1. **INTRODUCTION**

All batteries have certain voltage limit to discharge, if it goes beyond the prescribed limit, the life span of the battery will reduce drastically. Being electronics enthusiasts, we all might have a battery for testing our prototype circuits. Since we concentrate on the prototype during experiment, we care less on the battery.

The proposed battery charger circuit will show you how much energy left in the battery, this circuit may be connected to battery, while you prototyping your circuits. When this circuit indicates low battery, you may put the battery to charge. The circuit has 6 LEDs, one LED glow at a time to indicate the voltage level of the battery.

1. **LITERATURE REVIEW**

Battery level indicator is a circuit that is used to check the battery life. We can easily recognize the battery level with the help of LED’s. It uses an Arduino bord use to drive 6 LED’s by passing through its sufficient amount of current. Brightness of LED’s are controlled by reference adjustable pin and reference out pin. A variable resistor is also deployed in the circuit to have variable input voltage at pin 5 of the driver IC. Since the driver IC has two modes of operation which is bar graph mode and dot mode, it enables us to have indication of the battery life either in bar form or in dot form.

We can have colored LED to indicate state of the batteries like first three LED will indicate low battery. Green LED will indicate full battery. Thus, by using various colored LED representation of battery life will be made easy to understand. Cascading the driver we can increase the range of battery input in terms of voltage.

In the below given block diagram we have 4 blocks. One is the power supply. other is the driver Arduino itself. We will have a led block and an input block. A reference input for comparison purpose. LED will glow according to the input given and the reference Input present.

1. **FUTURE SCOPE**

This project can be further extended in a useful manner, the basic design of battery level indication using single Arduino bord can be further improved by adding display control function and alert mechanisms to enable the easy knowing of the battery level. Instead of using the LED’S and LED controller in this circuit using a microcontroller can enable battery level indication as in mobile phones.

**3.1 ADVANTAGES**

A battery level indicator is with electronic appliances to arrange as to display, on an indicator, a real time voltage detected by voltage indicator.

Indicator indicates how much power the battery will be able to supply to electronic apparatus.

It is used to check the battery level with the help of LED’s for example if three LED’s, indicates battery capacity of 30 percent. And if 6 LED’s glow then it is 100 percent.

Easily indicate the battery level

Enhanced version of this circuitry is the present mobile phone battery level indication system

This circuit can be used in household applications like INVERTER. This circuit connected to inverter can help the users to know when to charge and when to leave the inverter idle

Minimized version of this circuit can be also used for automobiles to indicate the battery level and low cost.

**3.2 DISADVANTAGES**

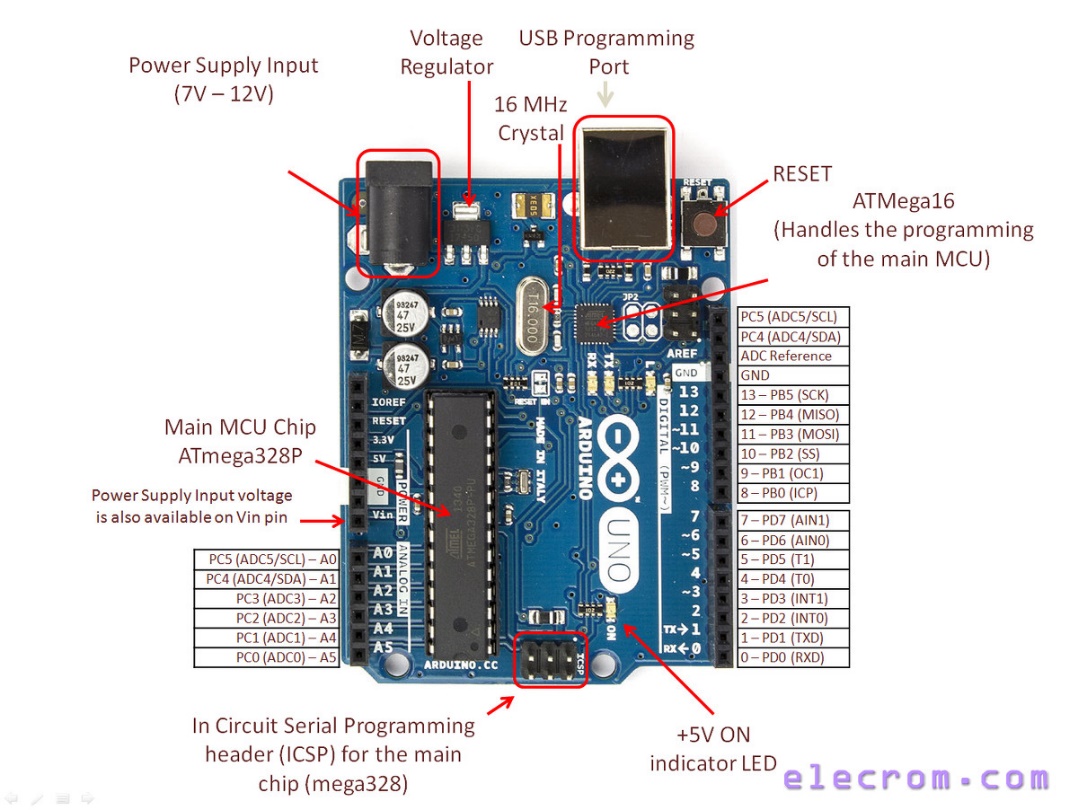
In case of voltage matching, correct level wouldn’t be indicated

1f input voltage is higher than the calibrated voltage, for full scale input error will be shown.

**4. SYSTEM DEVELOPMENT:**

**4.1 CIRCUIT COMPONENTS:**

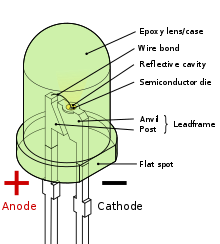
**1) ARDUINO UNO**



**Fig no.1 Arduino board**

Arduino is an open-source physical platform based on microcontroller board having the ATmega32 series controllers and Integrated Development Environment for writing and uploading codes to the microcontroller. It has input and output pins for interaction with the outside world such as with sensors, switches, motors and so on. To be precise it has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog 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 .It can take supply through USB or we can power it with an AC-to-DC adapter or a battery Arduino acts as the processing module of system. It takes input from the LDR, process the data and gives the output to LEDS directly or through a relay and a transistor mechanism

**2) LEDS**



**Fig no.2 LED**

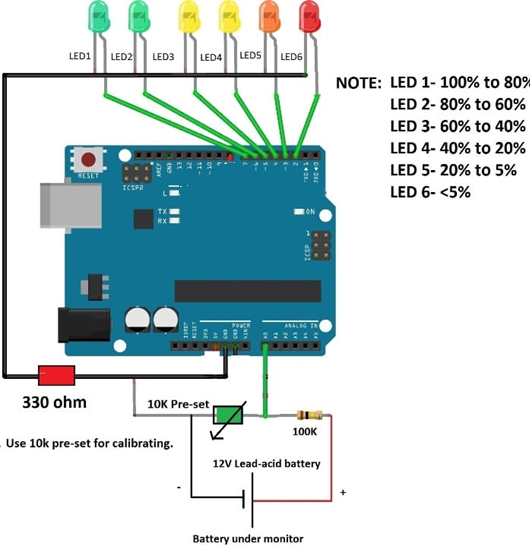
A light-emitting diode (LED) is a PN junction diode, which emits light when activated. When we apply voltage across its leads, electrons are able to recombine with holes within the LED, releasing energy in the form of photons which gives the light. Hence, it is a two-lead semiconductor light source Light emitting diodes represents our lighting system and the amount of light emitted by it is directly related to the amount of light in the environment that is when outside light is less than the light given by LEDS is more and visa-versa.

* 1. **WORKING PRINCIPLE**

Set the voltage of your variable power supply to precisely to 12.50V.   Open the serial monitor. Rotate the preset resistor clock wise or counter clock wise and bring the readings to 12.50 V. Now, reduce the variable power supply to 12.00V, the readings on the serial monitor should show the same or very close to 12.00VNow, increase the voltage to 13.00V, the readings on serial monitor should also show the same or very close. At the same time when you increase or decrease the voltage, each LED should turn on/off with different voltage levels. Once the above steps are done successfully, your battery level indicator circuit will be ready to serve the intended purpose. For calibrating the circuit, apply the power supply and adjusts VR1 so that LED6 just start glowing. Now, increase the input voltage in steps of 1.2V and see LED5 will glow in 2.4V and similarly increase the voltage in steps of 1.2V and LED4 will glow and so on. If above phenomenon is happening your circuit is ready is use. Connect it to voltage from any battery to be tested at the input which will become power source for the circuit. And now by seeing the number of LED's glowing you can easily calculate the status.

**4.3 SCHEMATICS VIEW:**

**The schematic diagram of the smart lighting system is given below:**



**Fig no.3 Circuit diagram**

The wire connections are simple as you can see on the images above. Using 16x2 LCD and its I2C adapter. Adapter GND to Arduino GND. Adapter VCC to Arduino 5V. Adapter SDA to Arduino A4 (or the pin next to AREF on Digital Pins side).Adapter SCL to Arduino A5 (or the pin next to SDA, two pins from AREF on Digital Pins side).Align two 10K resistors in series on breadboard .Connect the middle of the resistors-in-series to Arduino A0.Connect one end to Arduino GND and also to Battery - (negative).Connect the other end to Arduino Vin and also to Battery + (positive). I think you should connect this one after you load the Arduino Sketch as I tell you my reason before.

Using a LED instead of LCD:

Connect LED anode (small piece inside) to Arduino D13.

Connect LED cathode (large piece inside) to GND (next to D13).

Align two 10K resistors in series on breadboard.

Connect the middle of the resistors-in-series to Arduino A0.

Connect one end to Arduino GND and also to Battery - (negative).

Connect the other end to Arduino Vin and also to Battery + (positive).

When you plug the battery to Arduino Vin, it should work right away showing the voltage of your battery on your 16x2 LCD because Arduino is powered by that battery. If it is not working, please re-check your connection or the battery you use might be lower than 5 volts needed by Arduino to power up. Please try another battery or check it with your voltmeter .On my test with multimeter, the voltage shown on the LCD is slightly lower then the multimeter display. We are loosing around 0.05V to 0.15V on breadboard and Arduino circuits.

**4.5 CODE**

int analogInput = 0;  
int f=2;  
int e=3;  
int d=4;  
int c=5;  
int b=6;  
int a=7;  
int s=13;  
float vout = 0.0;  
float vin = 0.0;  
float R1 = 100000;  
float R2 = 10000;  
int value = 0;  
void setup()  
{  
Serial.begin(9600);  
pinMode(analogInput,INPUT);  
pinMode(s,OUTPUT);  
pinMode(a,OUTPUT);  
pinMode(b,OUTPUT);  
pinMode(c,OUTPUT);  
pinMode(d,OUTPUT);  
pinMode(e,OUTPUT);  
pinMode(f,OUTPUT);  
digitalWrite(s,LOW);  
digitalWrite(a,HIGH);  
delay(500);  
digitalWrite(b,HIGH);  
delay(500);  
digitalWrite(c,HIGH);  
delay(500);  
digitalWrite(d,HIGH);  
delay(500);  
digitalWrite(e,HIGH);  
delay(500);  
digitalWrite(f,HIGH);  
delay(500);  
digitalWrite(a,LOW);  
digitalWrite(b,LOW);  
digitalWrite(c,LOW);  
digitalWrite(d,LOW);  
digitalWrite(e,LOW);  
digitalWrite(f,LOW);  
}  
void loop()  
{  
value = analogRead(analogInput);  
vout = (value \* 5.0) / 1024;  
vin = vout / (R2/(R1+R2));  
Serial.println("Input Voltage = ");  
Serial.println(vin);  
if(vin>12.46) {digitalWrite(a,HIGH);}  
else { digitalWrite(a,LOW);}  
if(vin<=12.46 && vin>12.28) {digitalWrite(b,HIGH);}  
else { digitalWrite(b,LOW);}  
if(vin<=12.28 && vin>12.12) {digitalWrite(c,HIGH);}  
else { digitalWrite(c,LOW);}  
if(vin<=12.12 && vin>11.98) {digitalWrite(d,HIGH);}  
else { digitalWrite(d,LOW);}  
if(vin<=11.98 && vin>11.90){digitalWrite(e,HIGH);}  
else {digitalWrite(e,LOW);}  
if(vin<=11.90) {digitalWrite(f,HIGH);}  
else {digitalWrite(f,LOW);}  
delay(2000);  
}

1. **CONCLUSION**

This Arduino based project will provide a competent method for charging systems and make the whole process of energy saving indication. The battery level indicator consists of very less component and thus occupies very little space of the surrounding. Since only DC input voltage levels are indicated, components which work in AC supply can be used to detect AC levels. Since AC voltage having continuously varying amplitudes the LED’s will turn ON and OFF continuously making a good-looking pattern. Thus, our battery level indicator can be used for decoration purpose also.

1. **REFERENCES:**

1 www.electrom.com

2 www.homemade-circuits.com/battery-level-indicator-circuit-using

3 Raymond CHANG. Chemistry- 9th Edition. Boston, MA: Williams College; 2007.

4 GP International Limited. Technical Handbook on NiMH technology [online].

England: Worcestershire; June 2002.

URL: http://www.steatite-batteries.co.uk/assets/files/NiMH\_technical.pdf.

Accessed 17 February 2014.

5 Steven S. Zumdahl and Susan A. Zumdahl. Chemistry 7th Edition. Chicago,