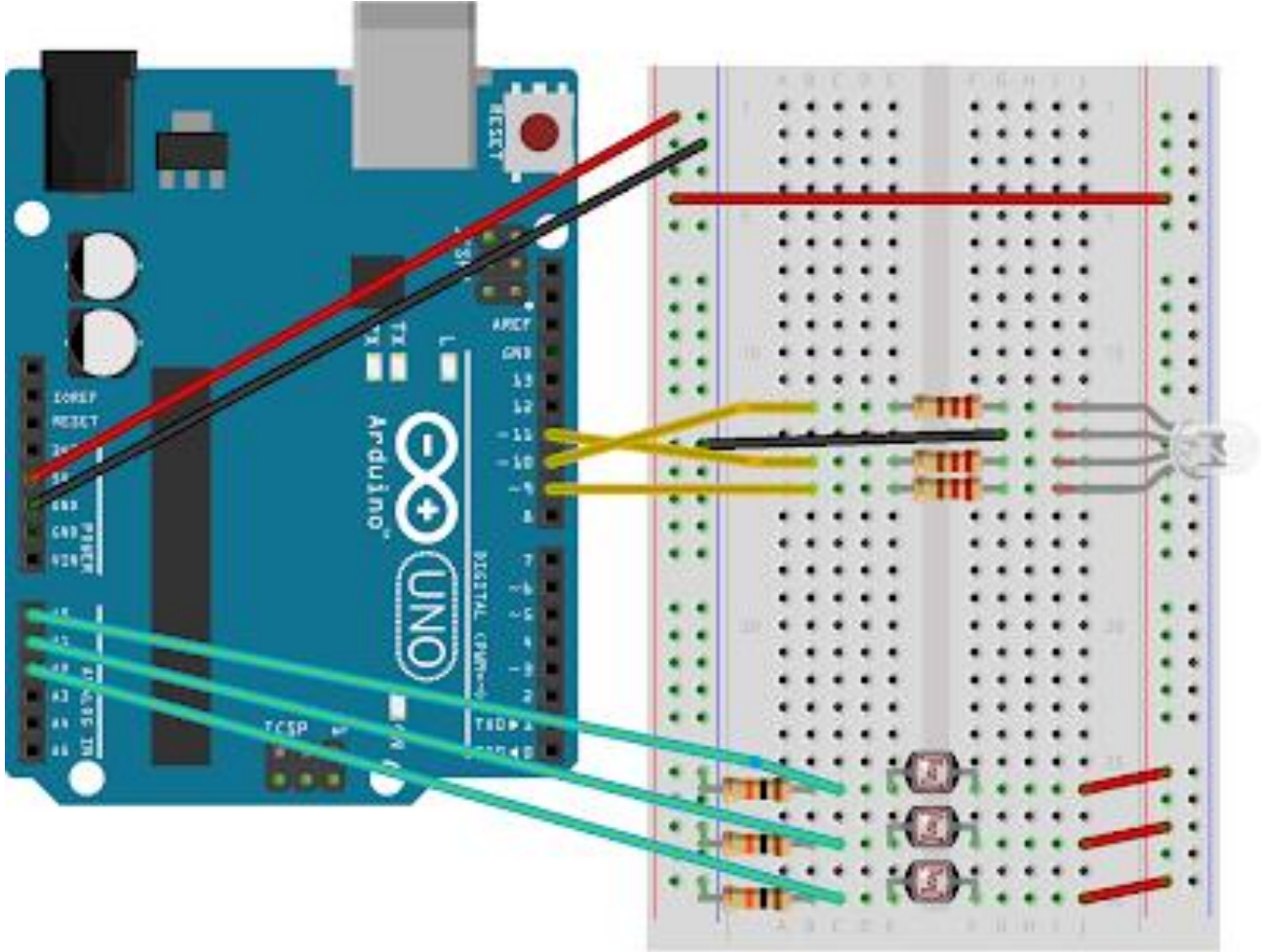


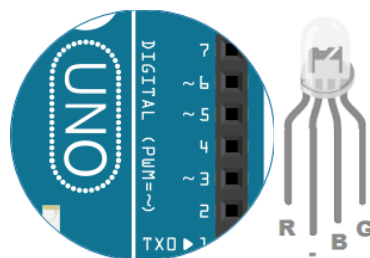
Pràctica-5: Llum-2

CREACIÓ D'UNA LÀMPADA QUE CANVIA SUAUMENT DE COLOR DEPENENT DE LES CONDICIONS D'IL·LUMINACIÓ EXTERNA



L'Arduino no pot variar la tensió de sortida en els seus pins, només pot 5V de sortida (complets). Per tant hauràs d'utilitzar una tècnica anomenada Modulació **per amplitud de pòsols (PWM)** per apagar gradualment els LEDs amb `analogWrite()`.

PWM converteix ràpidament el pin de sortida d'alta i baixa tensió durant un període de temps fix. El canvi es produeix més ràpid que el que l'ull humà pot veure.



L'Arduino Uno disposa de sis pins reservats per PWM (pins digitals 3, 5, 6, 9, 10 i 11), podem ser identificats per el símbol ~ al costat del número.

Programació:

```

const int  greenLEDPin = 9;
const int  redLEDPin = 11;
const int  blueLEDPin = 10;

const int  redSensorPin = A0;
const int  greenSensorPin = A1;
const int  blueSensorPin = A2;

int  redValue = 0;
int  greenValue = 0;
int  blueValue = 0;

int  redSensorValue = 0;
int  greenSensorValue = 0;
int  blueSensorValue = 0;

configuración      vacía () {
    Serial . comenzar (9600);

    pinMode (greenLEDPin,  SALIDA );
    pinMode (redLEDPin,    SALIDA );
    pinMode (blueLEDPin,   SALIDA );
}

bucle             vacío () {
    redSensorValue = analogRead (redSensorPin);
    retraso (5);
    greenSensorValue = analogRead (greenSensorPin);
    retraso (5);
    blueSensorValue = analogRead (blueSensorPin);

    Serial . print ( "Valor RAW del sensor \ t Vermell:" );
    Serial . imprimir (redSensorValue);
    Serial . print ( "\ t Verd:" );
    Serial . imprimir (greenSensorValue);
    Serial . imprimir ( "\ t Blau:" );
    Serial . imprimir (blueSensorValue);

    redValue = redSensorValue / 4;
    greenValue = greenSensorValue / 4;
    blueValue = blueSensorValue / 4;

    Serial . print ( "Valor del sensor assignat \ t
Vermell:" );
    Serial . imprimir (valor rojo);
    Serial . print ( "\ t Verd:" );
    Serial . imprimir (greenValue);
    Serial . imprimir ( "\ t Blau:" );
    Serial . println (blueValue);

    analogWrite (redLEDPin, redValue);
    analogWrite (greenLEDPin, greenValue);
    analogWrite (blueLEDPin, blueValue);
}

```

Explicació:

- Estableix les constants per als pins que es fan servir per als orígens del LED i els pins d'entrada de les fotoresistències. Utilitzem el tipus de dades int .
- Afegim les variables dels valors entrants dels sensors i dels valors de sortida que utilitzarà per apagar gradualment el LED.
- Al setup () comença la comunicació amb el monitor sèrie a 9600 bps on es veuran els valors dels sensors. També, es defineix els pins del LED amb un sortides amb pinMode().
- En el loop() es llegeix els valors dels sensors a A0, A1 i A2 amb analogRead() i s'emmagatzema els valors a les variables apropiades. Posa un petit retard () entre cada lectura ja que l'ADC pren unes mil·lèsimes de segon per funcionar.
- Imprimeix els valors dels sensors a una línia. " \ t " és l'equivalent a pressionar el tabulador al teclat.
- La funció per canviar el brillantor del LED via PWM és analogWrite(). Té dos arguments: a escriure i el valor entre 0-255. Aquest segon nom representa el cicle de treball d'Arduino que donarà sortida al pin específic.
- El valor 255 establirà el pin HIGH tot el temps (LED encès), el valor de 127 establirà el pin HIGH la meitat del període (regulant la intensitat) i un valor de 0 establirà el pin LOW tot el temps (LED apagat). Per convertir la lectura del sensor entre 0-1023 a un valor entre 0-255 per analogWrite () , dividim la lectura del sensor entre 4.
- S'imprimeixen els nous valors assignats a la seva pròpia línia.

1. Introducció/Objectius:

2. Components/Materials:

3. Anàlisi-funcionament:

4. Anàlisi-Codi:

5. Canvis-realitzats:

6. Experimentacions:

7. Simulació-Tinkercad:

8. Fotos/Videos:

9. Aplicacions:

10. Problemes/Conclusions: