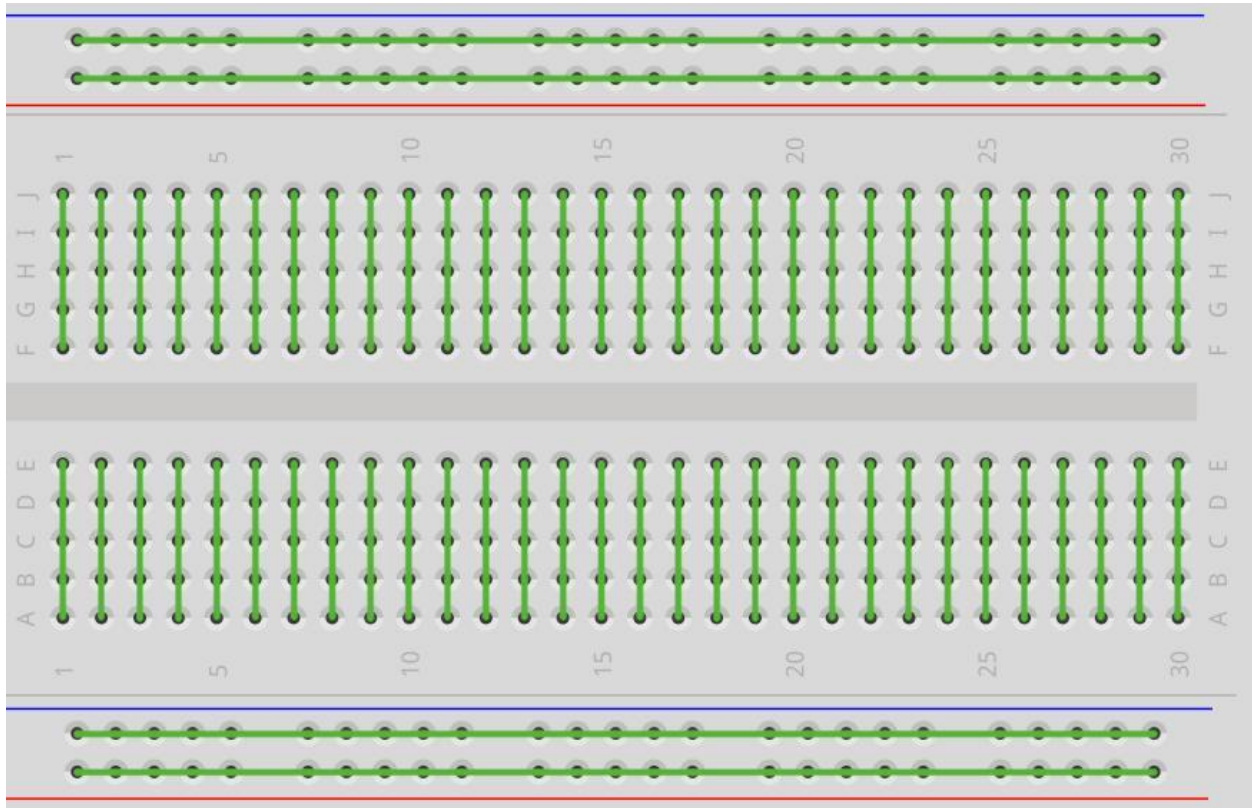


# Chapter 1

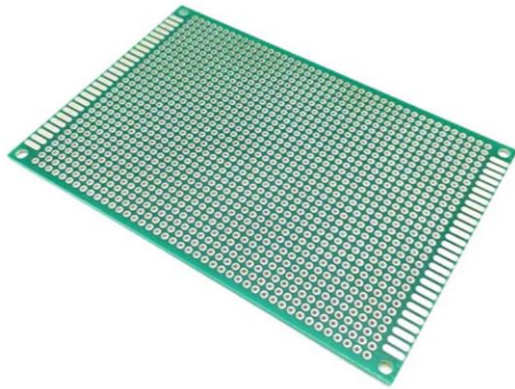
## Before we start with Arduino programming

### Familiarization with Breadboard:



**Fig 1.1: The internal connection of a medium size bread-board**

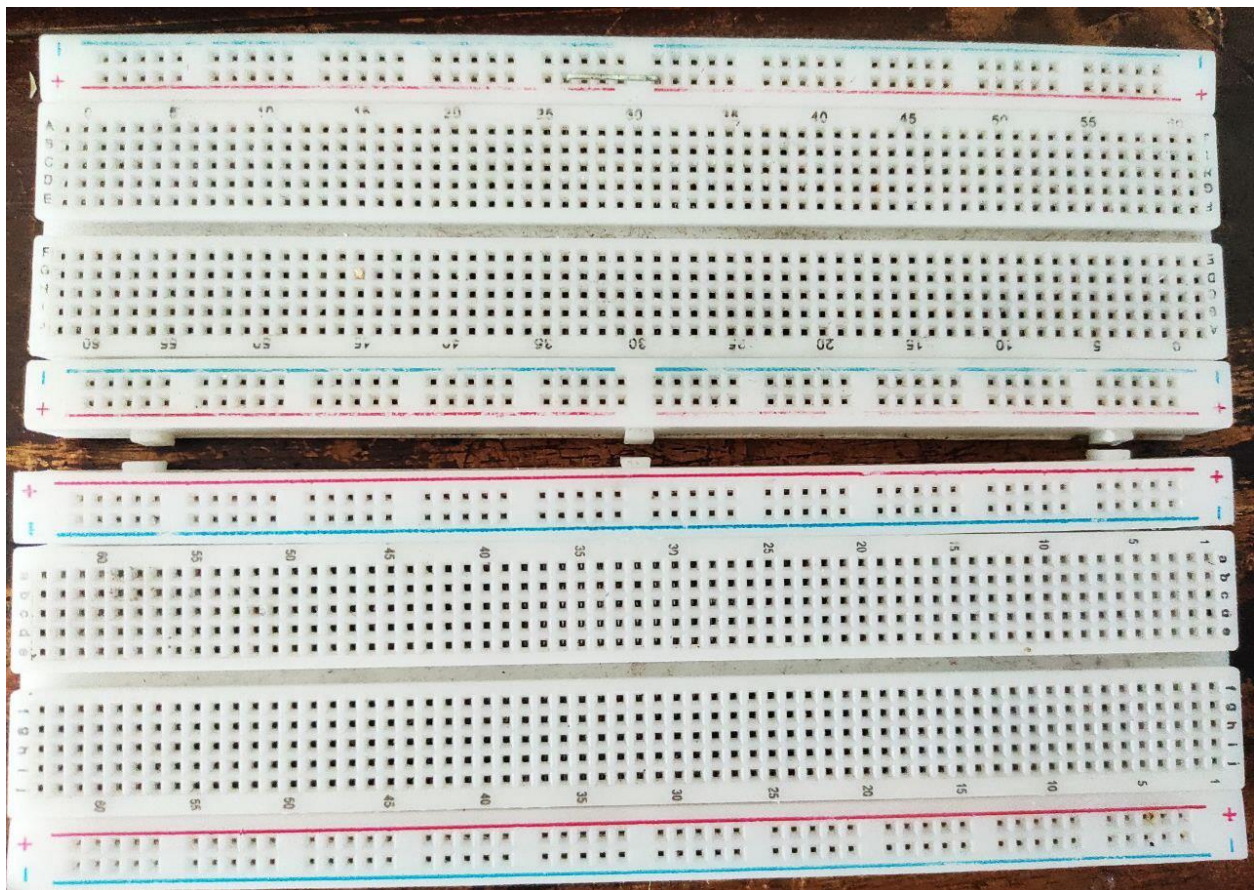
We only use bread board to for testing purpose. The final circuit is built on a board named Veroboard and it is soldered using soldering iron.



**Fig 1.2: A Veroboard**



**Fig 1.3: A Soldering Iron**



**Fig 1.2: Two different types of large bread-board**

## Familiarization with Resistors and it's colors:

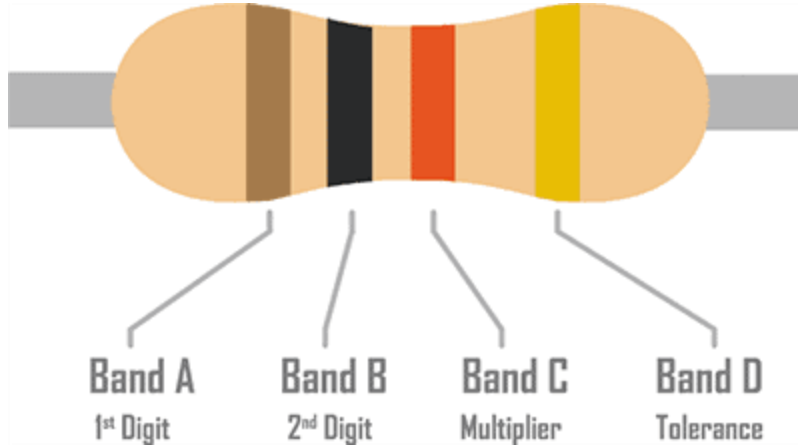


Fig 1.3: A 4 color (band) resistor

## A website to calculate the resistance of a resistor from its color:

<https://www.digikey.com/en/resources/conversion-calculators/conversion-calculator-resistor-color-code>

## Familiarization with Arduino UNO:

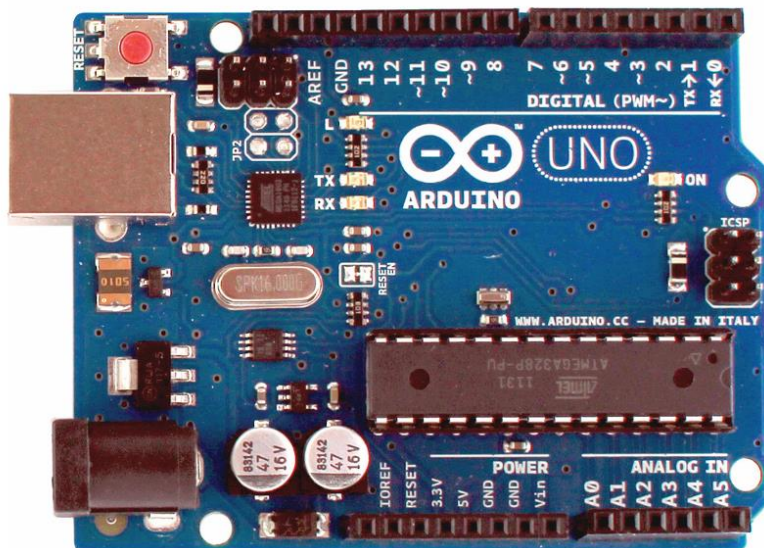


Fig : The front side of an Arduino UNO





**Fig : The brain of an Arduino UNO (ATMEL ATMEGA328 Chip)**

This is called a microcontroller.

Microcontroller = Processor + RAM + ROM + INPUT/OUTPUT + etc

Arduino UNO has:

1. Atmega328 brain that runs with 16 MHz clock speed.
2. RAM: 2 KB
3. Memory: 32 KB
4. 14 digital I/O pins
5. 6 analog input pins.
6. And many more feature

Note: Just to mention there are many more variants of Arduino in Bangladesh Market than Arduino UNO. Each of them has some specific

benefit. Some honorable mention: Arduino nano, Arduino Mega, Arduino Pro mini, ESP32, STM32, etc.

# **Chapter 2**

## **Setting up the PC for Arduino Programming**

### **Download the Arduino IDE:**

1. Download the Arduino IDE from <https://www.arduino.cc/en/software/> and install it on your own. We know you can do it.
2. If you face any problem look for the tutorial video already uploaded to google classroom.
3. Still if you are facing problem, ask in our messenger group.

### **Ensure the Arduino is OK:**

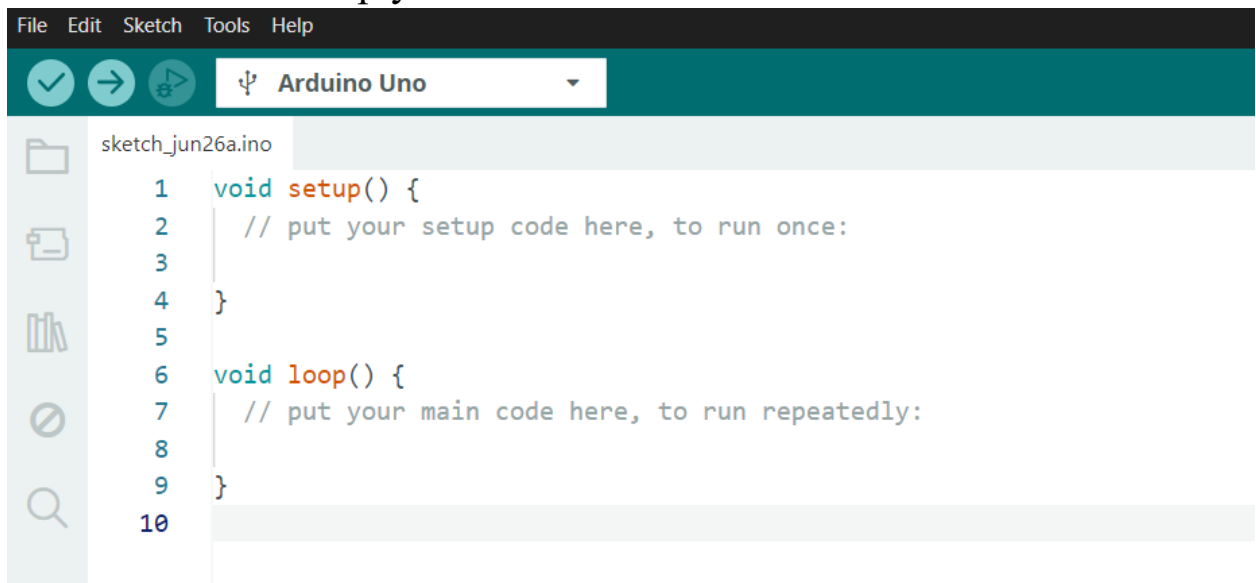
1. Connect the cable to Arduino UNO and to your PC. You should hear a sound from the PC when you plug in the cable. If you don't hear any sound try plugging again.
2. Open your device manager. You should see Arduino UNO or something like this under ports (COM & LPT). (Arduino must be connected through cable to PC).

If things are all right, you are ready to go. If you are facing any problem don't hesitate to ask in our messenger group.

# Chapter 3

## Our First Arduino Program

1. Connect the Arduino UNO to the PC using the Cable
2. Open the Arduino IDE software.
3. You should see a empty code like this below



```
File Edit Sketch Tools Help
sketch_jun26a.ino
1 void setup() {
2   // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7   // put your main code here, to run repeatedly:
8
9 }
10
```

4. Now write this code below:



```
File Edit Sketch Tools Help
sketch_jun26a.ino
1 void setup() {
2   pinMode(13, OUTPUT);
3
4 }
5
6 void loop() {
7   digitalWrite(13, HIGH);
8   delay(1000);
9   digitalWrite(13, LOW);
10  delay(1000);
11
12 }
```

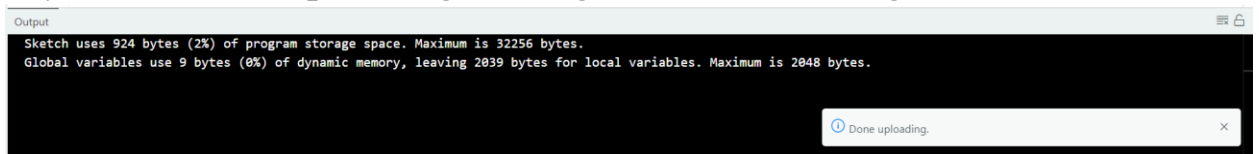
5. Select Arduino UNO and the port.



6. Click the upload button at upper left corner.



7. If you see Done uploading message below then congratulations.

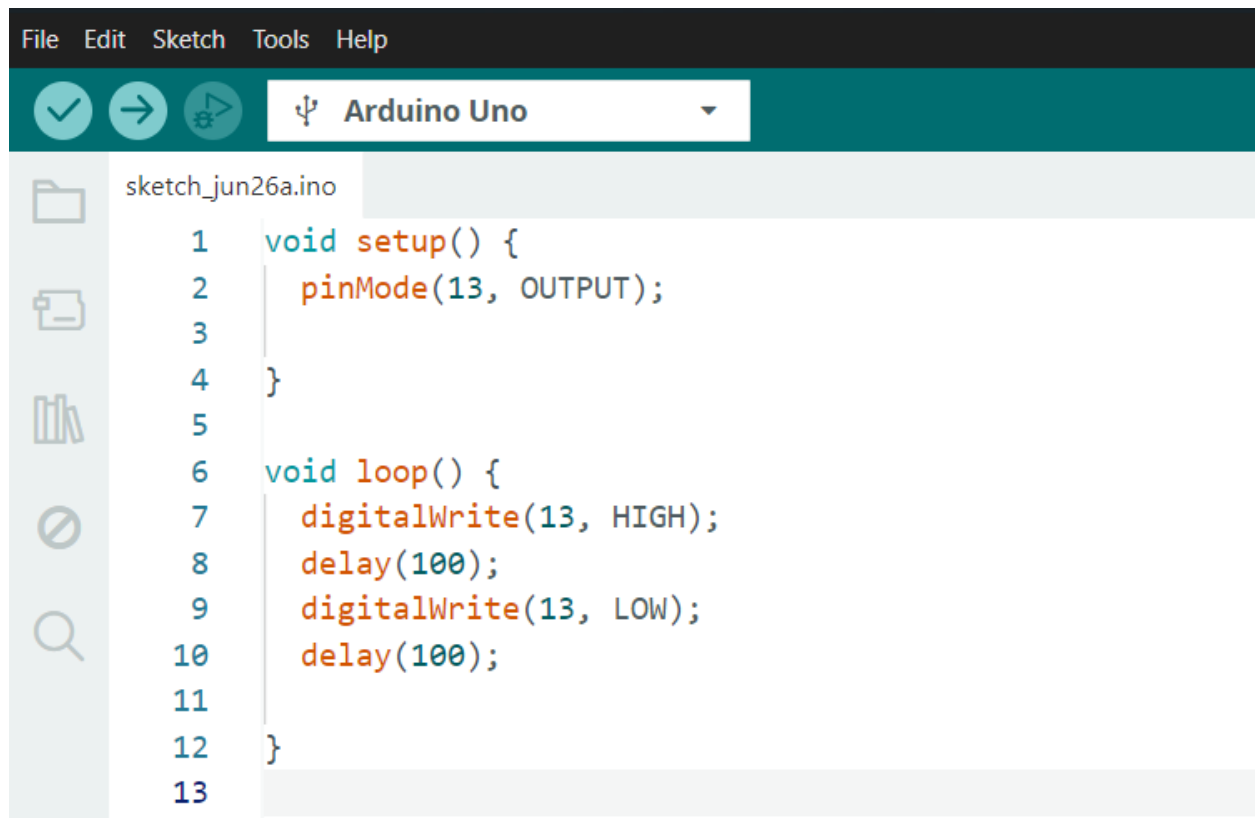


(If you faced any problem, it is totally fine. Just try again from 1-7.  
If the problem is still there, ask in our messenger group.)

An LED should be ON and OFF on your Arduino UNO. Hurrah!!!!  
Look at this for 5 minutes and think how beautiful a LED can be.

Let's understand the code later. Just write the below code now and  
upload it to Arduino UNO using that same upload button.





```
1 void setup() {
2   pinMode(13, OUTPUT);
3
4 }
5
6 void loop() {
7   digitalWrite(13, HIGH);
8   delay(100);
9   digitalWrite(13, LOW);
10  delay(100);
11
12 }
13
```

(100 instead of 1000)

What did you notice?

The LED is blinking faster. WOW!!!! Waste 5 more minutes by looking at the LED.

**Now Let's understand the whole code:**

The basic structure of an Arduino code is this:



What you write in the void setup() {} will happen for only once.

What you write in the void loop() {} will happen for forever.

The rest of the code means:

```
1 void setup() {
2   pinMode(13, OUTPUT); // It means pin 13 will be used as an OUTPUT pin.
3
4 }
5
6 void loop() {
7   digitalWrite(13, HIGH); // Turn on the LED (HIGH voltage to pin 13)
8   delay(100);             // wait for 100 milliseconds
9   digitalWrite(13, LOW);  // Turn off the LED (LOW voltage to pin 13)
10  delay(100);             // wait for 100 milliseconds
11
12 }
```

(Lines starting with // (double slash) are comments and they do nothing in the code. Just help you to understand. It is a really good habit to write

comments with code for many reasons. You will understand the reasons on your own with time)

I know everything looks complex. But take don't think too much. Let's play a game.

Upload this below code to Arduino UNO:



The screenshot shows the Arduino IDE interface. The top menu bar includes 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. Below the menu bar is a toolbar with icons for checking, running, and uploading, followed by a dropdown menu set to 'Arduino Uno'. The main workspace displays a sketch named 'sketch\_jun26a.ino' with the following code:

```
1 void setup() {  
2   pinMode(13, OUTPUT);  
3  
4 }  
5  
6 void loop() {  
7   digitalWrite(13, HIGH);  
8   delay(70);  
9   digitalWrite(13, LOW);  
10  delay(70);  
11  
12 }
```

What just happened?

Now upload this code below:



```
File Edit Sketch Tools Help
sketch_jun26a.ino
1 void setup() {
2   pinMode(13, OUTPUT);
3
4 }
5
6 void loop() {
7   digitalWrite(13, HIGH);
8   delay(50);
9   digitalWrite(13, LOW);
10  delay(50);
11
12 }
```

Did you see any change after uploading this upper code?

Now continue reducing the value inside delay. 40, 30, 25, 20, 15, 14, 13, 12, 11, 10. What do you see now?

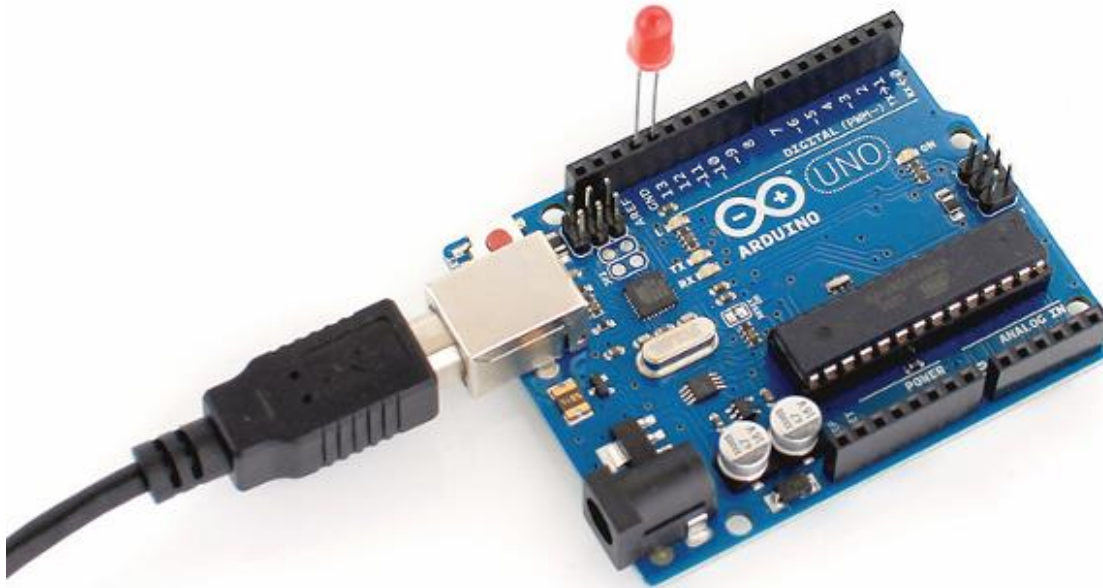
You know longer see the LED blinking. Instead, what you see is the LED is permanently ON though the LED is turning ON and OFF very quickly. This phenomenon is called persistence of vision. The ON and OFF is so fast that your brain does not process the OFF state any more.

# Chapter 4

## Adding extra things with Arduino

Now, honestly speaking, we don't like the built-in LED on Arduino UNO. So, Let's add extra LED to it.

Make the Circuit below:



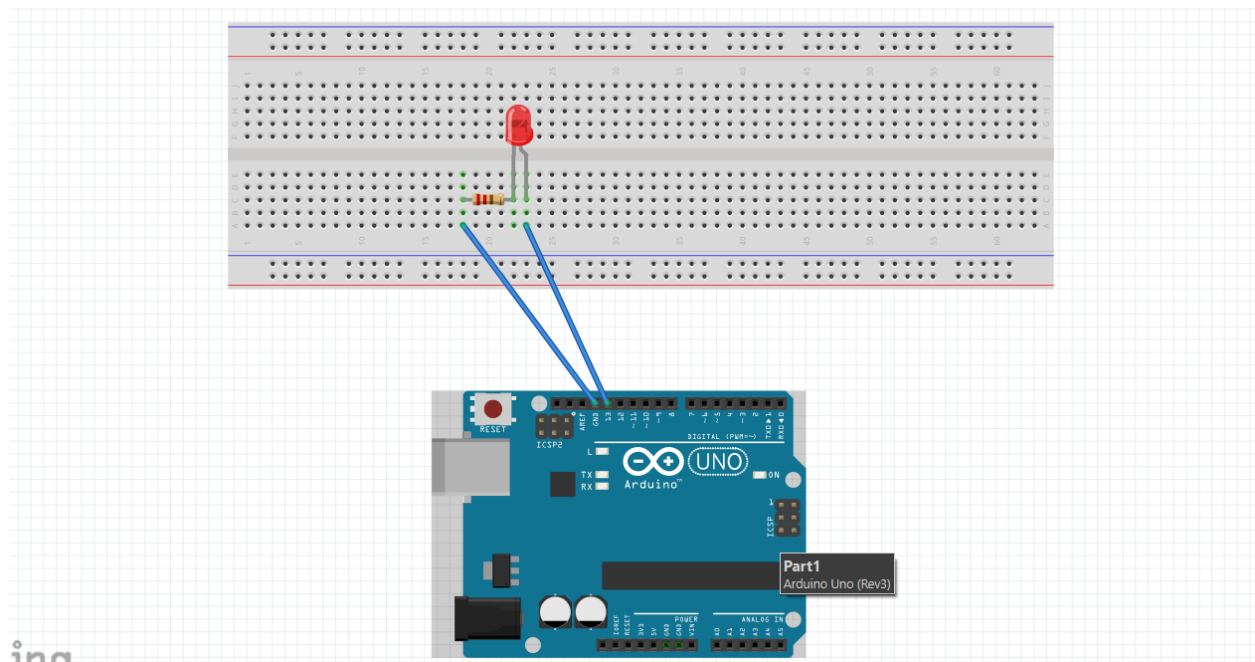
(The LED has two legs. The shorter leg goes to the GND pin And the longer leg goes to the 13 pin)

And Now Upload this code below to Arduino UNO:

```
File Edit Sketch Tools Help
Arduino Uno
sketch_jun26a.ino
1 void setup() {
2   pinMode(13, OUTPUT);
3
4 }
5
6 void loop() {
7   digitalWrite(13, HIGH);
8   delay(1000);
9   digitalWrite(13, LOW);
10  delay(1000);
11
12 }
```

Impressive, isn't it! But that is not how engineers build a circuit.

Instead, build the below circuit. Using jumper wire and bread board. Much better.

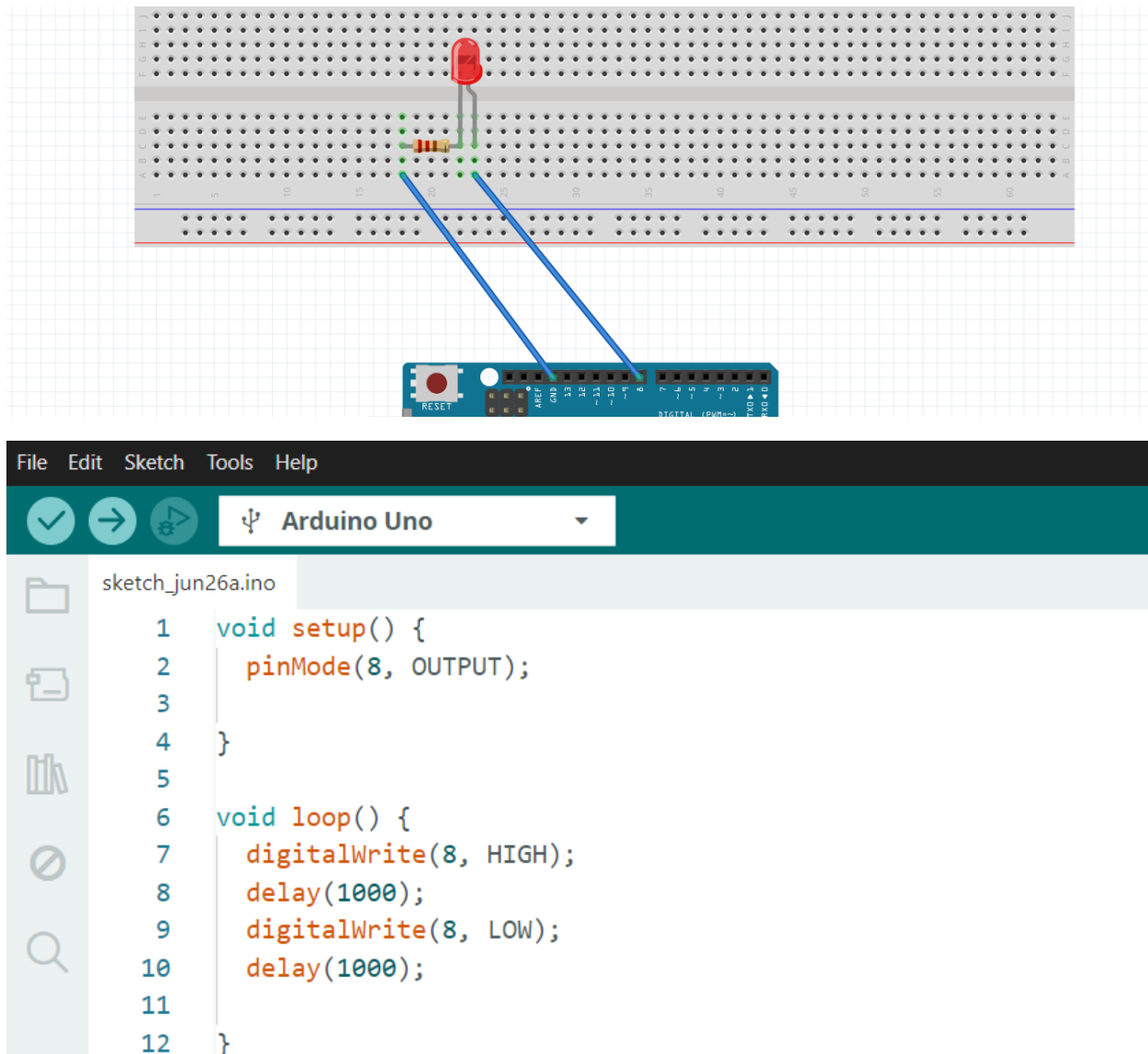




(That resistor is a 220 ohm resistor. You know how to find it by seeing the color.) The job of this resistor is to protect the LED from HIGH current.

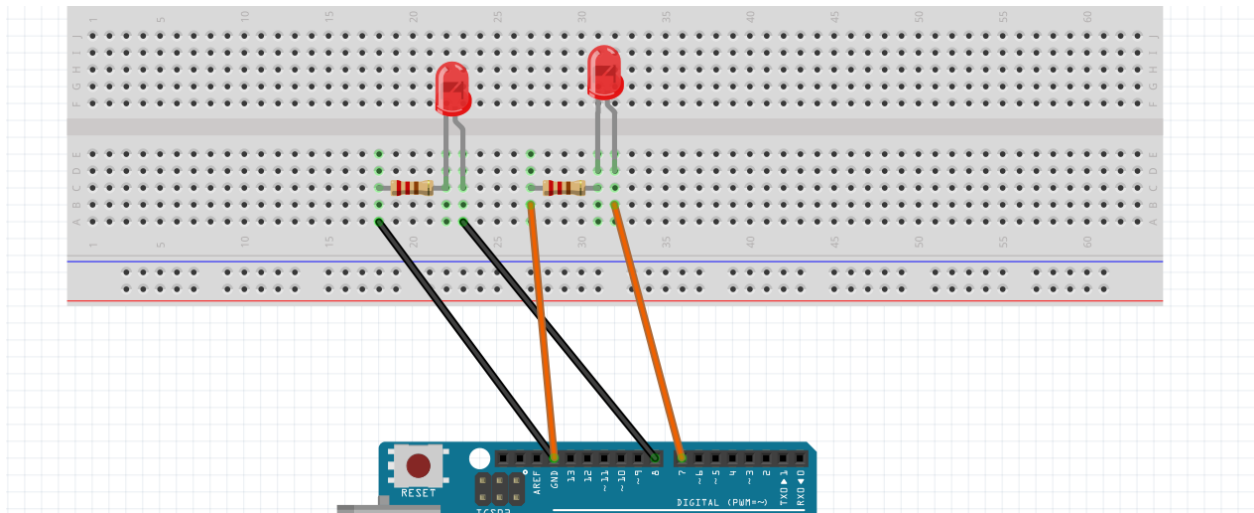
Enough of using pin 13. Let's use any other pin.

Build the circuit below and upload the below code.



Our pin 13 can take some rest now. Let the pin 8 do some work for you.

Let's use two LED with Arduino. Built the circuit below and upload the code. (Note: you cannot plug in two jumper wire into a same port in Arduino UNO. So how can we build this circuit below)



```
File Edit Sketch Tools Help
Arduino Uno
sketch_jun26a.ino
1 void setup() {
2   pinMode(8, OUTPUT);
3   pinMode(7, OUTPUT);
4 }
5
6
7 void loop() {
8   digitalWrite(8, HIGH);
9   digitalWrite(7, HIGH);
10  delay(1000);
11  digitalWrite(8, LOW);
12  digitalWrite(7, LOW);
13  delay(1000);
14 }
15 }
```

Hope things have started to make sense a little. Even if things do not make sense don't get frustrated. Starting is always the hardest part.

Upload the below code keeping the circuit same. (Two LED: one at pin 7 and one at pin 8)



```
File Edit Sketch Tools Help
sketch_jun26a.ino
1 void setup() {
2   pinMode(8, OUTPUT);
3   pinMode(7, OUTPUT);
4
5 }
6
7 void loop() {
8   digitalWrite(8, HIGH);
9   delay(1000);
10  digitalWrite(8, LOW);
11  delay(1000);
12  digitalWrite(7, HIGH);
13  delay(1000);
14  digitalWrite(7, LOW);
15  delay(1000);
16
17 }
```

What do you see? Try to understand how does the code controls the real world.

# Chapter 5

## The seven segment display

In previous part, you have learned how to control an LED using Arduino Code. The Seven Segment Display is just a combination of 8 LEDs. (7 LEDs for the digit and 1 LED for the dot).

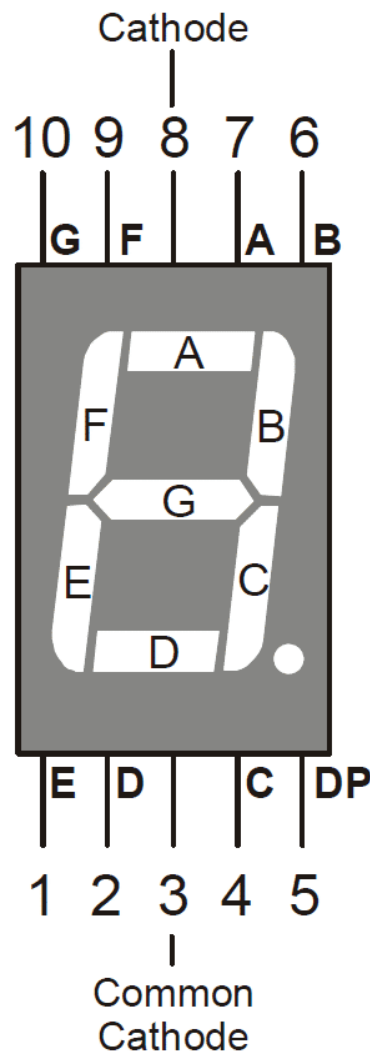
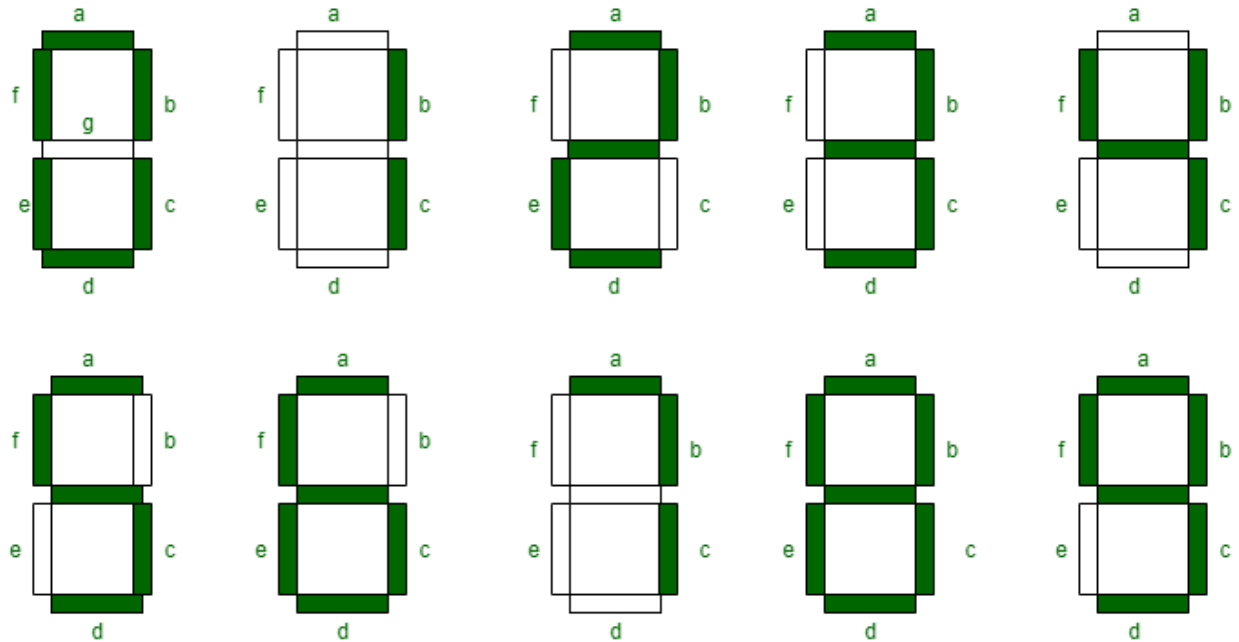


Fig: Pinout of a Seven Segment Display

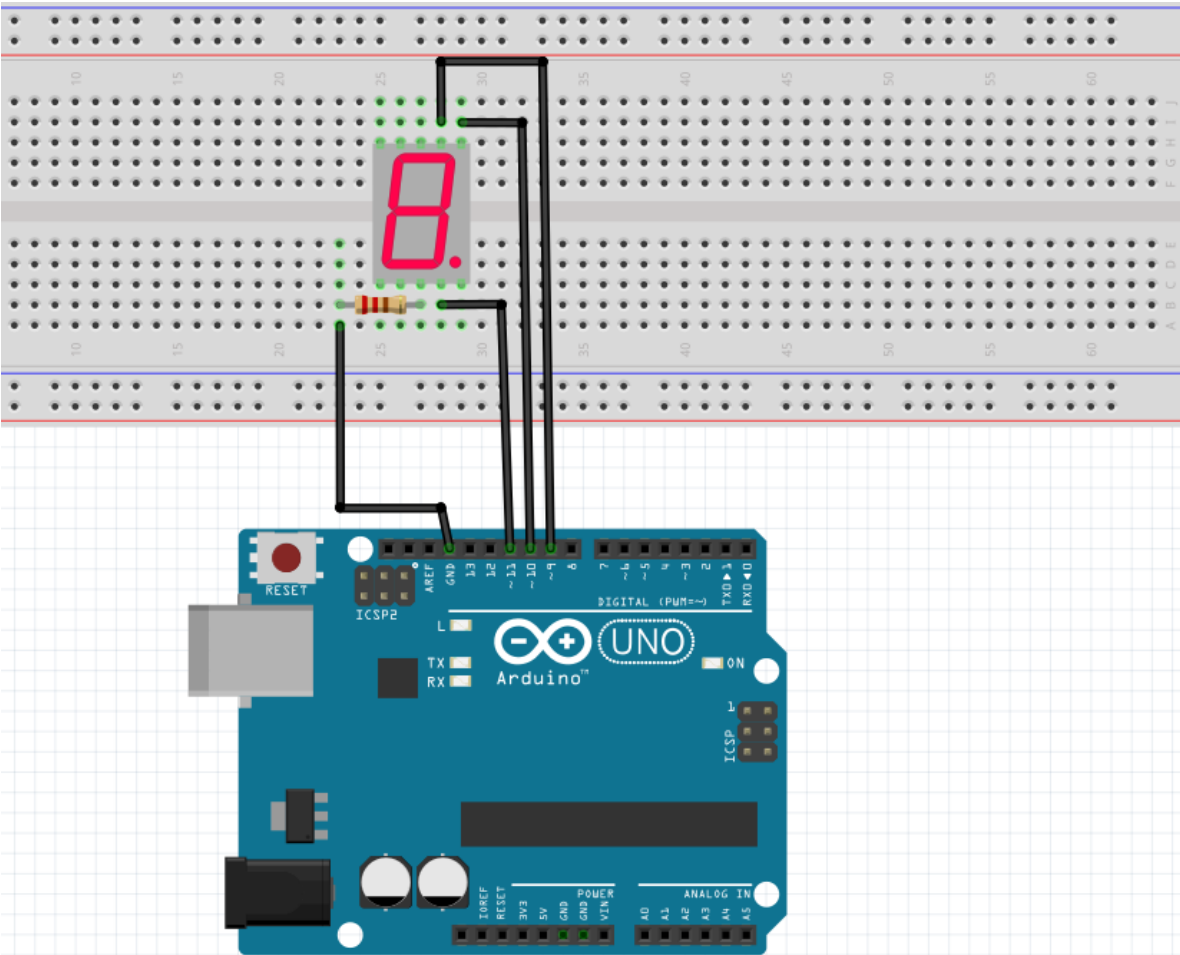
We can write different digits by just turning these LED ON and OFF.



Now, we will see how to write 7 on Seven Segment Display (SSD).

See that each LED has a name. And each pin is connected to a pin with the same name. To print 7 on this display we have to turn on a, b and c LED.

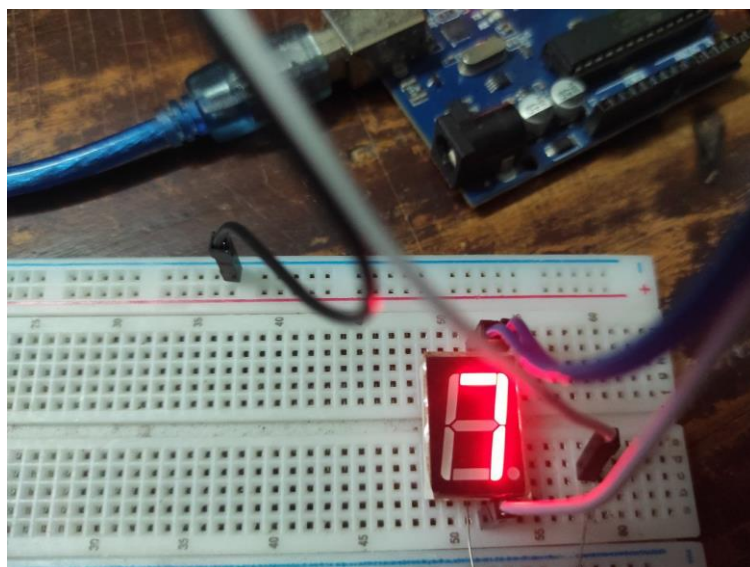
Build the circuit below and upload the code:





```
File Edit Sketch Tools Help
Arduino Uno
sketch_jun26a.ino
1 void setup() {
2   pinMode(9, OUTPUT);
3   pinMode(10, OUTPUT);
4   pinMode(11, OUTPUT);
5
6 }
7
8 void loop() {
9   digitalWrite(9, HIGH);
10  digitalWrite(10, HIGH);
11  digitalWrite(11, HIGH);
12  delay(1000);
13  digitalWrite(9, LOW);
14  digitalWrite(10, LOW);
15  digitalWrite(11, LOW);
16  delay(1000);
17
18 }
```

The output should be like this: (turns ON and OFF)



## Assignments

1. Try all the codes demonstrated in this whole PDF.
2. Find someone better than Arduino UNO. Describe why this microcontroller is better. (Anything available in BD online market. Just google it to see if it is available or not).
3. Find the limit in milliseconds for your perception of vision.
4. Write 0 on seven segment display.
5. Write 1 on SSD.
6. Write 2 on SSD.
7. Write 3 on SSD.
8. Write A on SSD. (Yes, you heard that right. Write A.)
9. Write C on SSD.
10. Write H on SSD.
11. Hard one: Display 0 then 1 then 2 then repeat 0, 1, 2....  
Details: show 0 then turn off all LED of SSD, then show 1 then turn off all the LED, then 2 then turn off all LED...thus
12. Display 4, 5, 6 then repeat on SSD.
13. Hardest one: Display your name on Seven Segment Display:  
Such as: A then B then C then D if your name is ABCD. (May not be possible for all characters of your name but give it a try.)
14. Stay happy. Don't let the toughness of assignments ruin your happiness. And most importantly, have fun.

(Note: Submit your assignments PDF, Video, Image, Text in google classroom: Any format that you like.)