

## Práctica 5 - Luces que se encienden dependiendo de la oscuridad

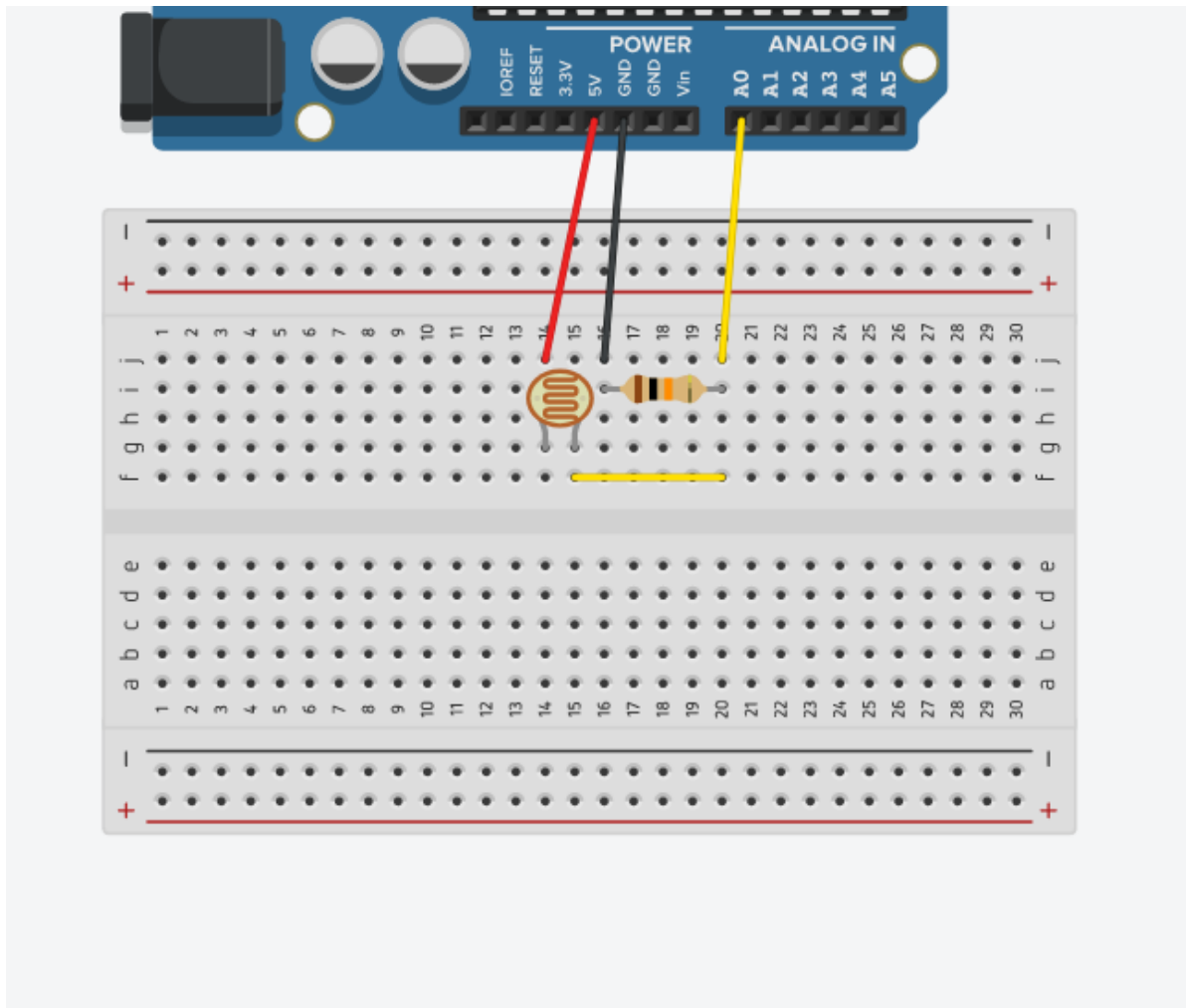
Link del proyecto: [https://www.tinkercad.com/things/jRzB2d5N6Sc-luces-en-la-oscuridad/editel?sharecode=-mUZphg\\_VUkH7-q-ZJBWARL5DJsGJL3Wgn\\_xmveRTc](https://www.tinkercad.com/things/jRzB2d5N6Sc-luces-en-la-oscuridad/editel?sharecode=-mUZphg_VUkH7-q-ZJBWARL5DJsGJL3Wgn_xmveRTc)

Este proyecto de arduino consiste en 6 luces LEDs que se van a encender más o menos dependiendo de la iluminación de un fotorresistencia.

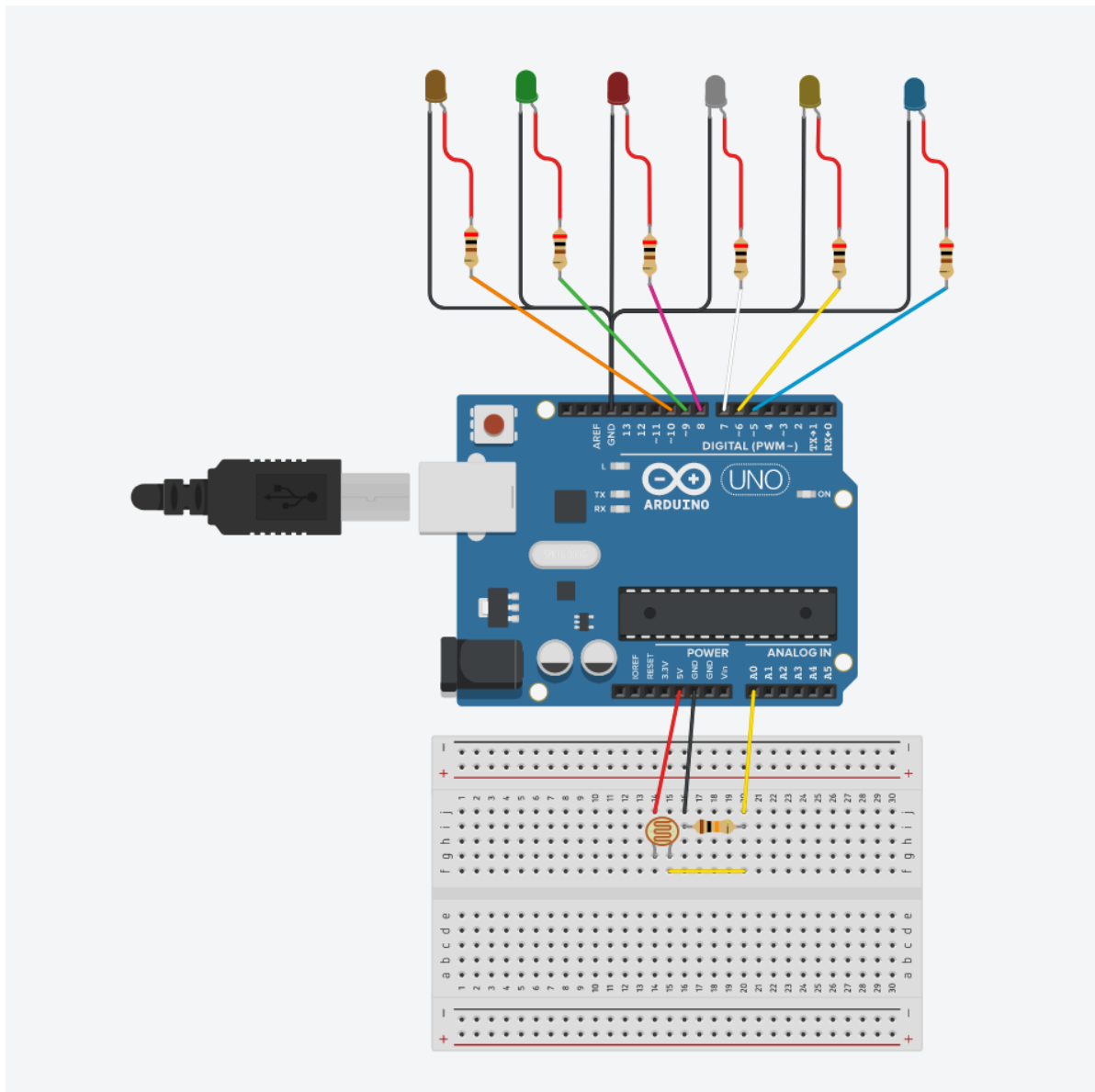
Vamos a dividir la realización del proyecto en dos etapas:

### 1. Montaje del circuito

Primero montamos el fotorresistencia, usando para ello una resistencia de  $10k\Omega$  y un pin analógica de la placa de arduino. Ayudándome de una placa de pruebas pequeña la conexión queda tal que así:



Ahora vamos a usar 6 LEDs de diferentes colores por mera estética, cada uno con una resistencia de  $200\Omega$ . Además, conectaremos cada resistencia a un pin del 5 al 10 para posteriormente controlar la corriente en cada una. Al final todo queda conectado tal que:



## 2. Programación de los componentes

Mi programa se basa en leer la iluminación de la fotorresistencia y dependiendo de esta mandar a que se encienda un número de LEDs.

Primero empezamos declarando las variables que van a necesitar, estas son:

```
const long A = 1000;    //Resistencia en oscuridad en KΩ
const int B = 15;       //Resistencia a la luz (10 Lux) en KΩ
const int Rc = 10;      //Resistencia calibracion en KΩ
const int LDRPin = A0;  //Pin del LDR

int V;
int ilum;
```

Ahora a priori lo siguiente que voy a necesitar van a ser una función que encienda los LEDs y otra que los apague, estas son:

```
/**
 * @brief Funcion que apaga todas las luces LEDs que esten encendidas
 */
void clear() {
    for (int i = 5; i < 11; i++) {
```

```

        digitalWrite(i, LOW);
    }
}
/**
 * @brief Funcion que enciende un numero de LEDs
 * @param n Numero de LEDs que queremos encender
 */
void enciendeLED(int n) {
    clear();
    for (int i = 5; i < n+5; i++) {
        digitalWrite(i, HIGH);
    }
}

```

Ahora una vez hecho esto paso al cuerpo del programa. Primero inicializo todos los pin de los LEDs a `OUTPUT` y los apago inicialmente en la función `setup()`:

```

void setup() {
    pinMode(8, OUTPUT);
    pinMode(9, OUTPUT);
    pinMode(10, OUTPUT);
    pinMode(5, OUTPUT);
    pinMode(6, OUTPUT);
    pinMode(7, OUTPUT);
    clear();
}

```

Ahora creo la función que se va a estar ejecutando constantemente, `loop`. Esta función es en la que leemos el valor de iluminación y con unas cuantas estructuras condicionales llamamos a encender más o menos LEDs:

```

void loop() {
    V = analogRead(LDRPin);

    ilum = ((long)V*A*10)/((long)B*Rc*(1024-V));    //calculamos la iluminacion

    if (0 <= ilum && ilum <= 200) {
        enciendeLED(6);
    }
    if (200 < ilum && ilum <= 400) {
        enciendeLED(5);
    }
    if (400 < ilum && ilum <= 600) {
        enciendeLED(4);
    }
    if (600 < ilum && ilum <= 800) {
        enciendeLED(3);
    }
    if (800 < ilum && ilum <= 1000) {
        enciendeLED(2);
    }
    if (1000 < ilum && ilum <= 1200) {
        enciendeLED(1);
    }
    if (1200 < ilum) {
        clear();
    }
}

```

```
}  
}
```

Con esto mi proyecto queda acabado, aunque claro, para sacarle el mayor provecho en la vida real recomiendo utilizar todo los LEDs blancos.

