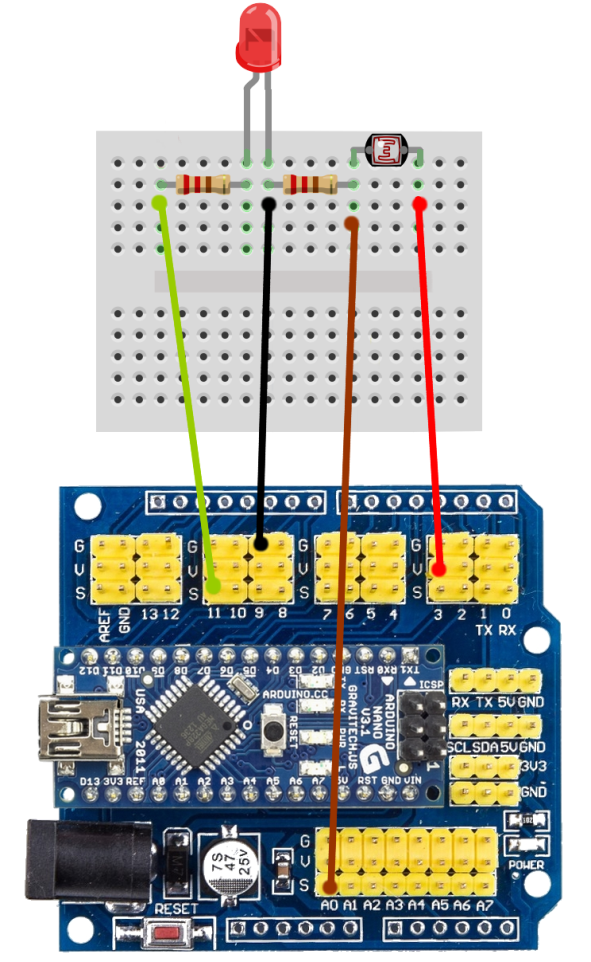
**Ejercicio 4: LDR-Puls**

**Uso de una fotoresistencia como sensor de luz**

Este es un ejercicio muy sencillo en el que veremos cómo obtener lecturas analógicas de un sensor y darles salida por el monitor del puerto serie para verlas.

Necesitaremos una placa Arduino NANO y una resistencia sensible a la luz o fotoresistencia (LDR) que nos servirá como interruptor.

También utilizaremos un led para que nos de una salida (ON OFF)

Circuito:

**1ªParte: Lectura del valor analógico de laLDR**

Comenzamos creando las variables que contendrán los números de los pines:

**int LDRPin = A0;** // Pin para la fotorresistencia

En el **setup()** las configuraremos como entradas o como salidas, según corresponda e inicializaremos el puerto Serie:

**void setup()  
{  
Serial.begin(9600);    // Inicia la comunicacion serie a 9600 baudios**

**pinMode(LDRPin, INPUT);  
}**

Ahora vamos al loop()**.** Primero leemos el valor que nos da la fotorresistencia y lo mostramos por el puerto serie

**void loop()  
{**

**Serial.println(analogRead(LDRPin));  
  
delay(100);  
}**

Tapando la LDR con la mano veremos la variación del valor analógico. Tomaremos como referencia un valor (**umbral**) en que creamos que debería apagarse nuestro led.

**2ª Parte: Encendido y apagado del LED con LDR**

Crearemos una variable **umbral**. Seguido utilizamos ese **umbral** para encender y apagar el Led en función de la cantidad de luz que recibe la LDR.

**int LEDPin = 13;** // Asigna LEDPin al pin 13

**int LDRPin = A0;** // Asigna LDRPin al pin A0

**int umbral = 100;**//este valor lo obtenemos poniendo la mano sobre la LDR

**void setup()**

**{**

**pinMode(LEDPin, OUTPUT); //** Establece un pin digital como salida.

**pinMode(LDRPin, INPUT); //** Establece un pin analógico como entrada.

**}**

**void loop()**

**{**

**if (analogRead(LDRPin)< umbral) {** // compara el valor de la LDR con en que hemos asignado en la variable “umbral”

**digitalWrite(LEDPin, HIGH);**

**}**

**else {**

**digitalWrite(LEDPin, LOW);**

**}**

**}**