

# Práctica 2.3: ¿Cuánta luz hay aquí?

Ha llegado el momento de que veamos un nuevo sensor, el cual resulta interesante tanto por la facilidad de su uso como por su reducido coste. Existen diferentes versiones, como podemos ver en [este apartado](#), ya debería sonarte porque lo vimos hace solo unas páginas. Para esta práctica, vamos a utilizar la versión sencilla, directamente la fotorresistencia, ya que al estar trabajando alrededor de la idea de construir un vestible, algo que podamos llevar puesto, es necesario que nuestros proyectos sean lo menos aparatosos posible. Bueno, o no, dependerá de qué tipo de proyecto queramos hacer...

## Antes de empezar, ¿qué es un divisor de voltaje?

Ya hemos visto que para trabajar con Arduino, son necesarias una serie de estructuras de programación. Aparte de eso, a veces vamos a necesitar ciertos componentes que permitan leer los valores de nuestros sensores.

La mayor parte de los sensores que utilizamos con Arduino son plug-and-play, es decir, no necesitan nada extra para funcionar, pero, en ocasiones, necesitaremos alguna cosilla extra. Ese es el caso de nuestra fotorresistencia. Para poder leer esa variación en la fotorresistencia que se produce cuando aumenta o disminuye la luz, necesitaremos conectar una **resistencia extra** que funcionará como **divisor de voltaje**. Esta resistencia también podría ser un potenciómetro, o lo que es lo mismo, una resistencia con un valor variable que podemos ajustar.

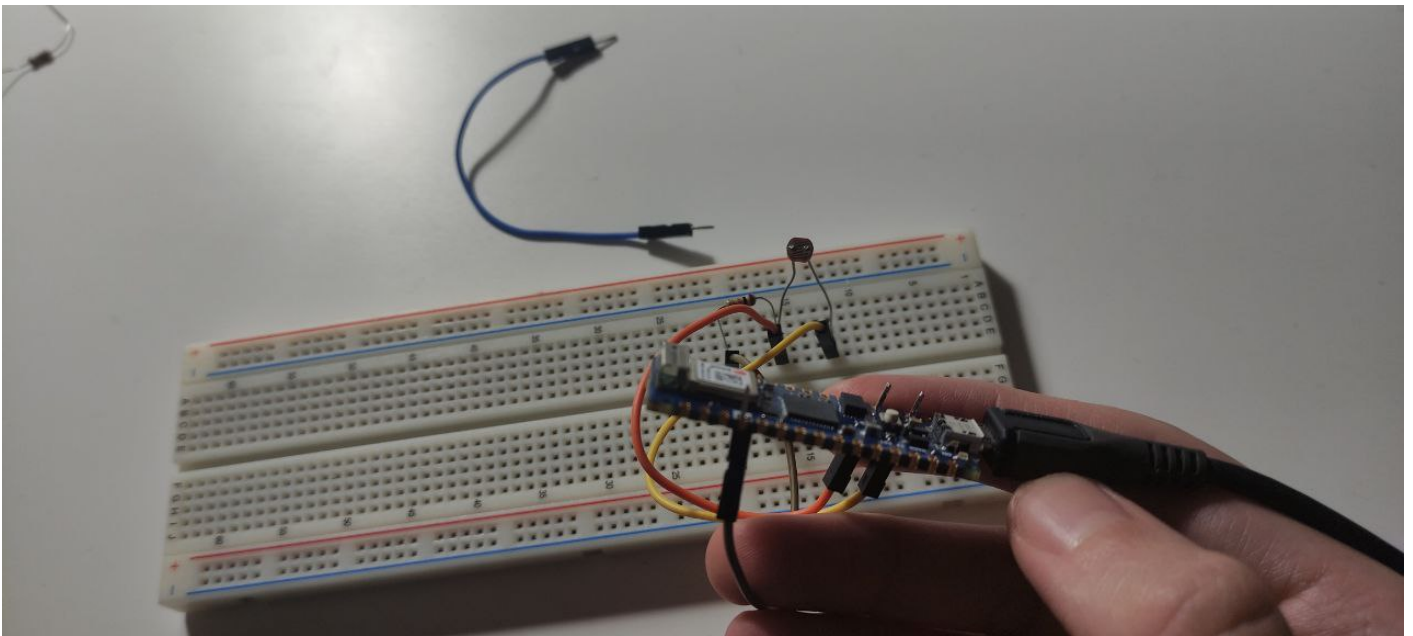
Con un divisor de voltaje, seremos capaces de obtener a partir de la variación en la resistencia, una variación en el voltaje que será lo que lea nuestro Arduino a través del pin que configuremos como entrada.

Para saber más sobre divisores de voltaje puedes visitar [esta página](#).

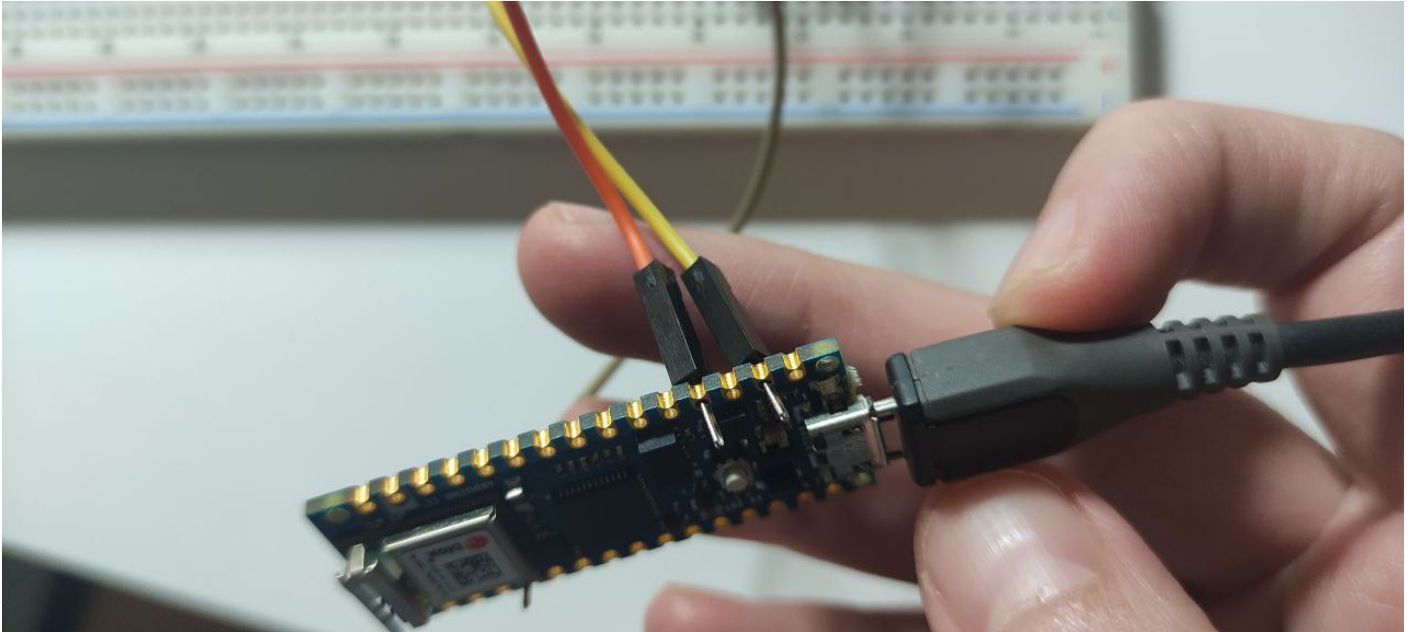
## Materiales

Para realizar esta práctica necesitaremos:

1. Nuestro **Arduino Nano 33 IoT**
2. **3 Cables macho-macho**. ¿Por qué se llaman así? pues porque ambos extremos se insertan en alguna parte. Existen también cables macho-hembra y hembra-hembra, los cuales son útiles dependiendo del tipo de pines que tengan tanto el microcontrolador como los sensores y actuadores que usemos.
3. 1 resistencia de **10KOhm** que funcionará como [divisor de voltaje](#).
4. **Cable microUSB** para conectar nuestro Arduino.
5. La **fotorresistencia**.

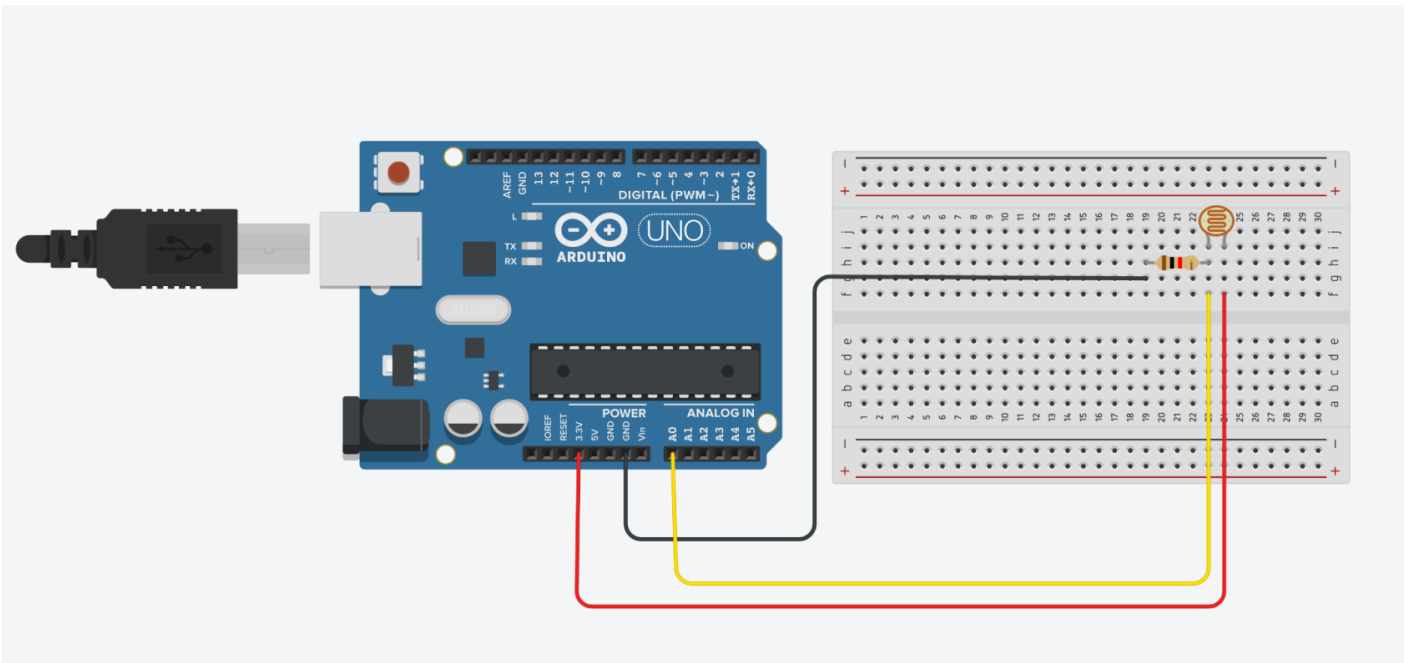


En la imagen superior podemos ver la colocación de los cables macho en nuestro Arduino. Realmente esta no es la manera más correcta de realizar las conexiones. Lo correcto sería soldar unos pines en nuestro Arduino, pero como necesitaremos los agujeritos para coserlos despues, introduciremos los cables como se ve en la siguiente imagen:



## Estableciendo la conexión

Las conexiones que tenemos que realizar son las siguientes:



Para realizar este circuito se ha utilizado [Tinkercad](#), una aplicación online nos permite diseñar circuitos de una manera visual. También podríamos haberlo hecho usando Fritzing, otra aplicación de código abierto que también tiene versión de escritorio. En este caso la versión de escritorio requiere de un pago/donación (8€), si bien una versión anterior está

disponible para los equipos Vitalinux a través de **Vitalinux Play**.

Ya te habrás dado cuenta de que el Arduino que aparece en la imagen no es el NANO 33, pero los pines a los que has de conectar los cables son los mismos. Ahí vemos que estamos empleando 3 pines:

**A0:** va desde ahí hasta una pata de nuestra **resistencia**, la cual está conectada a una pata de la **fotorresistencia**.

**3.3V:** va desde ahí hasta la pata de la **fotorresistencia que no está conectada** a la resistencia.

**GND:** va desde ahí hasta la pata de la **resistencia que no está conectada** a nada.

Tenemos que tener cuidado con varios aspectos a la hora de realizar las conexiones en nuestra protoboard. El más importante es que todos esos puntitos están conectados unos con otros en línea. Es decir, en la posición en la que está colocada la protoboard de la imagen superior, los puntos están conectados en líneas verticales (**pistas**), de arriba a abajo, pero no de forma horizontal. Eso es así para las líneas centrales pero no para las líneas roja y negra señalizadas con el símbolo + y - respectivamente (**buses**), las cuales están conectadas de izquierda a derecha, si mantenemos la posición de la protoboard de la imagen de arriba.

Por tanto, vemos que nuestro cable negro está conectado a uno de los extremos de nuestra resistencia, mientras que el otro extremo se conecta a un extremo de la fotorresistencia y al pin A0 de nuestro Arduino. Para más información sobre las conexiones en pistas y buses en una protoboard consultar [aquí](#).

Una vez hemos terminado nuestro circuito va a llegar el momento de ver el código que necesitaremos para que funcione:

```
/*  
 * Created by ArduinoGetStarted.com  
 *  
 * This example code is in the public domain  
 *  
 * Tutorial page: https://arduinogetstarted.com/tutorials/arduino-light-sensor  
 * Modificado por Marta Pérez  
 */
```

```
void setup() {  
  // comienza la comunicación del puerto serie en 9600 bits por segundo:  
  Serial.begin(9600);  
}  
  
void loop() {  
  // Lee el valor de A0 (entre 0 y 1023)  
  int analogValue = analogRead(A0);  
  
  Serial.print("Lectura analógica: ");  
  Serial.print(analogValue);  // lectura del valor analógico obtenido  
  
  // Definiremos los umbrales dependiendo de la cantidad de luz recibida  
  if (analogValue < 10) {  
    Serial.println(" - Oscuro");  
  } else if (analogValue < 200) {  
    Serial.println(" - Tenue");  
  } else if (analogValue < 500) {  
    Serial.println(" - Iluminado");  
  } else if (analogValue < 800) {  
    Serial.println(" - Bastante iluminado");  
  } else {  
    Serial.println(" - Mucha luz");  
  }  
  
  delay(500);  
}
```

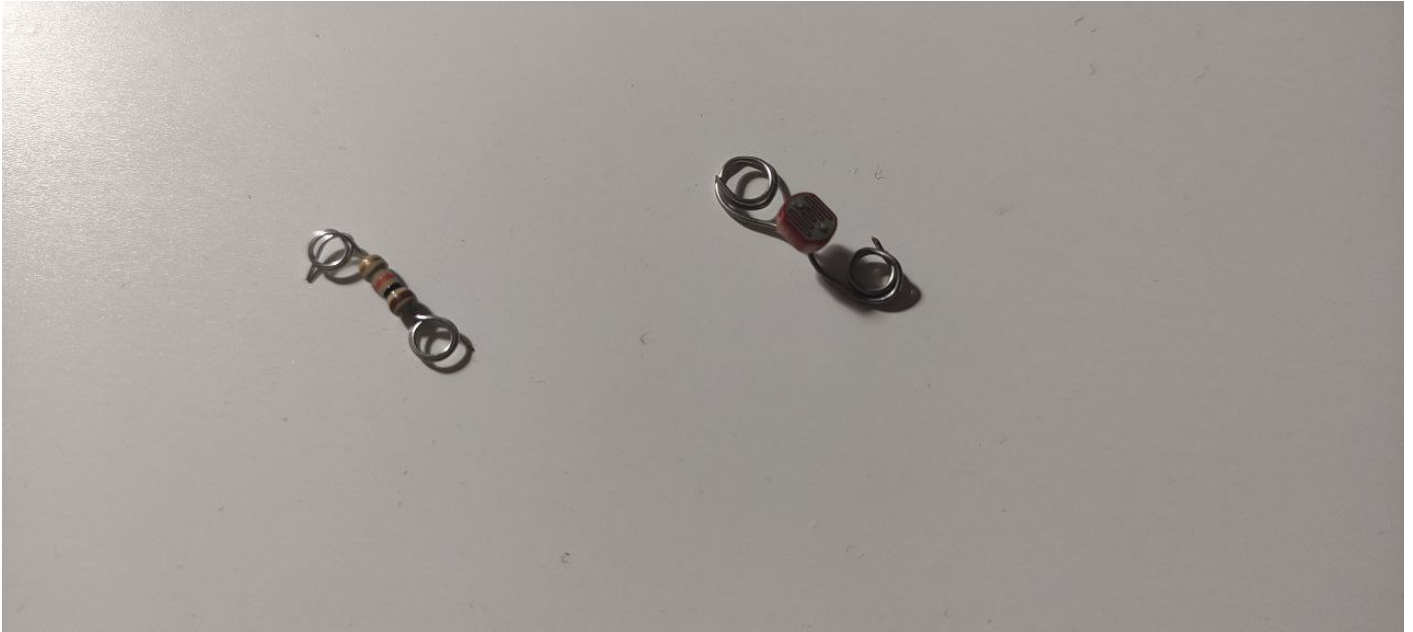
En la práctica anterior se ha explicado línea a línea. En este caso al ser un código compuesto por condicionales en su mayoría, no se ha considerado necesario explicar este código línea a línea.

El resultado que obtenemos con el código superior es:

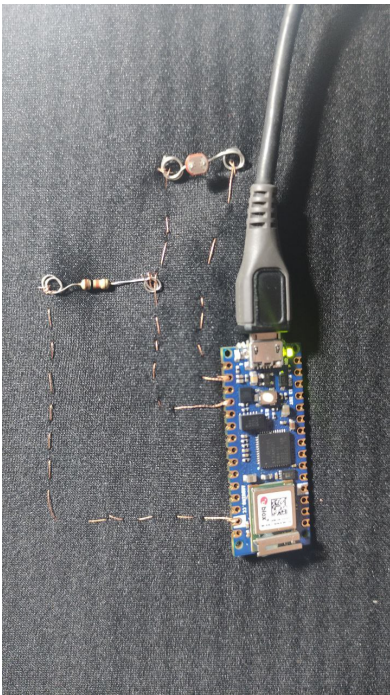
[https://player.vimeo.com/video/749590494?h=fb795813efbadge=0&autoplay=0&player\\_id=0&pp\\_id=58479](https://player.vimeo.com/video/749590494?h=fb795813efbadge=0&autoplay=0&player_id=0&pp_id=58479)

## De la protoboard a la tela

El próximo paso es trasladar nuestro circuito desde la protoboard a la tela. Para ello será necesario modificar levemente tanto la resistencia como nuestro fotorresistencia, para que adquieran el siguiente aspecto:



Para modificarlos, una manera sencilla de hacerlo es enrollar los extremos alrededor de la punta de un lápiz. Una vez los tengamos así, podemos comenzar a coserlos a nuestro trozo de tela, para obtener algo parecido a esto.



## Ciertas cosas a tener en cuenta...

**1.** Cuando cosemos circuitos, es necesario que tengamos en cuenta que **la corriente no se transmite igual de bien que si estuviésemos usando un cable**, por lo que si el hilo que

empleamos es demasiado fino, puede que sea necesario que realicemos varias costuras en lugar de una, para hacer que el hilo sea más grueso y se transmita mejor la electricidad. Tenemos que tener en cuenta que, con el tiempo, la conductividad del hilo se va perdiendo al estar expuesto a la intemperie, no como un cable que está protegido por plástico.

2. Es muy importante que cortemos **todos los hilos largos** que se crean al hacer los nudos, ya que si, por ejemplo se tocan el hilo que va **GND** y el que va a **3.3V** tendremos un cortocircuito que puede, a parte de hacer que salga algo de humo, llegar a romper nuestro Arduino.

Un buen sustituto para realizar conexiones en nuestros wearables que se encuentra a medio camino entre los cables y el hilo conductor es el **hilo de cobre esmaltado**. Se trata de unos filamentos de cobre que pueden llegar a ser bastante finos, aunque no tanto como el hilo, los cuales se encuentran cubiertos de una capa aislante. Estos pueden usarse a modo de hilo si tenemos una aguja con un ojal lo suficientemente grande, o si no, podemos coserlos con hilo no conductor a la tela. Al ser más finos que los cables, es más fácil disimularlos en un vestible.

## ¿Qué tenemos que presentar?

En este caso será suficiente con que me envíes un **gif** o **enlace al video** con el circuito hecho en la protoboard en el que se vea también la medición del sensor en el puerto serie, similar al video que aparece al final del apartado "Estableciendo la conexión". Sigue las instrucciones descritas en la tarea moodle del curso.

No es obligatorio coser el circuito, pero si lo haces, puedes enviármelo o comentarme cualquier duda o dificultad que te haya surgido.

FUENTES:

Arduino y sensor lumínico: <https://arduinogetstarted.com/tutorials/arduino-light-sensor>

Divisor de voltaje: <https://mielelectronicafacil.com/analisis-de-circuitos/divisor-de-voltaje/#potenciometros>

Financiado por el Ministerio de Educación y Formación Profesional y por la Unión Europea - NextGenerationEU





---

Revision #24

Created 2022-07-05 17:43:38 CEST by Marta P. Campos

Updated 2023-01-17 16:08:01 CET by Equipo CATEDU