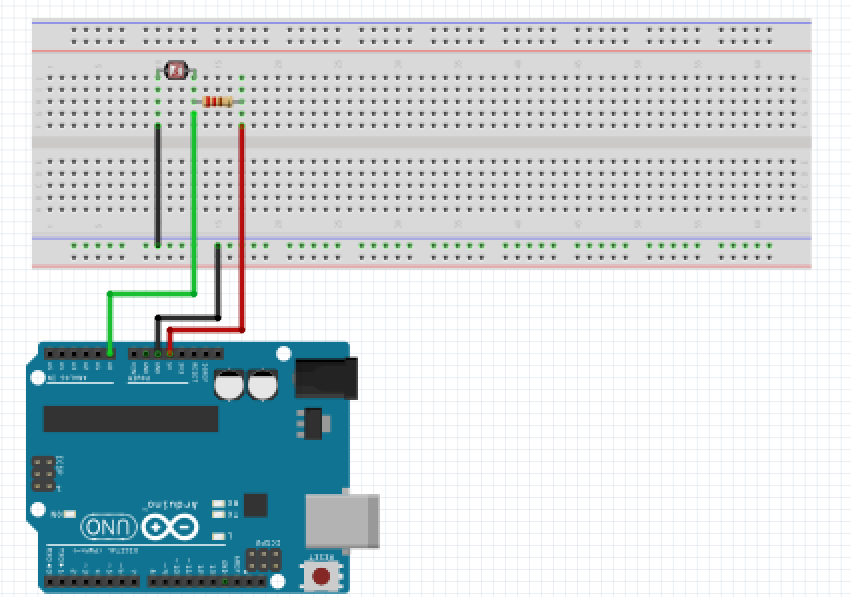
**ARDUINO LED PROJECT**

**Photocell**

For this project we will need:

* Arduino board.
* Photocell.
* Breadboard.
* 470 Ohm resistor.
* 3 Male to Male jumper wires.

Circuit Design:

1. First make sure that the Arduino is powered off (no USB cable plugged to power).
2. Check the two pins of the light sensor .Plug the legs of the sensor to separate horizontal lines to the breadboard.
3. Plug a black jumper wire from the shorter leg’s (cathode) horizontal line to pin GND on Arduino.
4. Plug one end of a 470Ohm resistor to the longer leg’s (cathode) horizontal line and its end plug to the 5V pin on the Arduino.
5. Plug a green jumper wire from the longer leg’s (cathode) pin and resistor horizontal line to theA0 pin of the Arduino.

|  |
| --- |
| const int ldrPin = A0;  void setup() {  Serial.begin(9600);  pinMode(ldrPin, INPUT);  pinMode(LED\_BUILTIN , OUTPUT);  }I  /void loop(){  int ldrStatus = analogRead(ldrPin);  Serial.println(ldrStatus);  if (ldrStatus <=900) {  digitalWrite(LED\_BUILTIN, HIGH);  delay(1000);  }  else {  digitalWrite(LED\_BUILTIN, LOW);  }  delay(100);  } |

const int ldrPin = A0; We instruct the Arduino to use these values as the pins attached to the flame sensor.

const the value cannot be changed during program execution or reassigned a new value.

int the value is a number.

Serial.begin(9600);

This initializes communication via serial communication port. This provides the user an interface to interact with the program during execution.

pinMode(lightSensor, INPUT);

pinMode(LED\_BUILTIN , OUTPUT);

This sets the declared pins , flame sensor and LED , functionality .

This allow as to receive INPUT and send OUTPUT to the Arduino.

LED\_BUILTIN this is the Arduino built in LED. Its attached to pin 13 for external LED connection if need be.

int ldrStatus = analogRead(ldrPin);

This reads the values received from the pin attached to sensor and stores it in number form.

Serial.println(ldrStatus);

This prints the value to a serial monitor where we can monitor them,

if (ldrStatus <=900) {

digitalWrite(LED\_BUILTIN, HIGH);

delay(1000);

}

This checks if the condition is met.

If the value meets this condition the arduino sends a high signal to the built in LED

else {

digitalWrite(LED\_BUILTIN, LOW);

if the input value did not meet the condition the program will send a signal LOW to the led.

This will turn it off.

voidloop(){}

After executing thevoid setup() function, we enter the void loop() and this function is executed continuously and repeatedly, until you Arduino is powered off.