

Experiment 4

Automated street lighting, staircase lighting design using Arduino

Aim: To learn about automated street lighting, staircase lighting design using Arduino.

Objective: To learn about:

PRO1: Automated Street lighting.

PRO2: Design using Arduino.

PRO3: Staircase lighting

PRO4: Design using Arduino

PRO1: Automated Street lighting:

Theoretical background: We are familiar with the working of an LDR that is a light-dependent resistor. Well in this blog we are using an LDR module to make an automatic light that can turn on and off automatically. We are using an Arduino UNO to operate the automatic street light control system. There is a problem associated with the street lights that they keep on during the daytime or early in the morning when there is no need for artificial (bulb) light. Sometimes the light from the sun is too bright especially in summers when the days are longer than night that you can clearly see the surroundings. So, keeping this in mind we are developing this project.

How Does automatic street light Work?

LDR module generates its output depending upon the light that falls on its upper surface. During the daytime, the sunlight falls on LDR so the AC bulbs are off, and after sunset, there is no source of bright light so the AC bulbs are turned on. By using this concept, we can manage to save a sufficient amount of energy per day.

Circuit Diagram:

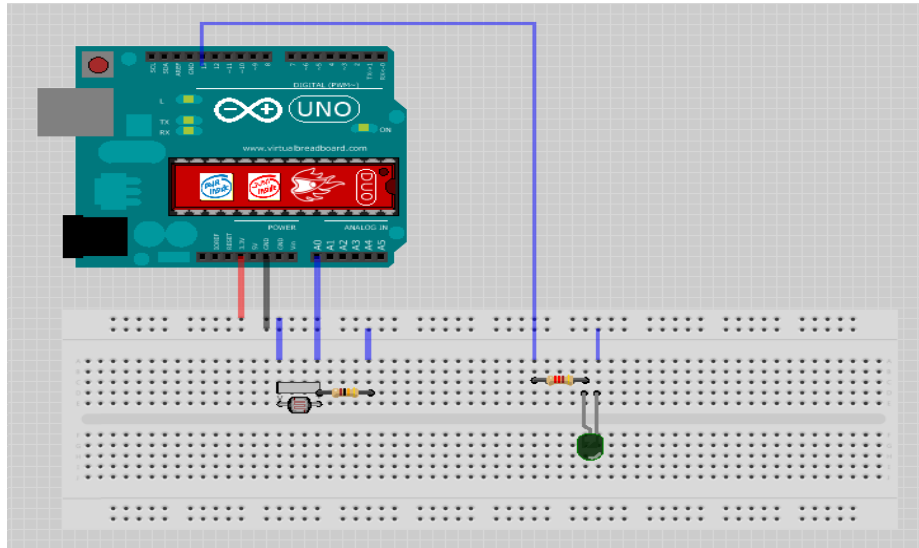


Figure 1. Circuit Diagram for Street light Controller using Arduino Uno Board

PRO2: Design using Arduino

Components Required

1. Arduino UNO
2. LDR
3. Breadboard
4. Jumper wires
5. USB cable for uploading the code
6. Resistor 10K-ohm, 22-ohm

Code:

Procedure:

1. Connect VCC of the Arduino to the positive rail of the breadboard
2. Connect the ground to the negative rail of the breadboard
3. Place the LDR on the breadboard
4. Attach the 10K resistor to one of the legs of the LDR
5. Connect the A0 pin of the Arduino to the same column where the LDR and resistor is connected (Since the LDR gives out an analog voltage, it is connected to the analog input pin on the Arduino. The Arduino, with its built-in ADC (Analog to Digital

- Converter), then converts the analog voltage from 0-5V into a digital value in the range of 0-1023). - Now connect the other end of the 10K resistor to the negative rail and the second (free) leg of the LDR to the positive rail
6. Open Arduino Uno Program Editor.
 7. Open new Sketch
 8. Write the code and save it.
 9. Connect Arduino Uno board with USB.
 10. Go to Tools-> Board ->& Select Arduino Uno
 11. Go to Tools-> Port ->& Select Com Port
 12. Compile & upload your program.
 13. Observe the effect of light on LDR.

Output: When light falls on LDR (day time) LED is off but when no light (night time) LED is on.

PRO 3: Staircase lighting

This relatively simple project uses a PIR motion sensor to detect people approaching the stairs. We can do staircase lighting by using this method. A **Passive Infrared Sensor** is used to detect the motion of the object that moves in or out of the sensor's range. It comes with two trim pots for adjusting the sensitivity and the delay time. When the sensor detects someone's movement on staircase then HIGH output is generated otherwise LOW output is generated. You can also observe the real-time readings on the Arduino serial monitor as shown below. When the HIGH output is received by Arduino then the LED will turn on and an AC bulb is also turn on. We use a Relay module here for turning the AC bulb on and off.

PRO 4: Design using Arduino

Components Required

1. Arduino UNO
2. USB cable for the project
3. PIR motion sensor
4. Jumper wires
5. LED and a 220-ohm resistor

Circuit Diagram

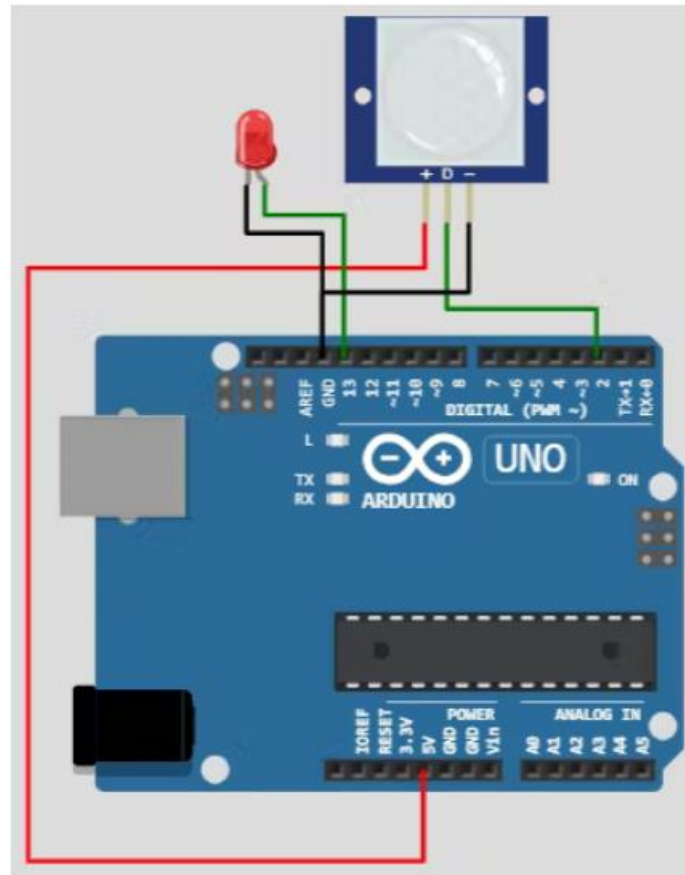


Figure 2: Circuit Diagram for staircase lighting using Arduino Uno

Procedure:

1. Attach the 5-volt pin of the Arduino with the VCC of the PIR motion sensor and the relay module
2. Connect the GND pin of the Arduino with the GND of the PIR sensor and the relay module.
3. Join the OUT pin of the PIR sensor with the digital-2 pin of the Arduino.
4. Connect the positive leg of the LED with the digital-9 pin of the Arduino through a 220-ohm resistor and the negative leg with the GND of the Arduino.
5. Open Arduino Uno Program Editor.
6. Open new Sketch
7. Write the code and save it.
8. Connect Arduino Uno board with USB.

9. Go to Tools-> Board ->& Select Arduino Uno
10. Go to Tools-> Port ->& Select Com Port
11. Compile & upload your program.
12. Observe the sensor effect of PIR.

Code:

Result: When PIR sensor sense someone in range LED is turned ON otherwise it is OFF.

