



Current Trends – Internet of Things

Intensity Control of LED Using LDR
Sensor and Arduino Board

Project by:

Rushabh Shah (PRN: 14030142004)

Neehar Paranjpe (PRN: 14030142008)

Jaykumar Jayswal (PRN: 14030142009)

Manali Sarkar (PRN: 14030142011)

Introduction

The client of this project is a smart things solution provider who wants to control the intensity of a Light Emitting Diode (LED) according to the light conditions during day and night automatically. This means that during the day, the intensity of the LED should be minimum and during the night, it should be maximum. In addition to this, there is a need for manual control of the LED. If there are major fluctuations in luminosity, then there is a need to log the changes to a remote server. Also, there should be a provision to study the pattern of these changes.

Problem Statement

To develop an application to control the intensity of an LED according to light conditions during the day and night automatically as well as manually.

Proposed Solution

Terms and Definitions

1. Light Dependent Resistor (LDR)

Light Dependent Resistor (LDR) is a type of resistor whose value changes with intensity or the amount of light falling on it. As the amount of light falling on it increases its resistance decreases and the other way round. An LDR can be applied in light-sensitive detector circuits, and light- and dark-activated switching circuits.

2. Light Emitting Diode (LED)

LEDs are just tiny light bulbs that fit easily into an electrical circuit. But unlike ordinary incandescent bulbs, they don't have a filament that will burn out, and they don't get especially hot. LEDs can be used in LCD HDTVs to make dramatically thinner televisions.

3. Resistor

A resistor is a passive two-terminal electrical component that implements electrical resistance as a circuit element. Resistors act to reduce current flow, and, at the same time, act to lower voltage levels within circuits.

4. Arduino board

Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analogue inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

5. Connecting wires

Connecting wire is a piece of wire used to attach two circuits or components together. The gauge or size of the wire must be large enough to support the amount of current flow.

6. Breadboard

Breadboard is a board for making experimental model of an electric circuit. As the breadboard does not require soldering, it is reusable. This makes it easy to use for creating temporary prototypes and experimenting with circuit design.

7. Pulse Width Modulation (PWM)

Pulse Width Modulation is a technique in which the width of the pulses can be varied according to the message signal. Varying the width of the pulses can be seen as varying the average voltage on the pin. So although only the amount of time for which the LED is on or off is being changed, due to the persistence of vision, the varying voltage across the LED is perceived as causing it to fade and brighten up.

Solution Explanation

The solution to the above mentioned problem comprises of three parts:

1. The Web interface
2. Manual mode
3. Major Fluctuations' Logging

1. The Web Interface :

The user will navigate to the web interface by entering the IP Address provided to him. The web interface consists of three radio buttons namely:

- On
- Off
- Manual Mode

The '**On**' button will turn on the application in automatic mode. This means that the light intensity will increase and decrease automatically.

The '**Off**' button will turn off the application completely.

The '**Manual Mode**' button will enable the user to manually select the intensity of light.



2. Manual Mode

In the manual mode, the user will be provided with a dropdown list consisting of the percentage of intensity of light. This means that the intensity of the light will change according to the user's selection.

3. Major Fluctuations' Logging

When there is a major increase or decrease in the intensity of light, the change in intensity will be logged to a file on the server. Also, there will be a provision to analyse the data and identify the pattern of fluctuation.

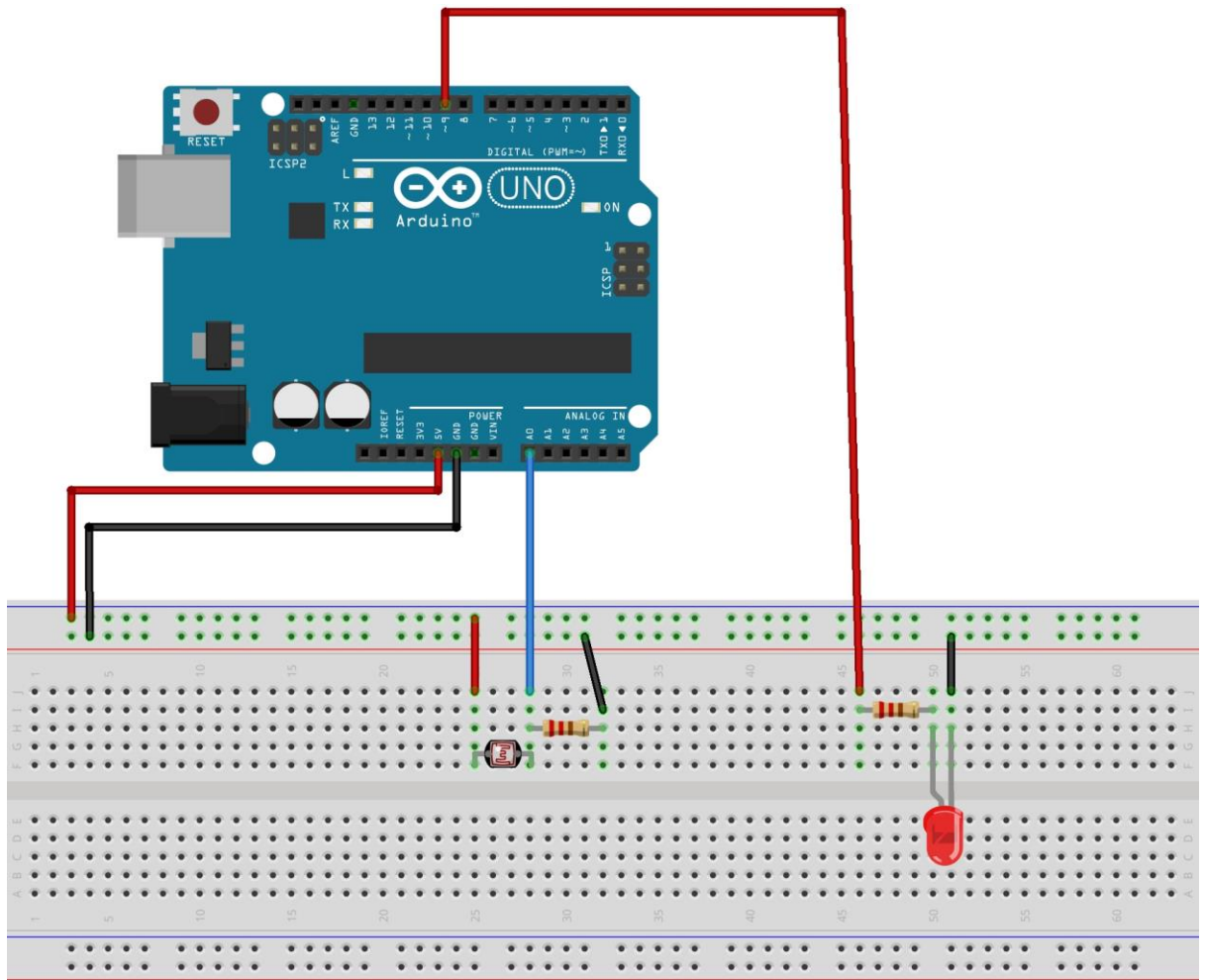
	A	B	C	D	E	F	G	H	I	J	K
	Date	Time	Min-Value	Max-Value	Real-Value						
1	9/29/2015	10:26:01 PM IST	65	250	500						
2	9/29/2015	10:26:01 PM IST	65	250	50						
3	9/29/2015	10:27:23 PM IST	65	250	257						
4											
5											
6											
7											

Connections

The connections are as follows:

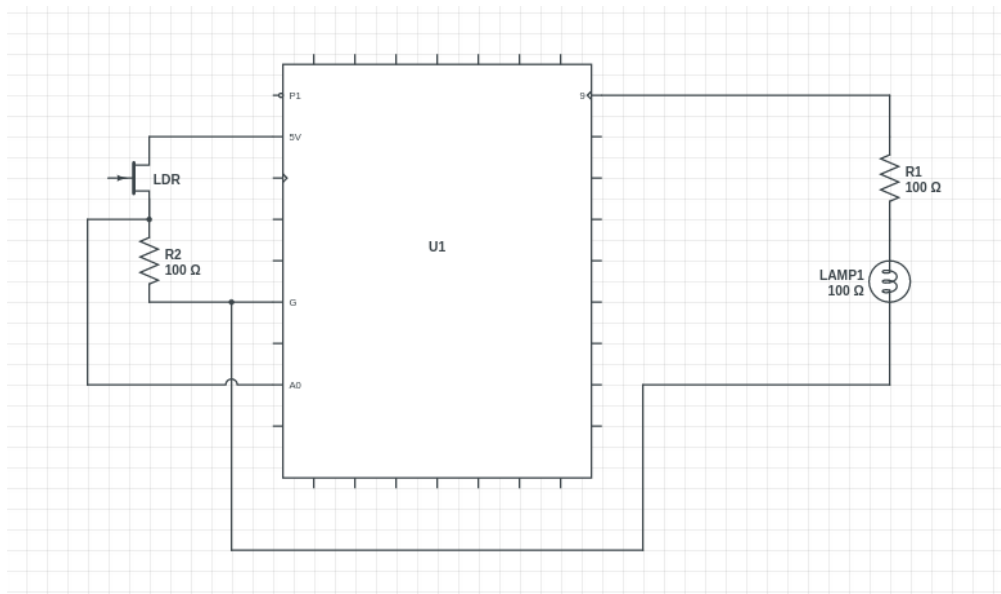
1. One end of a resistor is connected to the 5V pin of the Arduino board. The other end of the resistor is connected to one end of the LDR. The other end of the LDR is connected to the ground (GND).
2. A wire from the junction of the resistor & the LDR connected earlier should be taken and connected to the Analogue Input 0 on the Arduino board.
3. A wire from any of the PWM pins (in this case Pin 9) on the board should be taken and connected to the Anode of the LED. The cathode of the LED should be connected to one end of a resistor with the other end connected to the GND.

Fritzing Diagram



fritzing

Block diagram



Program Code

The program code is as follows: [code.pdf](#)

Application Areas

1. Street lights

The present system of street lights is that the lights will be switched on in the evening as the natural light decreases and they are switched off the next day morning as the natural light increases. But this is inefficient as sometimes, the lights are on even during the day. Due to this, the power is wasted. The above mentioned solution of using LDR can be implemented over here. The application program can be set to automatically switch on or off the lights, according to the natural light. So, the lights will be switched on only during the night, thus saving power.

2. Car headlights

Sometimes, due to human error, the headlights of the car remain on. As a result, the car battery drains and the car does not start. The above mentioned solution of using LDR can be implemented over here. The application program can be set to switch off the car headlights during the day and switch them on during the night. To determine when the headlights are to be turned on or off, the average intensity of light can be calculated. If the intensity is above average, then the headlights will be turned off. Alternatively, if the intensity is below average, then the headlights will be turned on.

Special cases

Limitations: User has to provide minimum and maximum intensity of light in first 5 seconds.

Best case 1: There is no client request for manual settings page as the sensor detects the brightness of light on its own and sets the LED accordingly.

Best case 2: There is no major fluctuation in the intensity of light.

Worst case 1: Major increase or decrease in the intensity of light after setting up the manual mode.

Worst case 2: There is major fluctuation in the intensity of light in automatic mode.

Conclusion

The solution to the problem of manipulating the intensity of an LED using an LDR sensor and Arduino board has been given above. From the above solution, it can be concluded that these Arduino-based applications can be very useful as they save a lot of power. This makes the application very convenient to use, thus making the world more advanced and aware about the Internet of Things.