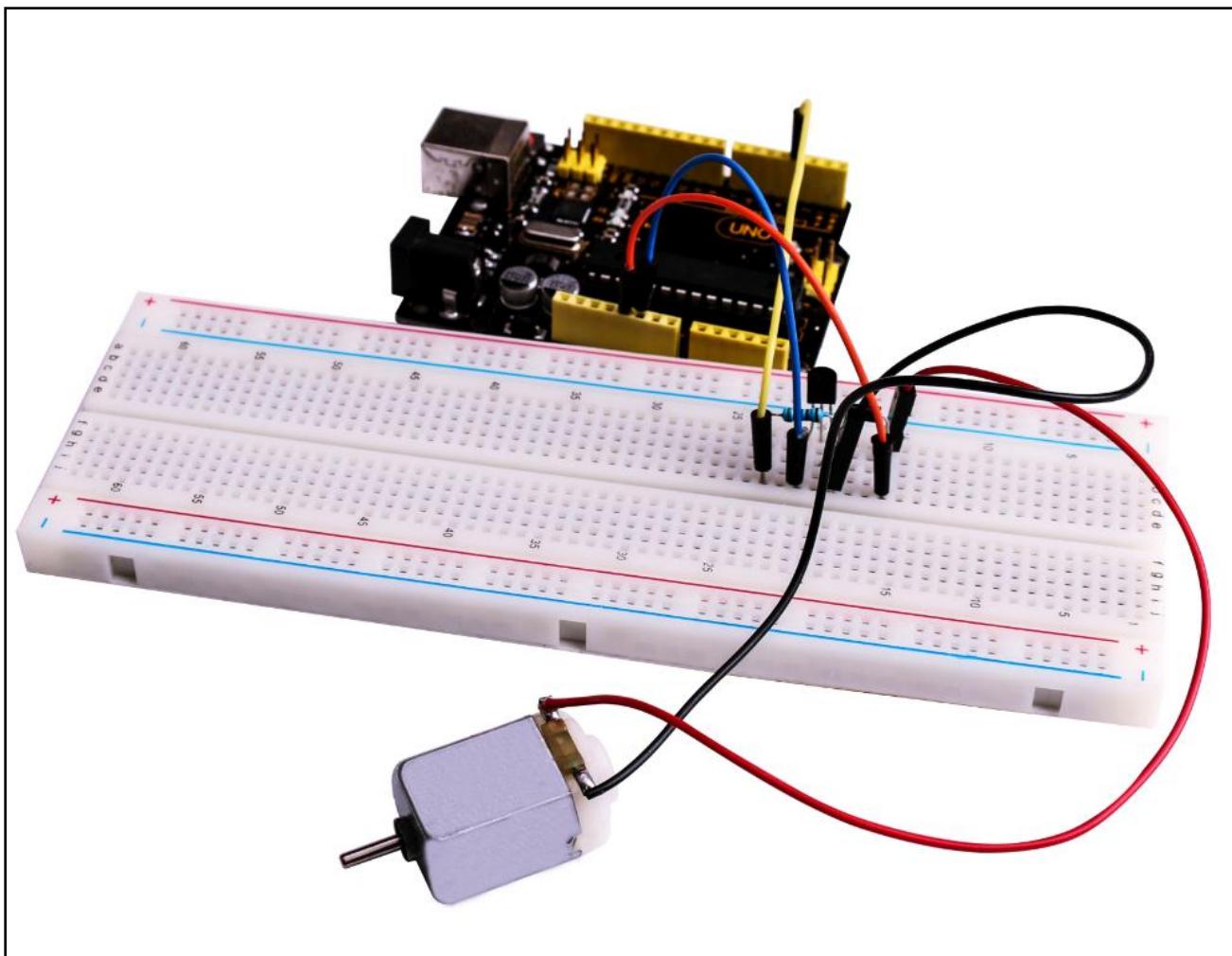
Cuando se inicia el boceto, se le solicita que le recuerde que para controlar la velocidad del motor, es necesario introducir un valor entre 0 y 255 en el monitor serial.



En la función de 'bucle', el comando 'Serial.parseInt' se utiliza para leer el número introducido como texto en el monitor serial y convertirla en un "int".

Se podría escribir cualquier número aquí, por lo que la sentencia "si" en la siguiente línea sólo hace una escritura analógica con este número si número se encuentra entre 0 y 255.

## Código



# Lección 23: Motor de Pasos

## Visión general

En esta lección, aprenderá una manera divertida y fácil de conducir un motor de pasos.

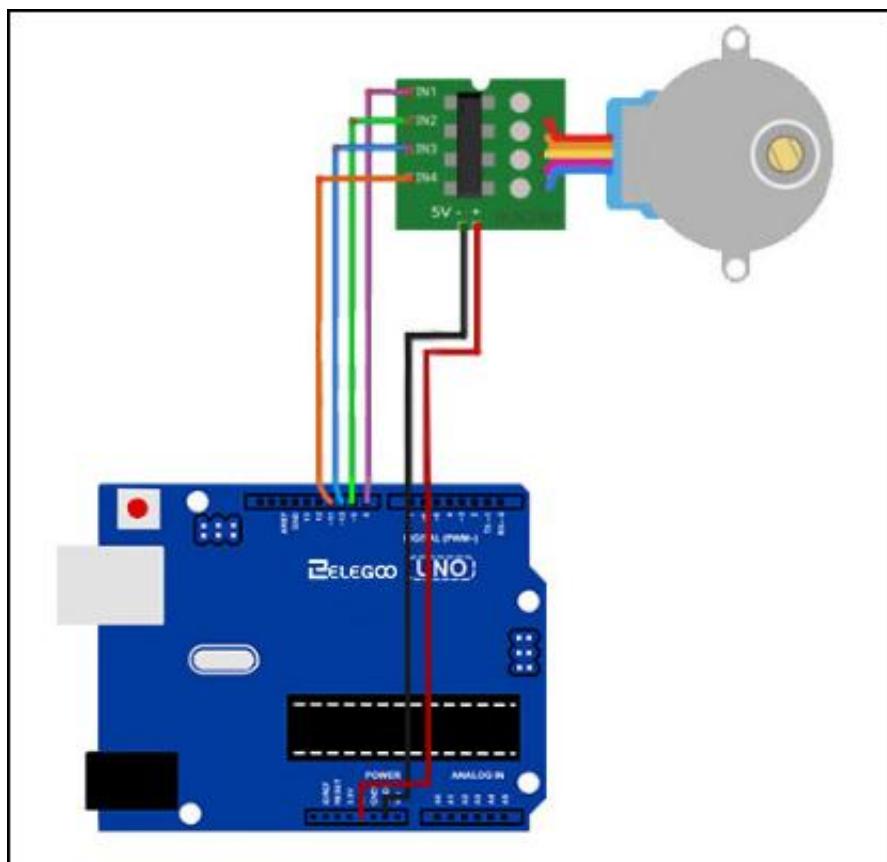
El motor de pasos esta usando viene con su propia placa de conductor por lo que es fácil de conectar a nuestra UNO.

## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) ULN2003 Módulo del controlador de motor de pasos
- (1) Motor de pasos
- (9) F-M alambres

## Conexión

### Diagrama de cableado



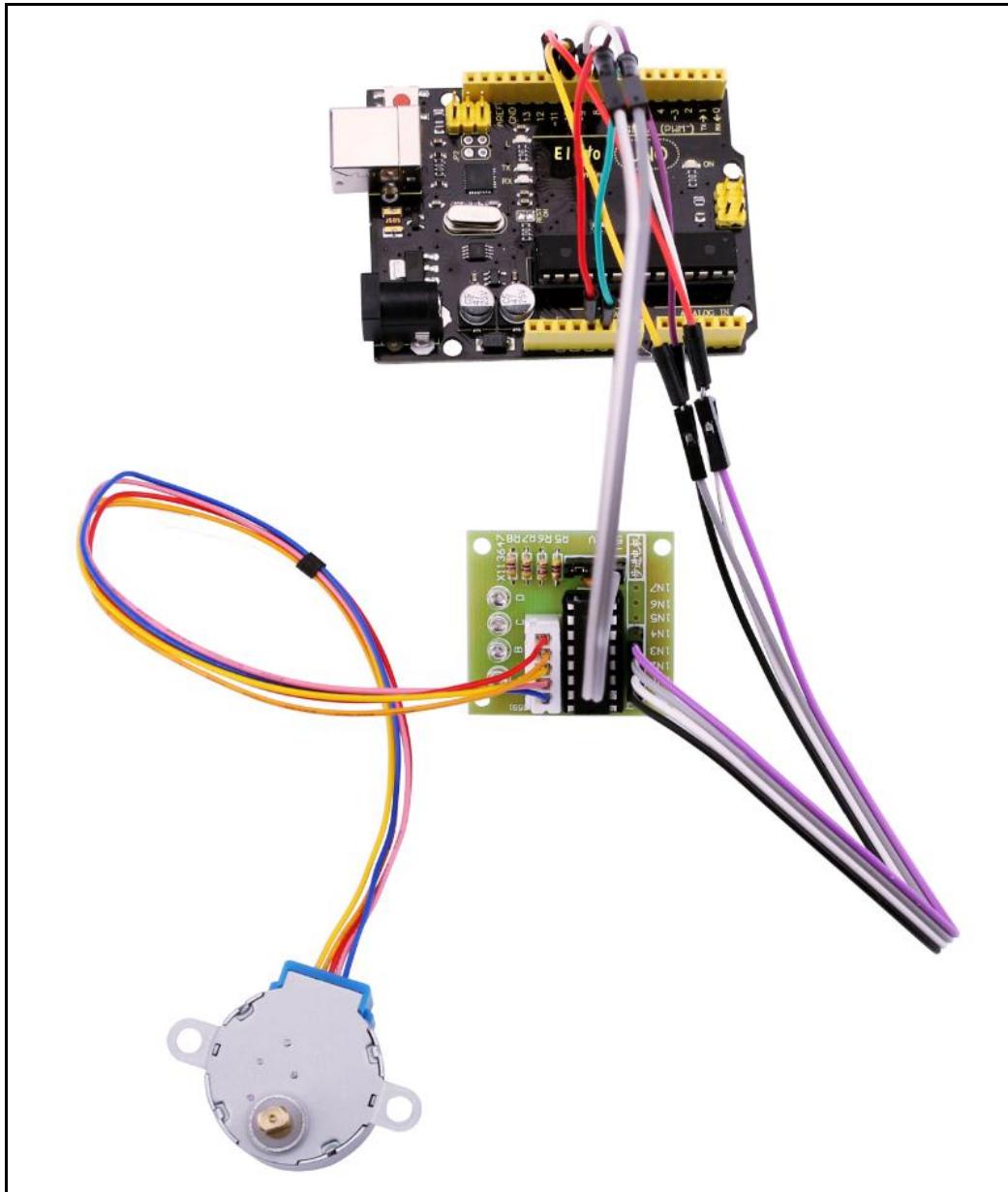
Estamos utilizando 4 pernos para controlar el motor de pasos.

Pernos 8-11 están controlando el motor de pasos.

Ponemos en contacto a 5V y Tierra de UNO para el motor de pasos.

## Código

**Antes de poder ejecutar esto, asegúrese de que ha instalado la biblioteca de <Control de Motor de Pasos> y volver a instalarlo, si es necesario. De lo contrario, su código no funcionará.**



# Lección 24: Motor de Pasos Controlando con Remoto

## Visión general

En esta lección, aprenderá una manera divertida y fácil de controlar un motor de pasos desde una distancia utilizando un control remoto IR.

El motor de pasos estamos usando viene con su propia placa de conductor por lo que es fácil de conectar a nuestra UNO.

Dado que no queremos para accionar el motor directamente de UNO, vamos a utilizar una fuente de alimentación de pequeña placa universal de bajo costo que se conecta directamente a nuestra placa y la potencia con una fuente de alimentación de 9V 1Amp.

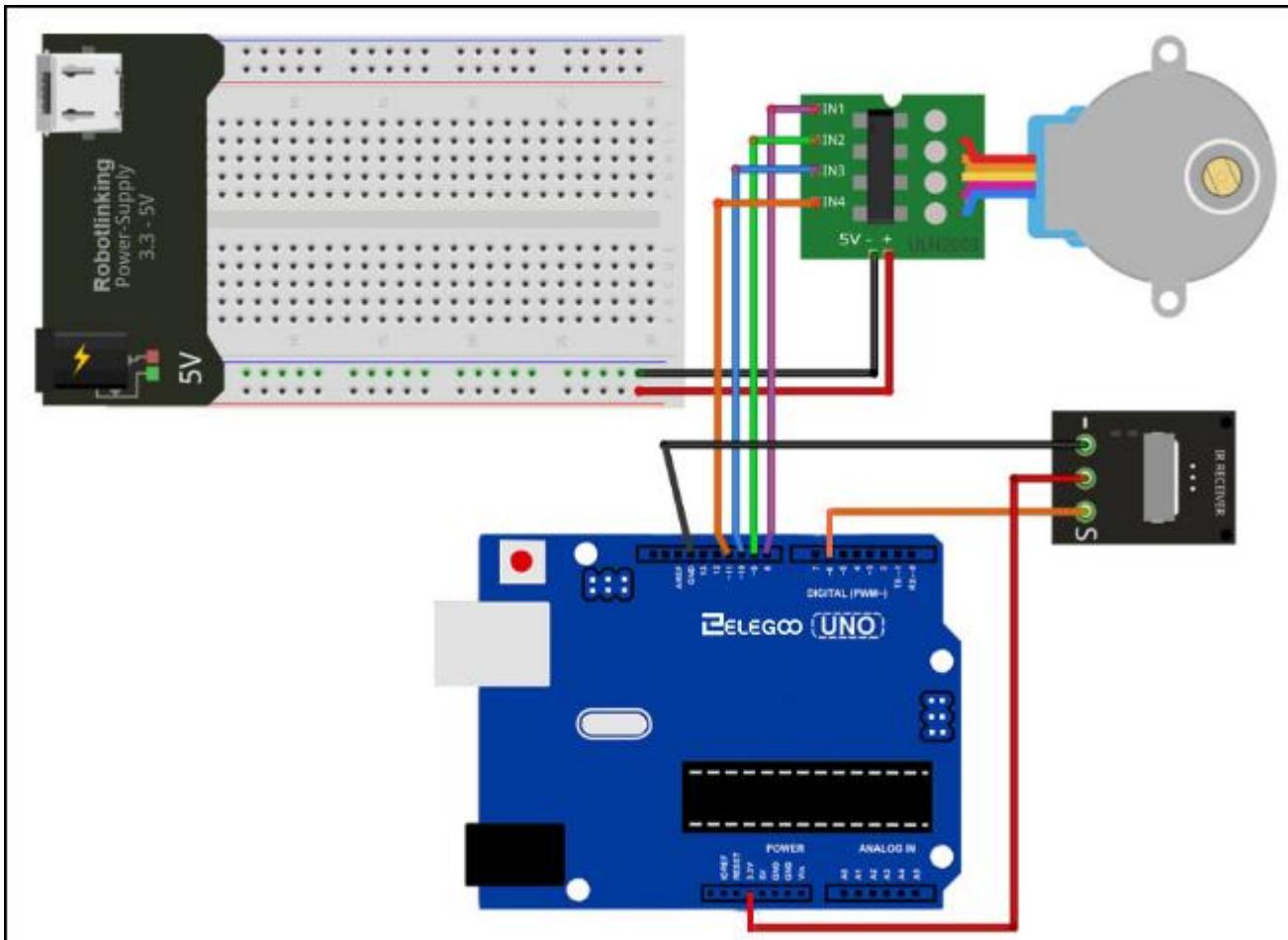
El sensor de IR está conectado a la UNO directamente ya que utiliza casi ningún poder.

## Componentes Requeridos

- (1) Elegoo UNO R3
- (1) Placa Universal
- (1) Módulo del receptor IR
- (1) Remoto IR
- (1) ULN2003 Módulo del controlador de motor de pasos
- (1) Motor de pasos
- (1) Módulo de fuente de alimentación
- (1) Adaptador
- (9) F-M alambres

## Conexión

### Diagrama de cableado



Estamos utilizando 4 pernos para controlar el motor de pasos y 1 perno para el sensor de infrarrojos.

Pernos 8-11 están controlando el motor de pasos y el perno 6 está recibiendo la información IR.

Ponemos en contacto a 5V y Tierra de UNO para el sensor. Como medida de precaución, utilizar una fuente de alimentación de placa universal para alimentar el motor de pasos, ya que puede usar más energía y no queremos dañar la fuente de alimentación de UNO.

## Código

**Antes de poder ejecutar esto, asegúrese de que ha instalado el <IR remoto> biblioteca de <Control de Motor de Pasos > o volver a instalarlo, si es necesario. De lo contrario, su código no funcionará.**

