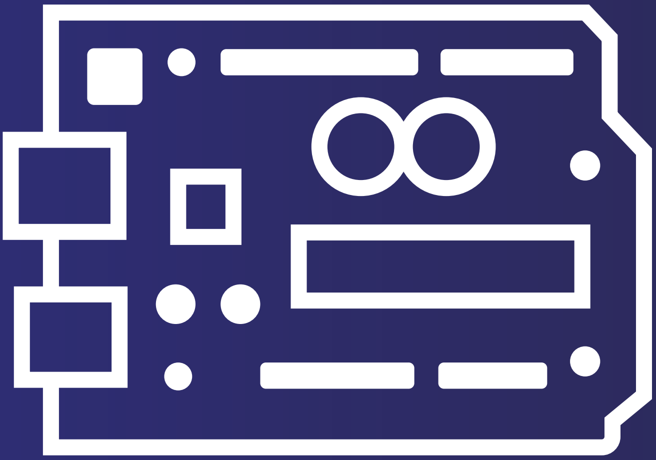
2021

Basic Kit

For Beginners



ORANGE KIT

Less hassle , more Fun !

01. Index

02. Index

01.Arduino Introduction

In this section of this blog, we will talk about the Arduino Development Board. Arduino is an open source project and was started with the intention of encouraging embedded systems students to learn about micro-controllers.

Before we begin, let me clear one very important doubt which every beginner faces in the initial stage of their learning phase and that is the difference between microcontroller and microprocessor.

1.1. Difference Between Microcontroller and Microprocessor

If you search this topic on the internet you will find a lot of information on this topic but as a beginner to all these things, this information will obviously confuse you.

But don't worry, I have explained that thing in the below section in such a way that it will clear all your doubts and after reading this you will not need to look back on this topic.

So, both microcontroller and microprocessor are things that are designed to control applications, perform logical and mathematical operations.

If both are designed to perform the same operation, what is the difference between these two things? This question may be on your mind.

The difference is, microcontrollers are designed to perform only predefined tasks whereas microprocessors are designed to perform tasks by considering the conditions that occur at the runtime of the process.

The other difference is that if you want to use a microprocessor then you have to interface the memory unit, EEPROM and everything that is essential to perform the tasks. Microcontrollers, on the other hand, don't need all those things because those things are built-in with them.

In the next section we will learn more about Arduino.

Do you know the first microprocessor was desined by intel in 1971 and the name of that microprocessor was 8081!!!

1.2. What is Arduino?

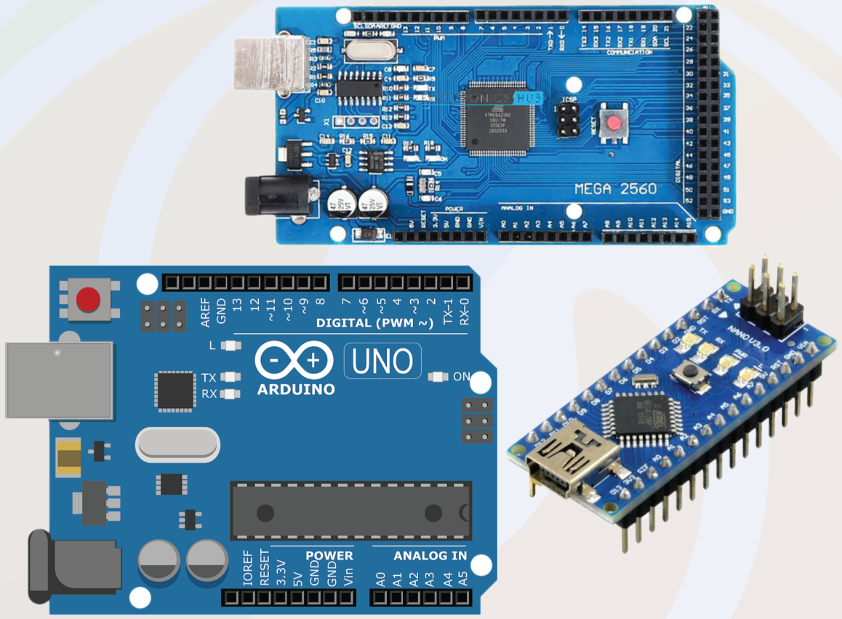
As discussed earlier, Arduino is a microcontroller that can be used to build small-scale applications. Nowadays, we can see the use of Arduino extensively in the automation industry, there are also a lot of jobs available in the Indian market for Arduino Masters.

The reason for the massive popularity of Arduino is that Arduino is an open-source project and is designed by the community that constantly works on it.

Since a large number of people contribute to this open-source project, Arduino users get quick solutions to their problems and their work never stops. And That is why arduino is popular in DIY community.

Varieties in board versions

This is another most important factor of Arduino boards.



Arduino boards are available in so many options. It gives an option to the individuals to select the boards as per their application requirements.

Talking about this orange basic kit, this kit is avilable in three options those options are as follows

1. Orange Basic Compatible With Arduino Uno
2. Orange Basic Compatible With Arduino Nano
3.. Orange Basic Compatible With Arduino Mega

1.3. How To Install Arduino IDE?

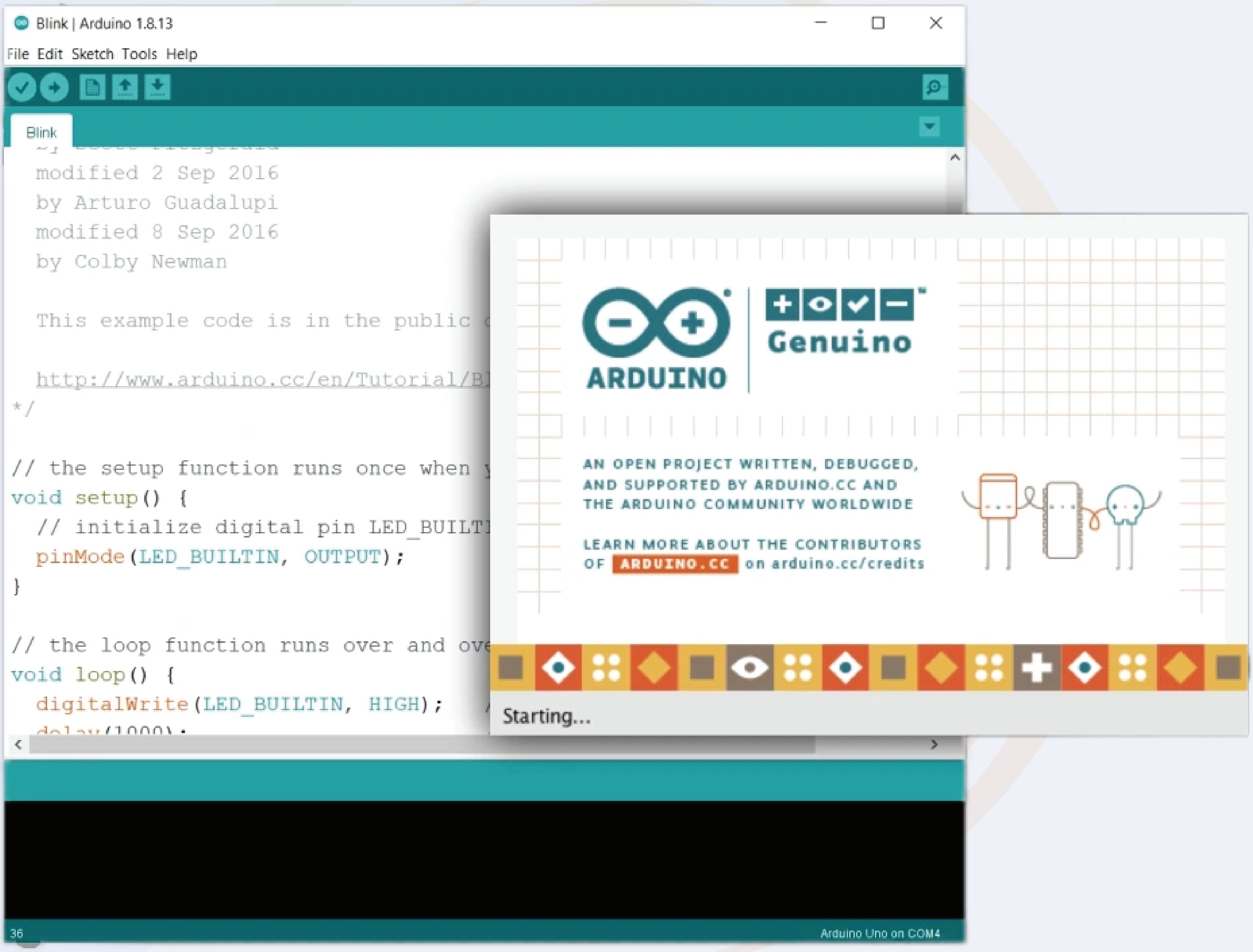
The Arduino IDE is a lightweight software used to upload your written programs to Arduino boards.

This IDE gives us an easy-to-use UI that makes complex stuff so easy to understand.

In the following part, I have explained all these things step by step. please take a look.

To install Arduino IDE on your system, you need to download the software package from the internet.

1. I have shared the download link to download Arduino IDE in the blog. You can download Arduino software from there or you can download it from their official website.



```
  modified 2 Sep 2016
  by Arturo Guadalupi
  modified 8 Sep 2016
  by Colby Newman

  This example code is in the public 

  http://www.arduino.cc/en/Tutorial/Bl
*/

// the setup function runs once when y
void setup() {
  // initialize digital pin LED_BUILTI
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and ove
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);   /
  delay(1000);
```

AN OPEN PROJECT WRITTEN, DEBUGGED, AND SUPPORTED BY ARDUINO.CC AND THE ARDUINO COMMUNITY WORLDWIDE

LEARN MORE ABOUT THE CONTRIBUTORS OF ARDUINO.CC on arduino.cc/credits

Starting...

36

Arduino Uno on COM4

1.3. How To Install Arduino IDE?

2. After the download is complete, extract that zip file. You can use WinRAR software to extract the file.

3. After extracting that zip file, you will see a folder. In that folder, you will find an .exe file.

4. Right-click on that file and select 'Run as administrator' option.

5. When you will click on that option, the system will start installing Arduino software.

6. Follow the instructions asked by the installer and do not make any changes to the settings of the installer.

7. Those options are for the user who has installed Arduino on their system and they are updating the software or reinstalling the software on their system.

8. In your case, you are installing the software afresh, so you do not need to make those changes to your Arduino installation process.

9. While installing the software, the software installer will ask you to install the driver on your system. You have to allow the installer to install that driver.

10. Those drivers are of Arduino board and if you do not install them on your system then your system will not recognize Arduino board.

11. After this process is over, you will see the Arduino IDE icon on your system's applications list.

Congratulations !! You have successfully installed Arduino IDE on your system.

1.3. How To Install Arduino IDE?

There is one USB driver that you need to install on your system. The name of that driver is CH340G driver and you will get the link on the product page to download that driver.

Please note - there is a misconception that boards with ch340G drivers are of poor quality but this is not true.

CH340G is just a driver used to transfer information from your computer to the controller board.

It is just a channel between the microcontroller and your computer. I have worked on those boards but never faced any problems.

So don't be concerned about it.
The board will function similarly to the other boards.

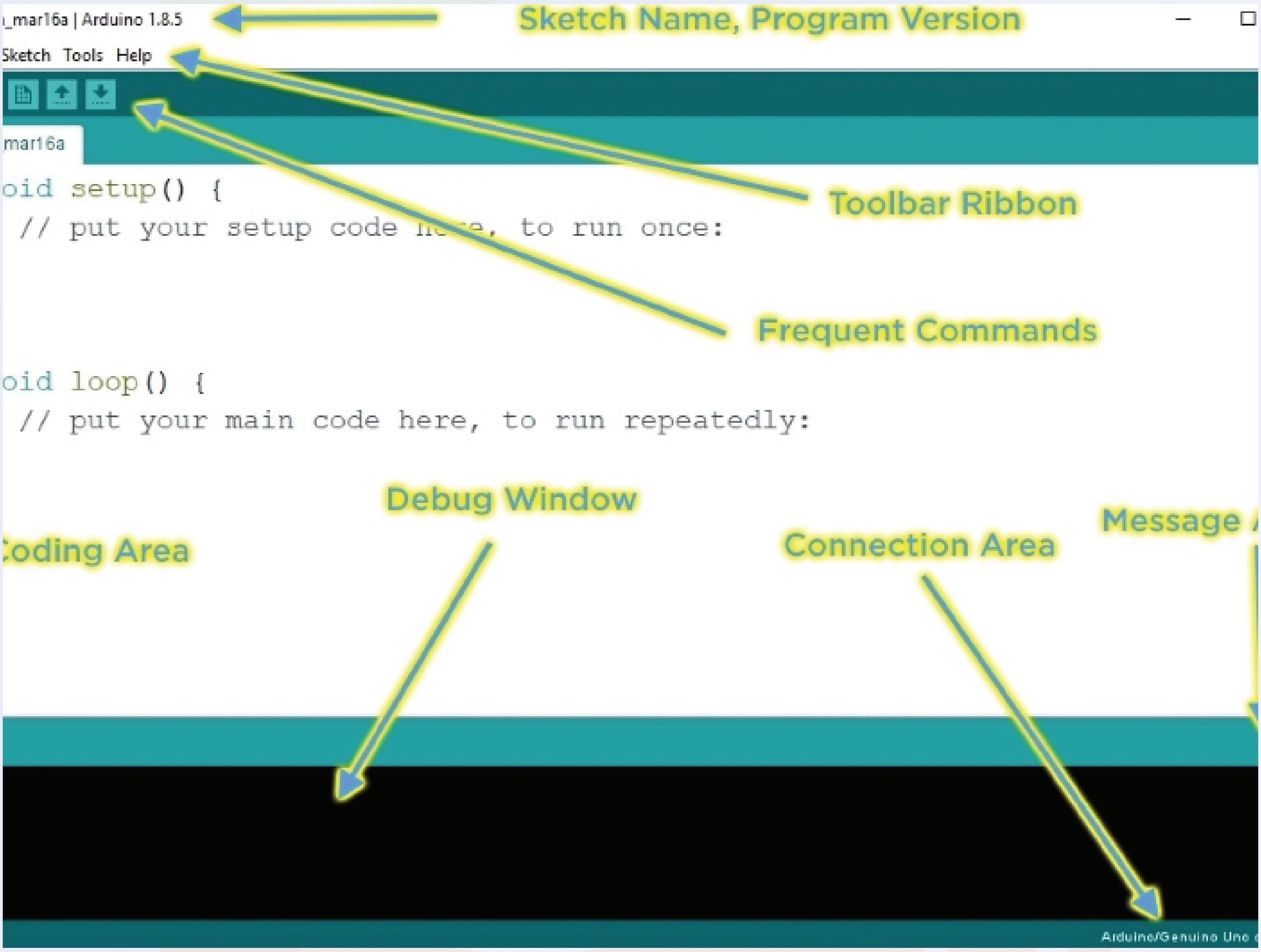
In the next part of this blog, we will learn about the UI of Arduino IDE and understand the usage of buttons available in the application.

The First Arduino Board was desingned in 2005. The main aim behind designing the arduino board was to tech microcontroller to novoice.

2.0. Arduino IDE - User Interface

Now that you've installed Arduino IDE on your computer, you can open it.

When you open the Arduino IDE, you'll see a text editor, as shown in the image below.



```
oid setup() {
  // put your setup code here, to run once:


oid loop() {
  // put your main code here, to run repeatedly:
```

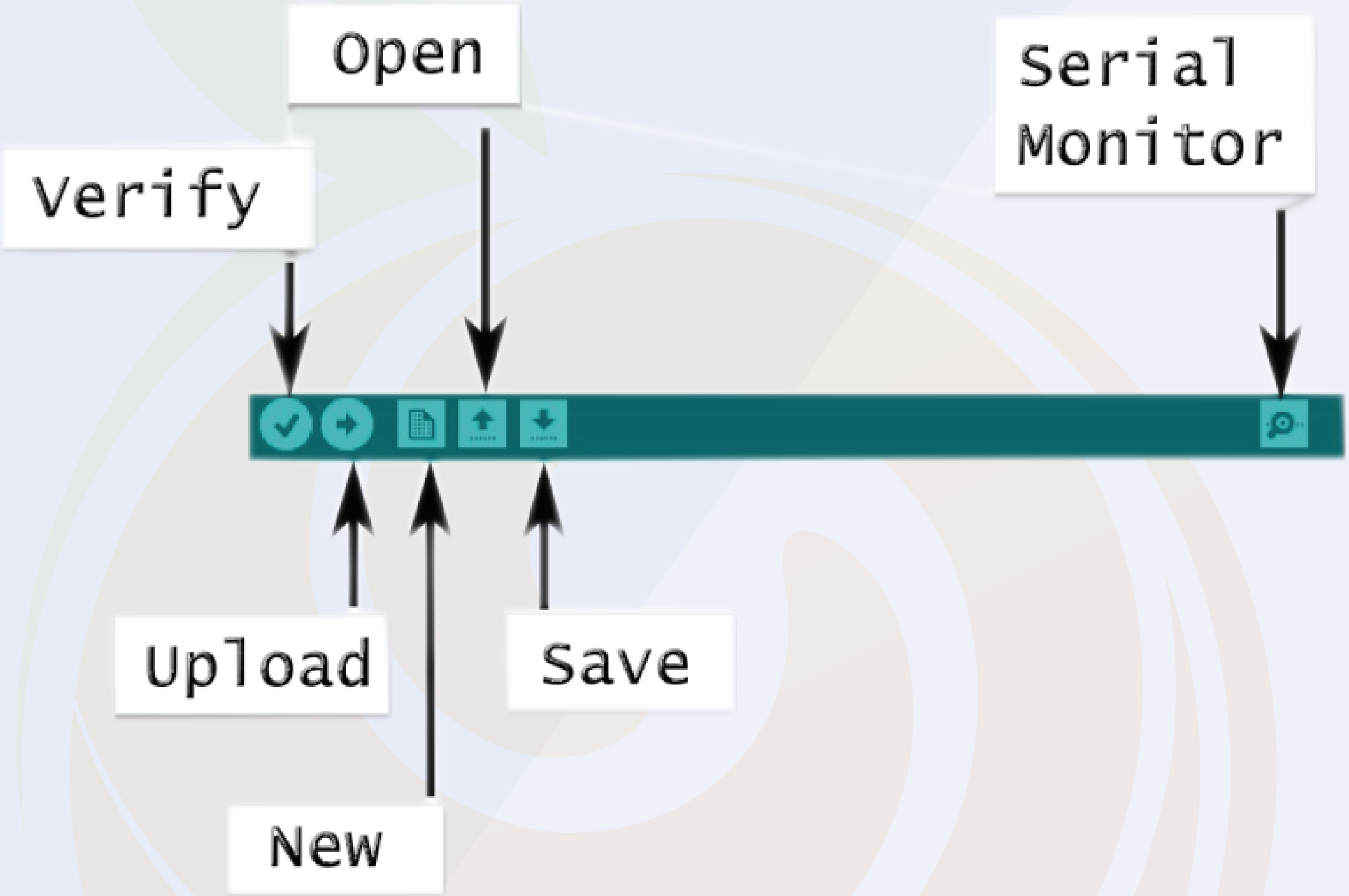
The Arduino IDE looks like this. In the above UI, you get all the functions that are useful for uploading code to your Arduino.

The IDE is divided into three sections. Text editor, output section and menu section.

You write your code in a text editor and you can upload your code to your Arduino with the help of the options available in the menu section.

2.1. Menu Option

Talking about the menu option, in that menu you get the following options.



The menu option looks like this. I have explained the working of those functions below please have a look.

2.1. Menu Option

Talking about the menu option, in that menu you get the following options.

1. File

- In this option you will get the option to save the file on the system and open the recent files if needed.

2. Edit

- Using this option, you can adjust the settings of your editor.

3. Sketch

- In this section you get the option to add libraries. To check what the library is all about, we request you to check the following section.

4. Tools

- This option is made for the selection of information related to the board. With the help of this option you can either add boards or install new boards in your IDE.

5. Help

- As the name suggests, you can use this section if you need any help regarding the software or program you have written.

5.Search

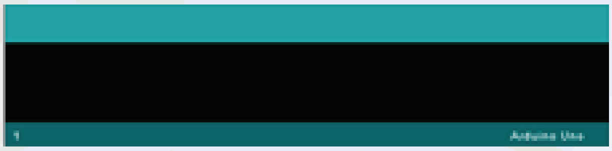
- This option is of serial monitor and with the help of this option you can open serial monitor.

If your code is using the serial print function, you will see the code's output on the serial monitor.

2.1. Menu Option

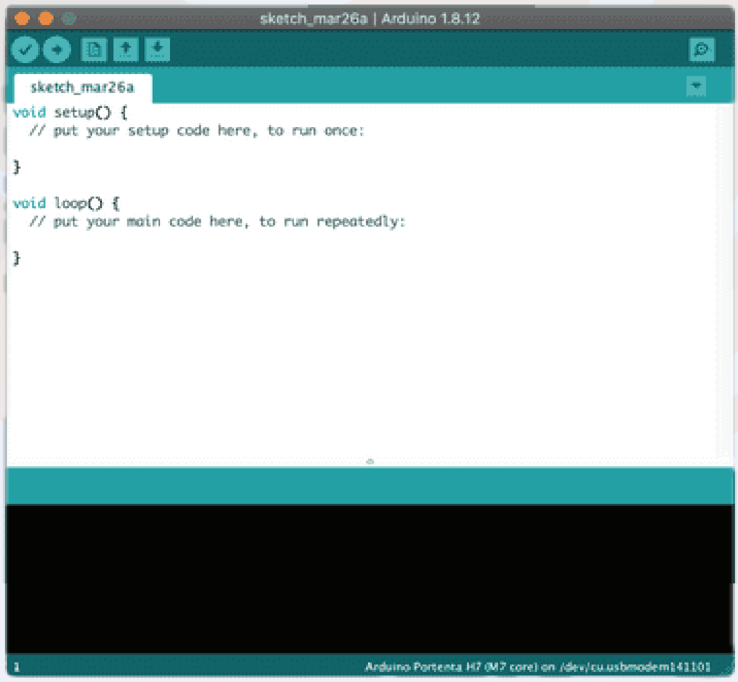
Output

In this option you will see the update of the uploading process of the code. If there is any problem with your code then IDE will generate some error and those errors we can see in this section.



Arduino Text Editor

This section is designed for Arduino Code. In this section, we can write our Arduino.


sketch_mar26a | Arduino 1.8.12
sketch_mar26a
1
Arduino Portenta H7 (M7 core) on /dev/cu.usbmodem141101

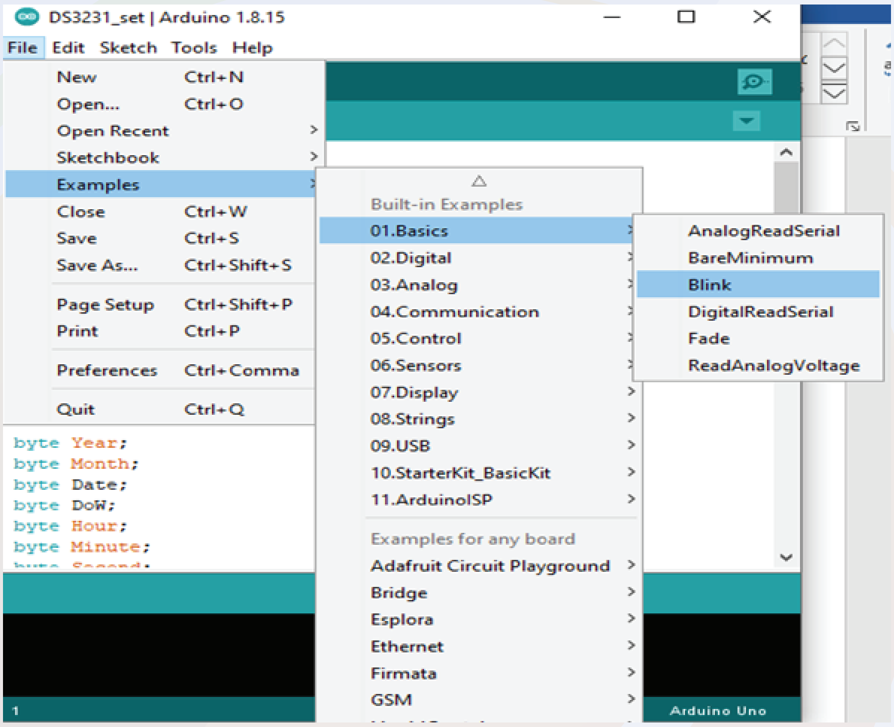
```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

2.2. How To Upload The Arduino Code To The Arduino Board?

We looked at various sections of the Arduino IDE in the preceding section.

We'll use those sections to upload our first Arduino code to the Arduino board in the next section.

We'll try to get the blink code to work on the board.



```
byte Year;
byte Month;
byte Date;
byte DoW;
byte Hour;
byte Minute;
```

When you click on the blink option in the image, a new Arduino tab will open, and in that tab, we'll find the Arduino code for toggling the 13-number pin.

So, now that we have the code, we will upload it to the Arduino. To do so, press the ctrl+U key on the keyboard to begin the process of uploading the code.

So this is how, we can upload the code to Arduino.

In the next part of this blog, we will learn how to add libraries to that Arduino IDE?

3. What is an Arduino Library?

The Arduino library is a combination of code that has been written by an Arduino contributor.

The main purpose of designing libraries is to reduce code redundancy.

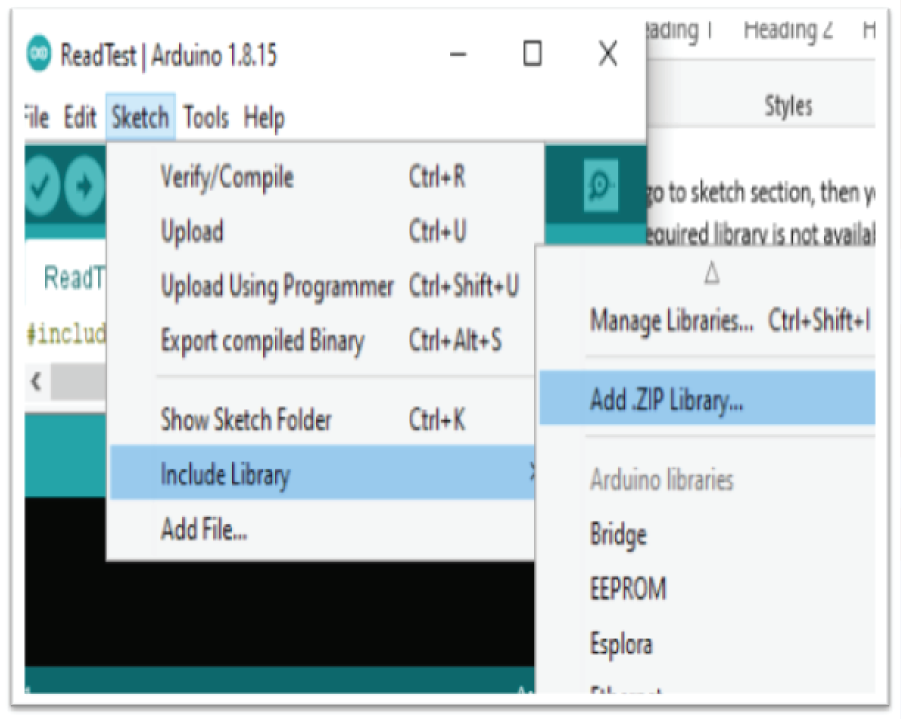
We use Arduino libraries, that means we use objects and methods of classes. So, even though we don't need to write much code, we can learn a lot by modifying existing code.

We can learn oops concepts, functional programming and much more.

So, that was about introduction to Arduino library, if you have any doubt you can contact me at info@robu.in.

How to Add an Arduino Library to the Arduino IDE?

There are two ways of adding library, either you can go to sketch section, then you can add the required library by clicking on the add library option or if the required library is not available on Arduino then we get 'Add Zip Library' option in that section.


ReadTest | Arduino 1.8.15
File Edit Sketch Tools Help
Verify/Compile Ctrl+R
Upload Ctrl+U
Upload Using Programmer Ctrl+Shift+U
Export compiled Binary Ctrl+Alt+S
Show Sketch Folder Ctrl+K
Include Library
Add File...
Manage Libraries... Ctrl+Shift+I
Add .ZIP Library...
Arduino libraries
Bridge
EEPROM
Esplora

3.1. What is an Arduino Library?

I have faced this issue many times, I never got my required library in the Arduino IDE. I always used to download the zip files of the library from the internet and then used to add that to the Arduino with the help of the ‘Add zip Library’ Option.

I hope the above section has cleared all your doubts about what is Arduino library and how to add a new Arduino library to the Arduino?

Arduino is an open source platform supported by huge community that continuously contributes to help others. So do not worry because the community will be always there to help

Void setup() is used for initializing the pins. This function executes once only.

If you are using a sensor that needs calibration before running, then you can do all those calibration settings in this section.

In Arduino, We use pinMode built in function to initialize the pins of the Arduino.

This function takes two parameters, pin number and the mode of operation. (Either output or input).

```
For Example: pinMode(12, INPUT);
             pinMode(12, OUTPUT);
```

Please check the above examples. In the above examples, the first line is defining 12 number pin as an input pin and the second line is defining the 12 number pin as an output pin.

Let me tell you a simple logic here, if you are defining any pin as an output pin that means you are putting some voltage on that GPIO pin or sending out some data.

Whereas if you are defining any pin as an input pin means, you are taking some data in.
Usually, we use this function when we connect sensors to the Arduino.

3.2. Void Loop And Void Setup

Voidloop()

This is one more inbuilt function that is used in the Arduino IDE. This loop keeps continuously running until someone don’t turn Off the Arduino.

We can write our code-cases in this loop and those cases will keep on running.

Analog Write And Analog Read

Analog read and analog write these are the two functions that deals with the analog values.

AnalogWrite function is used to write the analog values whereas analogRead function is used to read the analog values that we pushing on the GPIO pin of the Arduino.

```
Ex, analogWrite(1, pwm)
    analogRead();
```

The analogwrite function take two parameters, pin number and pwm value.

The pwm value could be in range of 0 to 255.

Whereas the analogread function only take one parameter and that is pin number.

```
Store = analogRead(A1);
```

In the above example the analogRead function is reading the the data which is coming on the pin number A1 and then storing the information in the store variable.

So, this was about the analogRead function. This function is used when we are required to work with the variable voltages.

In next section of this blog we will learn about the digitalRead and digitalWrite function.

3.2. Void Loop And Void Setup

Digital Read And Digital Write

DigitalRead and digitalWrite functions are used for reading and writing digital values.

Talking about the digitalWrite function, this function takes two parameters, pin number and HIGH/LOW String.

When we give HIGH parameter to the function, the function will put HIGH level signal on the GPIO pin of the Arduino and when we put LOW then the function will produce LOW level signal on the GPIO pin.

For Example,

```
digitalWrite (12, HIGH);
digitalWrite (12, LOW);
```

In the above line of code, the first line of code producing HIGH level signal on the 12 number pin and in the second line code the function is producing LOW level signal on the 12 number pin.

Talking about the digitalRead function, this function takes one parameter and that is pin number.

For example,

```
Store = digitalRead(12);
```

In the above function, the code is running the data of the pin number 12 and storing the received input in the ‘store’ variable.

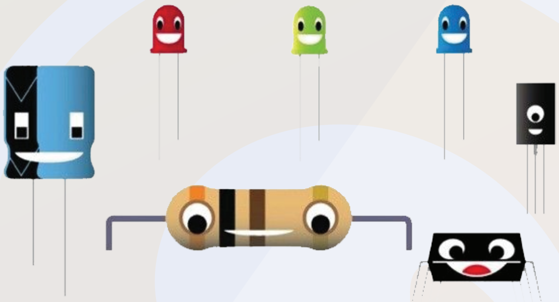
4. Electronic Components

As an electronic geek, you may have seen so many electronic components till now. Those components categorized into two main categories.

1. Active components.

2. Passive components.

Combination of these components used to manipulate and influence the electricity flow through them.
In the next section, we will study in depth information about these components.



Active Components

Active components are the components that amplifies the input power. Generally speaking, active components called donors. They amplify the input power and gives the output to the electronic circuit.

Common Examples of Active Components Include:

- Voltage sources
- Current sources
- Generators (such as alternators and DC generators)
- All different types of transistors (such as bipolar junction transistors, MOSFETS, FETs, and JFET)
- Diodes (such as Zener diodes, photodiodes, Schottky diodes, and LEDs)

4. Electronic Components

Passive Components

The function of the passive components depends on the output of the active components. These components store, consumes and dissipates the energy.

In normal terms, we can call these components accepters.

Common examples of passive components include:

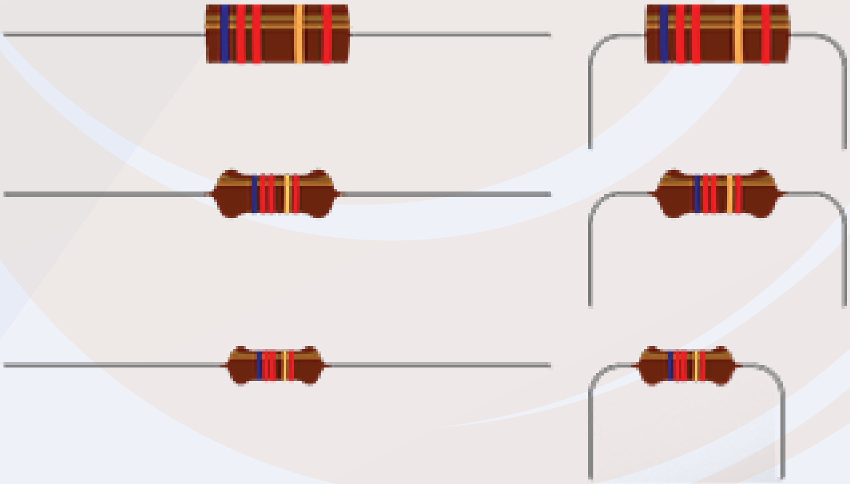
- Resistors
- Inductors
- Capacitors
- Transformers

Resistor

We are getting a few resistors with this kit. The main application of the resistor is of resisting the flow of the current.

The behaviour of the resistor depends on ohm's Law.

Those who don't know what is ohm's law, ohm's law states that the current flowing through the cable is directly proportional to the voltage and inversely proportional to the resistance of the cable.



4. Electronic Components

So, this was about the introduction to the resistor. But how to calculate the value of the resistor and how to choose the right value of the resistor.

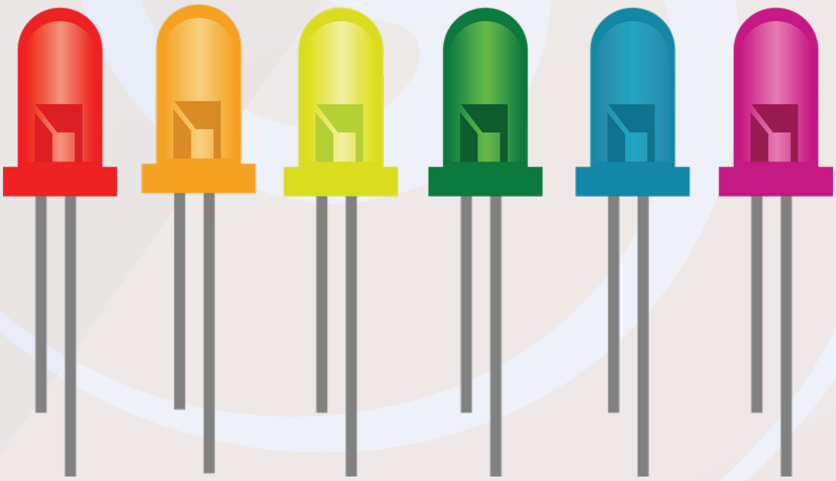
There are two ways of measuring the resistor. One is by measuring the value of the color bands that are printed on the resistor and the other one is by using the multi meter to measure the multi meter.

I have explained all these methods in the blogs of this kit. You can learn more about those methods from there.

5. LED (Light Emitting Diode)

Light Emitting Diode is a type of an PN diode. It emits the light when the applied voltage crosses cut-off voltage.

In the below section we will learn about the construction of the LED then we will learn about the working principle of the LED and then after we will understand the interfacing of the LED with the Arduino.



5. LED(Light Emitting Diodes)

LED are designed using a certain type of material. The color of the LED depends on the material that is used in the construction of the LED. I have shared the list of those materials below please take a look and if there is any doubt please let us know in the comment section.

Construction of The LED	Colour	Forward Voltage
GaP	Green/Red	2.2
GaAsP	Yellow	2.2
GaAsP	Red	1.8
GaN	White	4.1
AllGaP	Amber	2.1

Talking about the internal structure, of the LED. The P type material is placed on the top of the N type material. The P type material could have kept below the N-type Material but in that case, the Light was not be visible.

Talking about the P material, the majority carriers in the P type material are holes and the Electrons are majority carrier material in N-type material.

5. LED(Light Emitting Diode)

We know that the P-type material is placed on the top of the N-type material. If we talk construction wise, the P-type material is placed in the Valance band and N-type material is placed in the conduction band.

The Energy level of the valance band is more than the energy level of the conduction band. So, when we power the LED, the holes that are present in the valance band jumps in to the conduction band. And while coming back to the valance band, the extra energy that these holes are carrying; they release that energy in the form of light.

This process of emitting the light of the LED is called Electro-luminance effect.

So, in this way, we have learned the working of the LED.

In the next section, we will learn about the interfacing of the LED with the Arduino.

5.1. Interfacing of The LED With The Arduino

In this section, we will learn about the interfacing of the LED with the Arduino. We know that the LED has two pins anode and cathode.

5. LED Interfacing With The Arduino

In order to turn on the LED we will have to apply the voltage to the LED that will cross the cut off voltage of the material that is used in the design of the LED.

If you want to know the cut of voltage of the LED then you can check the following the table that I shared in the above section.

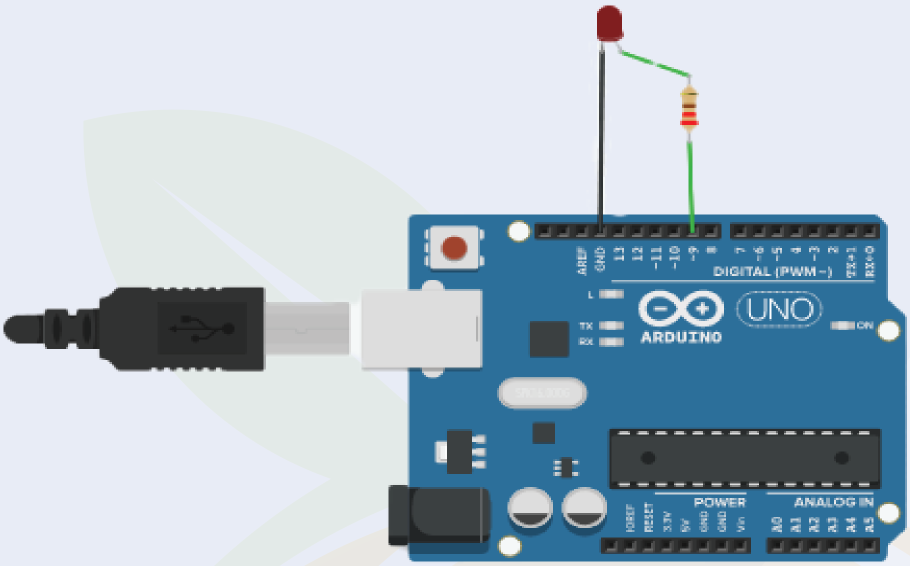
In that table, I have shared the name of the material and the cut-off voltage of the material. You refer to that sections.

So, let's assume that the cut-off voltage of your LED is 3.3V. In that case, if we connect LED directly to the GPIO pin of the Arduino, then we would damage the LED as the Arduino generates 5V output on its GPIO pin.

So, to avoid this, we will have to add resistor in between the GPIO pin of the Arduino and the LED.

In the below image, you can see we have added resister in between the GPIO pin and the LED.

5. LED Interfacing With The Arduino



Arduino Code For Controlling LED

There are two ways of controlling the LED. We will discuss about those function in the below section.

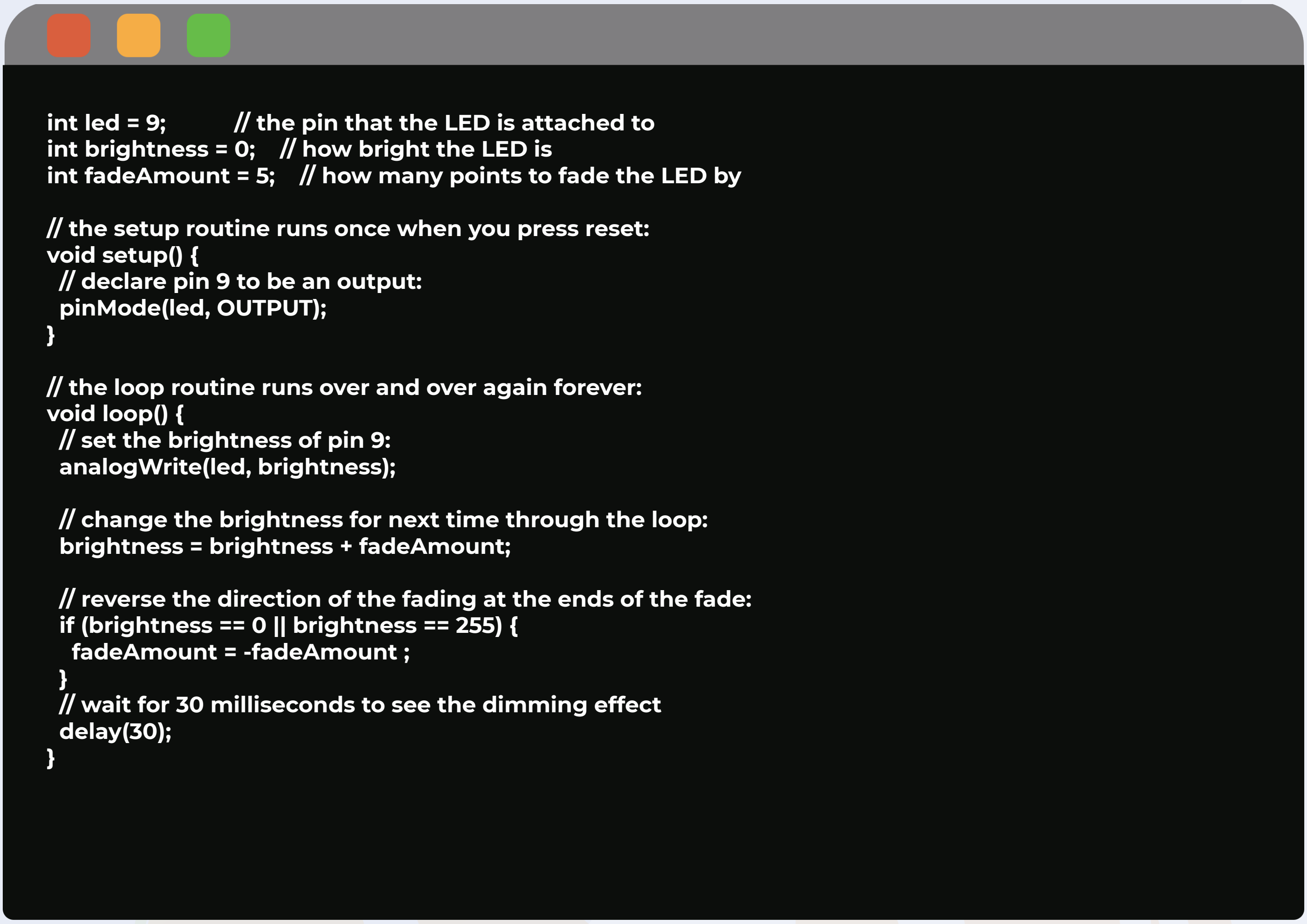
In this section, we are putting some voltage on the GPIO pin of the Arduino so we can use digitalWrite function of the Arduino.

This is the one method that we can use to control the LED. There is one more method and that is, controlling the LED using PWM method.

We can control the brightness of the LED using PWM method.

I have shared the snippets of both the codes below please take a look.

5. Arduino Code For Controlling The Brightness Of The LED



```
int led = 9;          // the pin that the LED is attached to
int brightness = 0;    // how bright the LED is
int fadeAmount = 5;    // how many points to fade the LED by

// the setup routine runs once when you press reset:
void setup() {
  // declare pin 9 to be an output:
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  // set the brightness of pin 9:
  analogWrite(led, brightness);

  // change the brightness for next time through the loop:
  brightness = brightness + fadeAmount;

  // reverse the direction of the fading at the ends of the fade:
  if (brightness == 0 || brightness == 255) {
    fadeAmount = -fadeAmount ;
  }
  // wait for 30 milliseconds to see the dimming effect
  delay(30);
}
```

6. Switch

Switches plays an important role in electronic and electrical industry. There are different types of switches are available in the market.
The switches that are used in the home appliances are known rocker switches.

The switch that you are getting with this Arduino starter kit is Push to on swtich type switch.
When we press the switch, the switch turns on and the signal connected to one end of the switch gets transferred to the other end of the switch.

6. Switch

So, this was about the instruction to the switches, in the next section we will learn about the interfacing of the switch with the Arduino.

6.1. Interfacing of The Switch with The Arduino

The switch has two pins as shown in the following image. We connect the signal pin to one end of the switch and the other end of the switch to the device where the signal needs to be transferred.

In this example, we have connected one end of the switch to 5v and the other to the 12-number pin of the Arduino.
When we are pressing the button the 5v supply of the Arduino is getting connected to 12-number of the pin.

This is one method of working with the switch. Another method of working with the switch is we can connect the one end of the switch to the GND pin of the Arduino and the other end to the 12-number pin of the Arduino.
If we are using the second method then we will have to use different programming method.
I have discussed that method in the following section of the blog.

6.2 Arduino Code For Switch

We now know the method of interfacing the switch with the Arduino, now we will talk about the programming part of this.

Important Note – If you are a beginner and really want to learn programming then you should always follow the following steps to be a best programmer.

Step 1 – Study the requirement of the project.

Step 2 – Gather the requirements that are needed to complete the project.

Step 3 – Understand the problems and design the codin]g cases

Step 4 – Break those cases into small pieces and start working on them.

These are the very important steps you should always follow if you want to be a best programmer.

So, in this case, we are reading the state of the switch. When someone is pressing the button, we are getting high level signal on the where the other end of the switch is connected.

In the following code, we are reading the output of the sensor and storing that sensor into a variable.

We have created one more variable in our code, to that variable we are saving the previous state of the switch.

In order to identify, whether the switch is pressed or not we are comparing the values of the store variable and the variable that is saving the previous value of the switch.

Do you know who invented the first switch? Well, The first light switch was invented by John Henry Holmes in 1884. The quick break technology used by Holmes is still found in almost every electric appliances.

7 LDR (Light Dependent Switch)

Light Dependent Resistor are also known as photoresistor they are widely used in electronic sectors.

They are easily available is low prices so it becomes most preferable choice to the product selection engineer.

Although similar components such as photo-diode, phototransistor are also available in the market and used for the same applications but the low-cost feature of the LDR makes the deal to take away.

Applications of the LDR

LDR are used in many applications. The stud light that we see on streets that has LDR in it. The purpose of use an LDR in those stud lights is they are used to detect the day light.

Another application of the LDR is, they are used in the IR video recording cameras. The IR video recording cameras works in two mode.

- **Day light Mode**
- **Night Light Mode**

In first mode, they work similar to the normal camera but in second mode they turn on the IR lights to record the videos and to switch between these two modes the IR video recording cameras make use of LDR.

So, these are the applications of the LDR, in the next section of this blog we will learn about the interfacing of the LDR with the Arduino.

7.1. Interfacing The LDR With The Arduino

Generally speaking, the LDR is nothing but a resistor. It has two pins and we can connect the LDR to the Arduino in a same fashion we connect the resistor to the Arduino.

In the following image you can see we have connected the one terminal of the LDR to the 5v pin of the Arduino and other end of the pin to the 7-number pin of the Arduino.

Please refer to the following image to understand the interfacing image of the LDR with Arduino.

7.2. Arduino Code LDR

In this section, we will learn about the programming part of this section.

As I told you earlier, the interfacing the LDR is nothing but a switch which resistance changes according to change in the resistance of the light.

So, we have to monitor that change thorough the Arduino and to do that we will have to use the following methodology.

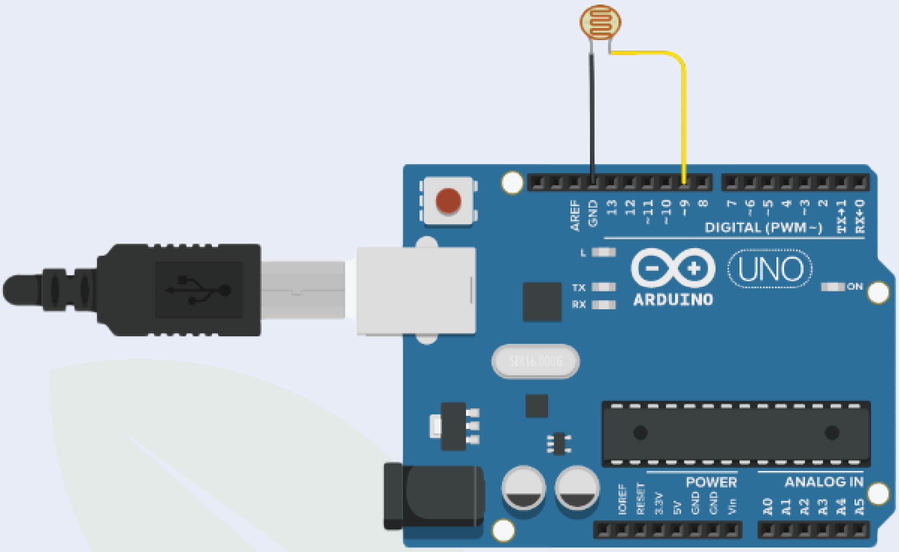
Here, when the intensity of light is increasing the resistance of the LDR is also changing. So to monitor that change, we will have read the output of the sensor and to read the output of the sensor we have used a variable.

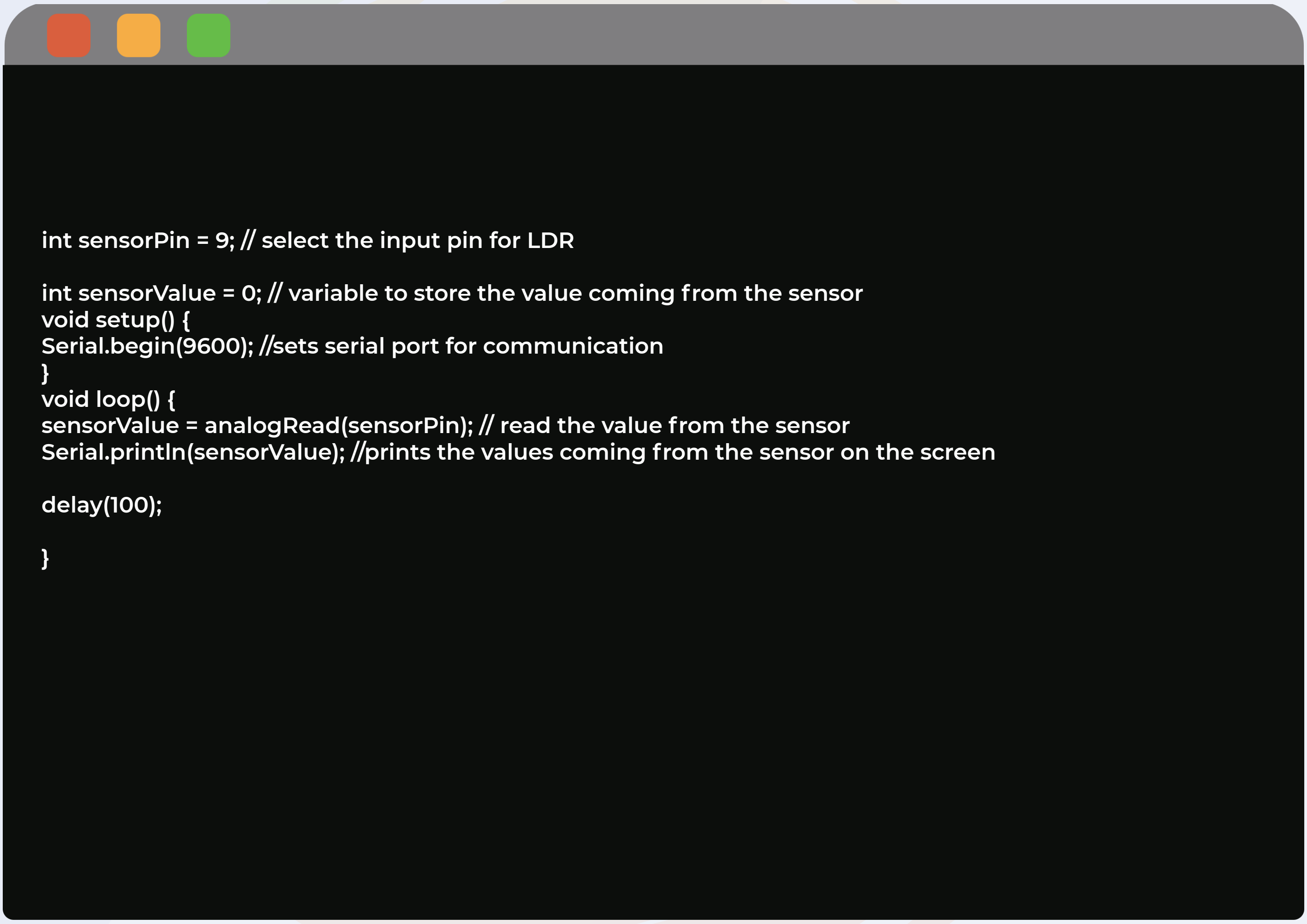
That variable is storing the output of the sensor and then we are printing the value of that sensor on the serial monitor.

You can use the below code and can see the output of the sensor on the serial monitor.

The LDR was designed in 1850 by the great electrical engineer Willoughby Smith.

7. LDR (Light Dependent Resistor)





```
int sensorPin = 9; // select the input pin for LDR

int sensorValue = 0; // variable to store the value coming from the sensor
void setup() {
Serial.begin(9600); //sets serial port for communication
}
void loop() {
sensorValue = analogRead(sensorPin); // read the value from the sensor
Serial.println(sensorValue); //prints the values coming from the sensor on the screen

delay(100);

}
```

P1) Arduino Blink Project

It is a very basic project specially designed for Arduino beginners. To complete this project, you will have to use the following components.

Parts required for the Arduino Blink Project

- LED
- Arduino
- Jumper Cables
- Breadboard

How to test the components?

Before you start, you should know the basics of all these sensors. If you do not know the basics of these components then you will not be able to control these components through Arduino.

So, I request you read about the components used in this kit.

In this section of this blog, we will do the testing of the components before using them.

LED Testing

As I told you earlier, the LED is a light-emitting diode and to tests the functionality of the LED. First, you have to connect the right terminal to the right pin of the Arduino.

P2) Arduino Blink Project

If you check the pins of the LED, you will see a difference. One pin of the LED is smaller than the other pin. You may be thinking that this may be a manufacturing defect, but it is not.

The pin which is shorter in length is the anode pin of that LED and the other pin is the cathode pin.

So, to test the LED, you have to connect the anode pin of the LED to the 5V pin of the Arduino and the other pin to the GND pin of the Arduino.

There is one more point I forgot to mention, and that is that you have to connect 320ohm or 200ohm between the anode pin of the LED and the 5V pin of the Arduino.

OK, Now, please verify all connections and see if the LED is blinking. If not, congratulations, you have successfully entered the world of electronics by blowing out the first component.

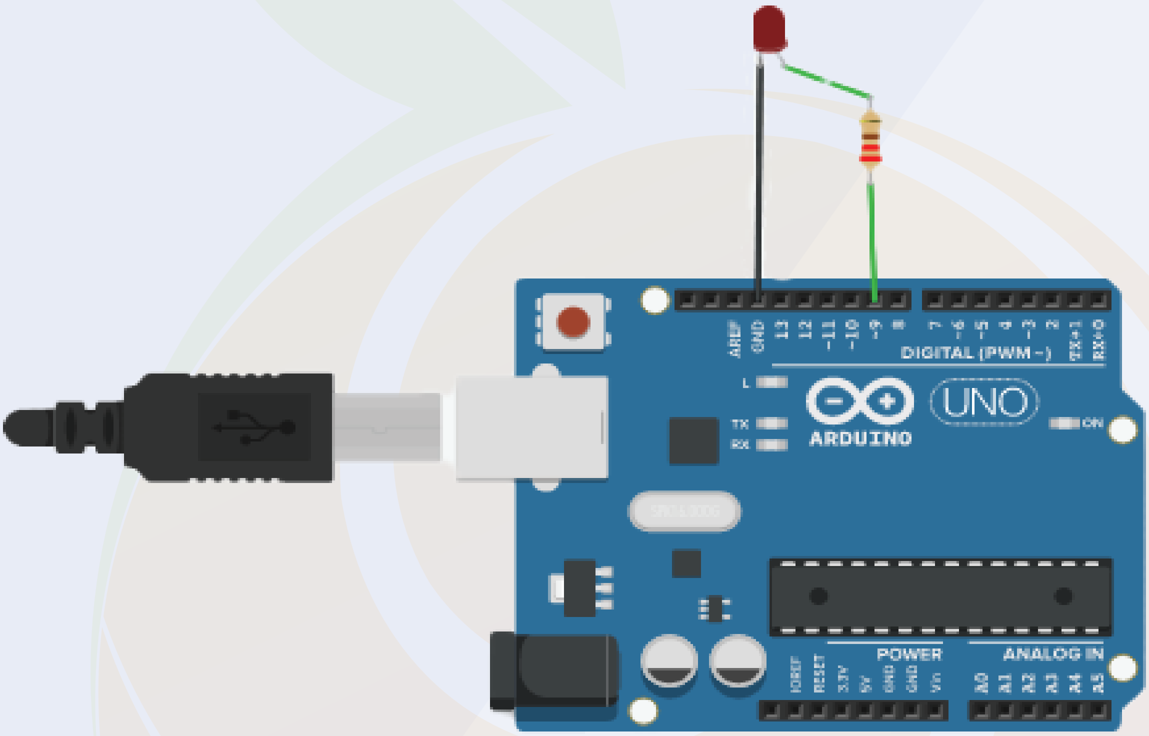
Jokes apart, if the LED is not flashing, it means that you have not made the connection properly. So please double-check the connection, and if the problem persists, please try using another LED.

As you have now completed the LED test and you know that your LED is in working condition. We will proceed to the next part of this blog.

P2) Arduino Blink Project

When testing the LED, we were connecting the anode terminal of the LED to the vcc terminal of the Arduino. But this time, we will connect the LED's anode terminal to the Arduino's 3 GPIO pins.

Please refer to the following image to understand the interfacing diagram.



Arduino Code for LED Blinking Project

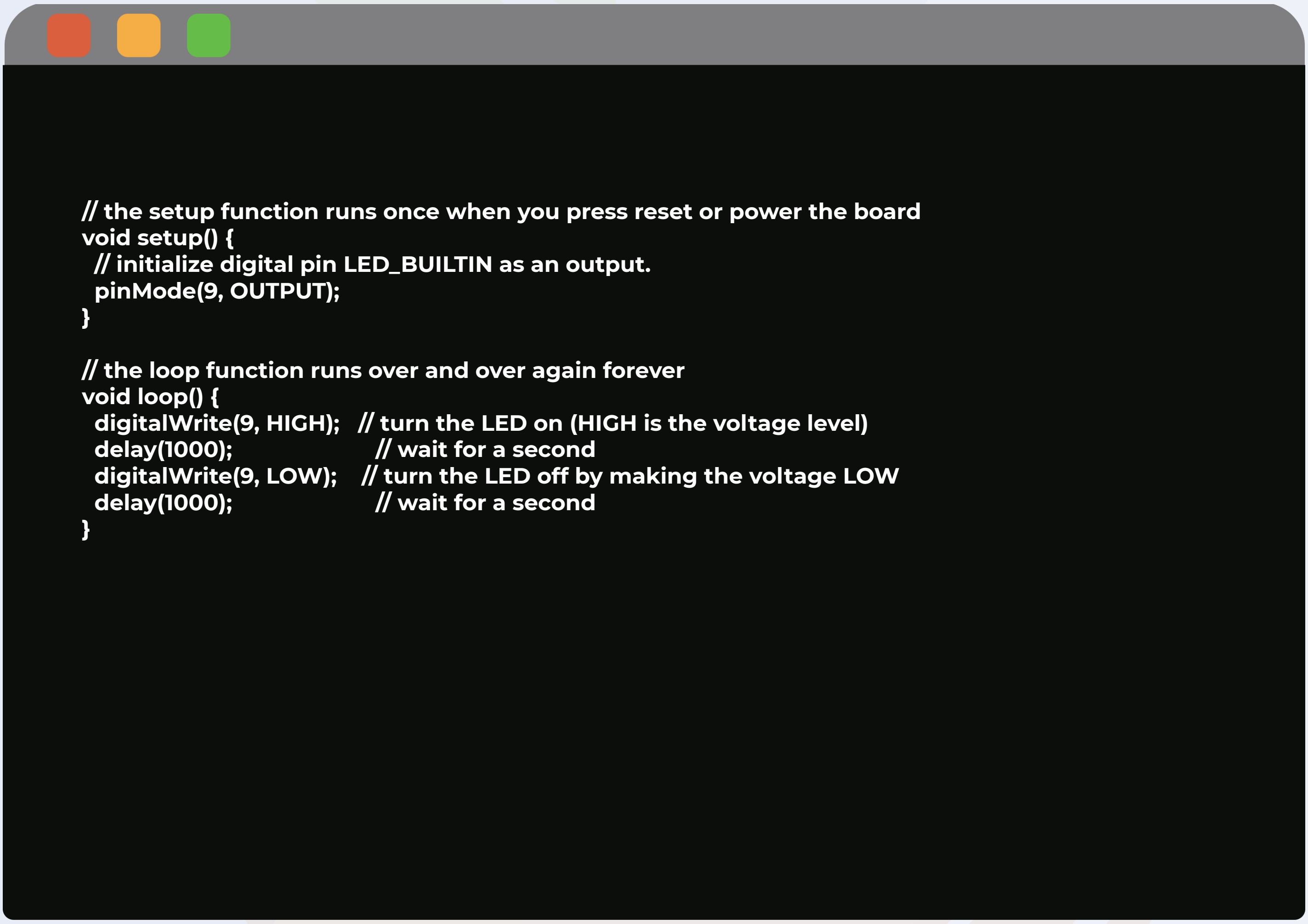
Please use the following Arduino code.

Arduino Code for LED Blinking Project

digitalWrite(Pin, HIGH or LOW);

This method is used to turn the LED on or off. It is the inbuilt function of the Arduino IDE and cannot be changed.

This function also takes two parameters. One is the pin number and the other is the output type.



```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(9, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(9, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);                       // wait for a second
  digitalWrite(9, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);                       // wait for a second
}
```

Arduino LED brightness control Project

In our previous section, we learned how to interface LED with Arduino and learned to blink Arduino. In this section, we will go a little further and learn how to control the brightness of the LED using PWM technology.

P3) Arduino LED brightness control Project

You do not need to change the connection for this project but you can change the connection if you want. If you have made up your mind to change the connection then make sure that you will change it in your code also.

Talking about this project, the basic information you should have while working on this project is controlling the brightness of the LED, which means that we are controlling the input voltage to the LED.

Components you may require for this project

- LED
- Arduino
- Jumper Cables
- Breadboard

Arduino Code for LED Brightness Project

As previously discussed, we are controlling the brightness of the LED, which means that we are controlling the input voltage in the LED.

In our previous project we used the digitalWrite method to control the LED, but in this case of the project, we cannot use the digitalWrite function because this function can put 5V or 0v on its GPIO pin.

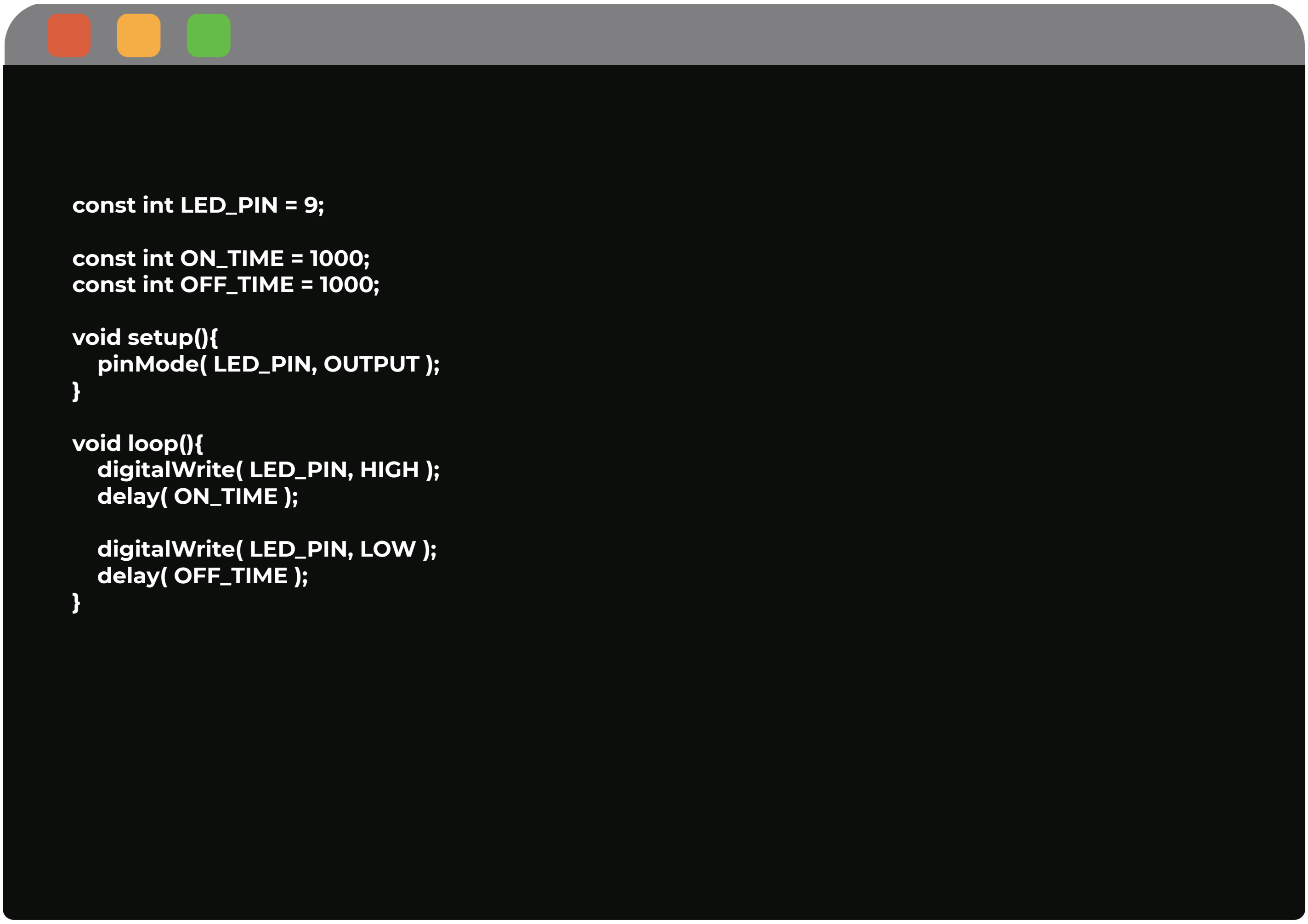
P3) Arduino LED brightness control Project

Which means that we cannot use the digitalWrite function to control the brightness of the LED.

We have to use the AnalogWrite function. If you have not heard of this function before then you can check more about that function here.

In the following code we have used the analog function and the For Loop. You can learn more about both of these terms here.

You can use the following code to control the brightness of the LED. In the following code I have used for loop but if you are good at coding or learning different methods of how to code then you can use your logic.



```
const int LED_PIN = 9;

const int ON_TIME = 1000;
const int OFF_TIME = 1000;

void setup(){
   pinMode( LED_PIN, OUTPUT );
}

void loop(){
   digitalWrite( LED_PIN, HIGH );
   delay( ON_TIME );

   digitalWrite( LED_PIN, LOW );
   delay( OFF_TIME );
}
```

P4) Arduino Day-Night Tracker Project

This is another project that we can design using the components of this kit. It will be a little advance project, but don't worry, I will clear all your doubts.

To complete the project, you need the following components.

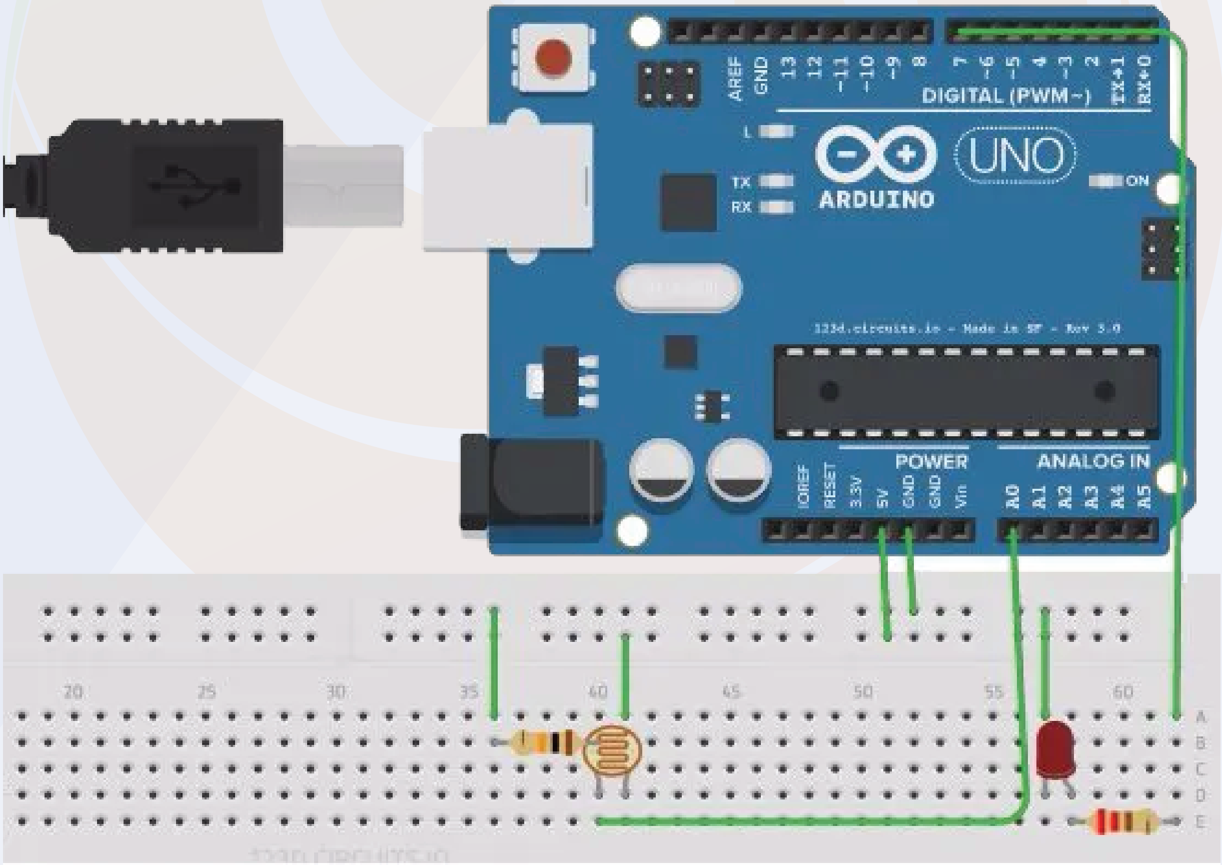
List of the components

LDR
Arduino
Jumper Cables
Breadboard
Buzzer

Interfacing Diagram for Arduino Day-Night Tracker Project

The following image shows the interfacing diagram for the day-night tracker project.

Please make the connection according to the diagram and if there is any doubt please let me know in the comment section.

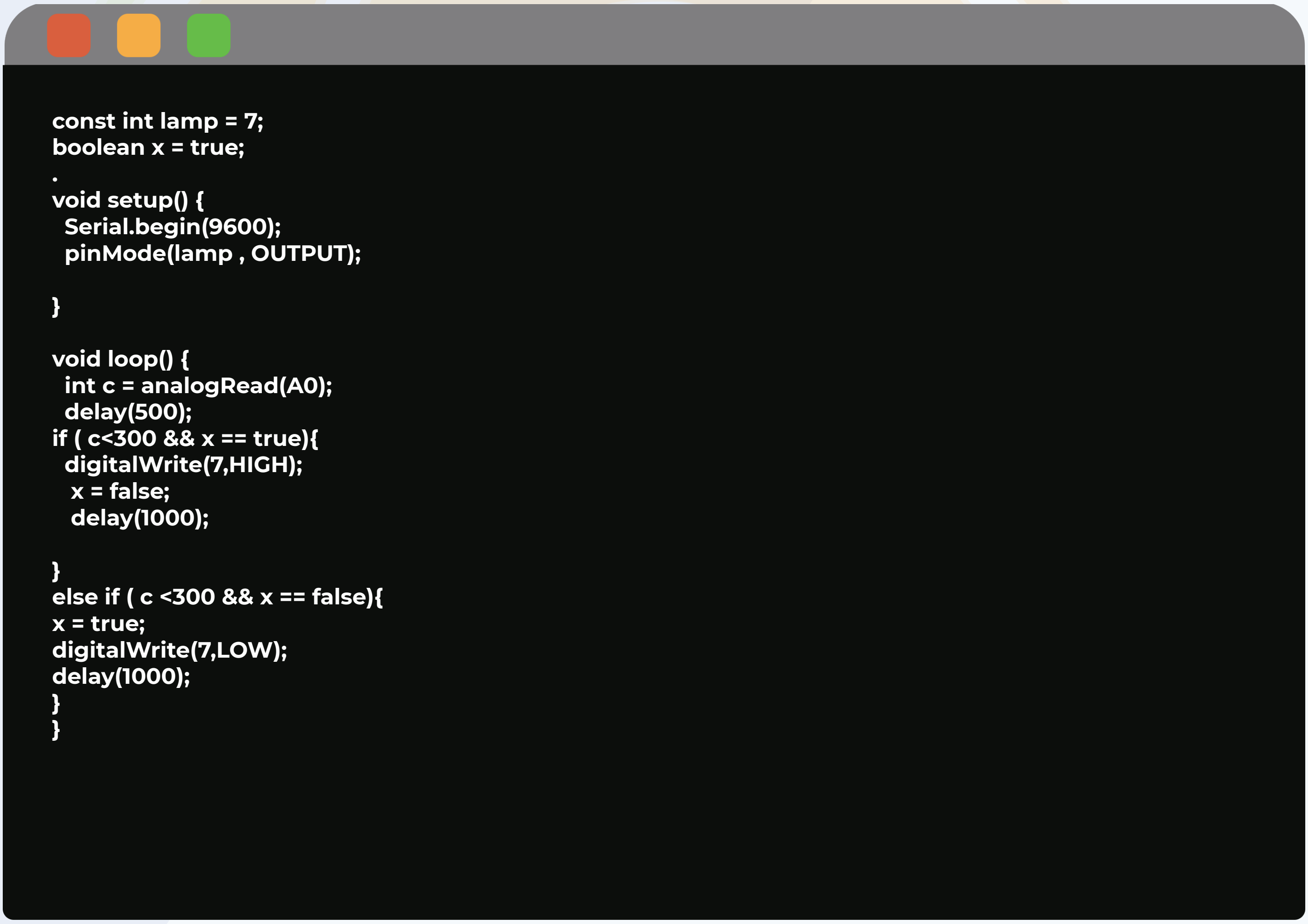


Arduino Code For Day-Night Tracker Project

The below code is designed for the Arduino Day-Night Tracker Project. In this project, we are using an analog read function. This function is used to read the amount of voltage generated by the LDR.

We are storing the output of LDR in a variable and when the value of that variable is crossing the specific value then we are turning on the LED.

I have created the following type of code using my logic but if you are using different logic then you can implement it.



```
const int lamp = 7;
boolean x = true;
.
void setup() {
  Serial.begin(9600);
  pinMode(lamp , OUTPUT);

}

void loop() {
  int c = analogRead(A0);
  delay(500);
if ( c<300 && x == true){
  digitalWrite(7,HIGH);
   x = false;
   delay(1000);

}
else if ( c <300 && x == false){
x = true;
digitalWrite(7,LOW);
delay(1000);
}
}
```

Thank You...!

A beginner at electronics systems comes across this common problem “How should I step into learning embedded systems? Which components should I buy? This is why Robu has designed the kits which will assist you to understand the embedded system from zero to hero level. This will solve the dilemma while selecting the proper products for your need.

Moreover, free e booklets and Video tutorials that are being shared with these kits have everything in it that's being taught within the paid lectures. That means you'll master those technical skills without paying an enormous amount to training institutes. So prepare to be the master of your dreams and start gaining the knowledge which is effective to you!

Happy Learning !!!