2019 Sec 4 Advanced Physics Arduino Lesson ( )

Use of LDR (Light-Dependent Resistor)

Name: _____ ( ) Class: _____ Date: _____

Objectives: At the end of this lesson, you would be able to

1. Apply basic circuitry to the Arduino microcontroller and other basic hardware
2. Write sketches for a light sensing circuit

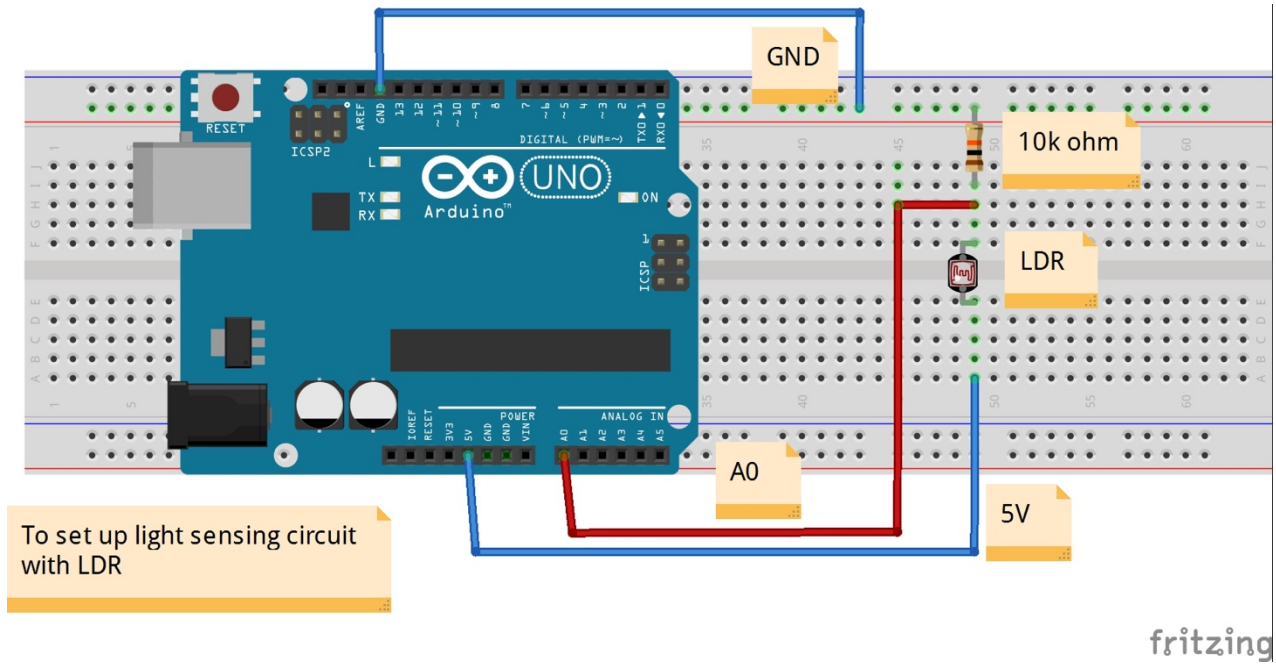
Apparatus:

- 01 Arduino UNO microcontroller
- 01 USB cable
- 01 breadboard
- 06 jumper wires

New item:

- 01 LDR (light dependent resistor)
- 01 resistor (10 kΩ) = R_1

***Note**: The light intensity readings can be calibrated using a **light meter**.*



Above diagram created with fritzing.org software

Using **potential divider**: same current $I = V_1 / R_1 = V_2 / R_2 = 5\ V / (R_1 + R_2)$

➔ $V_1 = R_1 / (R_1 + R_2) \times 5\ V$

Light intensity (in environment of LDR)	**High (bright)**	**Low (dim)**
Resistance of LDR (R_2)	Low	High
p.d. across LDR (V_2)	Low	High
p.d. across 10 kΩ resistor (V_1)	High	Low
A0: Reading on serial monitor	**High**	**Low**

Assignment: Set up a light sensing circuit using an LDR

1. Connect the LDR in series with a 10 kΩ resistor to GND (0 V) and 5 V.

2. Connect the analog pin A0 to middle of LDR-resistor circuit. This arrangement is called a **potential divider** (or voltage divider). The p.d. of 5 V is shared between the LDR and resistor.

- When the light intensity in the environment is high, the resistance of LDR will become low.
- The p.d. across LDR will become low, so the voltage output to A0 will be high, showing high light intensity.

3. Draw a simple circuit diagram for the LDR-resistor potential divider circuit.

4. Write a sketch to control the light sensing circuit which provides continuous readings of the light intensity in the environment. Include:

- Serial.begin (9600); //initialize the serial communication at 9600 bits/s
- Serial.print("light level ="); //output the light intensity detected by the LDR
- Serial.println(analogRead(A0)); //show light intensity readings from analog output A0
- delay (500);

5. Verify and upload the sketch.

6. Click on the "**serial monitor**" icon (top right corner) to open another window.

- Select "9600 baud" (bottom right corner menu).
- View changes in the light intensity readings by covering the LDR.

Sample sketch

```
/* Set up LDR to detect light intensity from the environment
*/

// the setup function runs once when you press reset or power the board
void setup() {
  Serial.begin (9600);      // initialize the serial communication at 9600 bits/s
}

// the loop function runs over and over again forever
void loop() {
  Serial.print("light level =");  // output the light intensity detected by the LDR
  Serial.println(analogRead(A0)); // show light intensity readings from analog output A0
  delay (500);                    // wait for 0.5 second
}
```

Additional item:

- 01 red light emitting diodes (LED)

Assignment 2: Light Sensing and Warning Circuit

To the previous circuit, add a red LED (connect to GND and to digital pin 6).
that blinks at different frequencies according to the light intensity using **while-loops**.

Write a sketch to include:

- Initialize digital pin 6 as an output for the Red LED
- Divide the entire range of light level sensed by your LDR into 3 segments.
- For the top segment, light the LED at 1.5 s intervals

```
while (analogRead(A0) > 600)
{
  digitalWrite (6, HIGH); delay (1500);
  digitalWrite (6, LOW); delay (1500);
}
```

- For the middle segment, light the LED at 0.7 s intervals
- For the bottom segment, light the LED at 0.05 s intervals

Verify and upload the sketch and observe the changes.

***Note**: The time duration is in milliseconds, e.g. 1500 means 1500 ms = 1.5 s.*

Sample sketch

```
void setup() {
  // put your setup code here, to run once:
Serial.begin (9600);
pinMode (6, OUTPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
Serial.println (analogRead(A0));

while (analogRead(A0) > 600)              // top segment light level condition
  {
    digitalWrite (6, HIGH); delay (1500);   // actions for top segment
    digitalWrite (6, LOW); delay (1500);
  }
while (analogRead(A0) > 400 && analogRead(A0)<600) // middle segment light level condition
  {
    digitalWrite (6, HIGH); delay (700);    // actions for middle segment
    digitalWrite (6, LOW); delay (700);
  }
    digitalWrite (6, HIGH); delay (50);     // actions for remaining bottom segment
    digitalWrite (6, LOW); delay (50);
}
```