

Experiment 2.1

Student Name:
Branch:
Semester:
Subject Name: BEEE

UID:
Section/Group:
Date of Performance:
Subject Code:

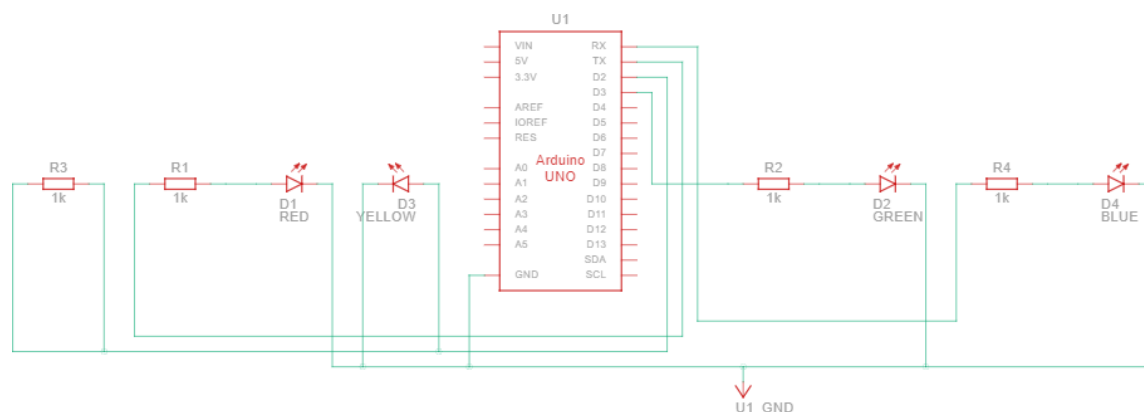
Aim:
To design LED flasher light.

Apparatus:



SR.NO.	EQUIPMENT	SPECIFICATIONS AND RANGE	QUANTITY
1.	Resistance	1 k Ω	3
2.	LED light	0- 2 V	3
3.	Breadboard	N/A	1
4.	Connecting wires	N/A	As per requirement
5.	Arduino uno	N/A	1

Circuit Diagram:



Steps for experiment:

1. Open Tinkercad website and create a new circuit. Drag and drop a breadboard to the console panel.
2. Now drag and drop a LED, resistor and Arduino Uno board. Place them in their respective positions.
3. Connect the anode of the LED to the 13th pin of the Arduino board through a resistor. Ground the cathode of the LED and the Arduino board.
4. Now write the code to get the desired result.
5. Theory
6. Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), 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. You can tinker with your Uno without worrying too much about doing something wrong, worst-case scenario you can replace the chip for a few dollars and start over again.
7. LEDs are a particular type of diode that convert electrical energy into light. In fact, LED stands for “Light Emitting Diode.”
8. With the help of these two and a few supporting components we can create a led flasher model. Now to control this chipset we need to write a code in Arduino and then we can control the glowing of the LED accordingly.

The first thing we need to do is configure as an output the pin connected to the LED. We do this with a call to the `pinMode()` function, inside of the sketch's `setup()` function: Finally, we have to turn the LED on and off with the sketch's `loop()` function. We do this with two calls to the `digitalWrite()` function, one with `HIGH` to turn the LED on and one with `LOW` to turn the LED off. If we simply alternated calls to these two functions, the LED would turn on and off too quickly for us to see, so we add two calls to `delay()` to slow things down. The `delay` function works with milliseconds, so we pass it 1000 to pause for a second.



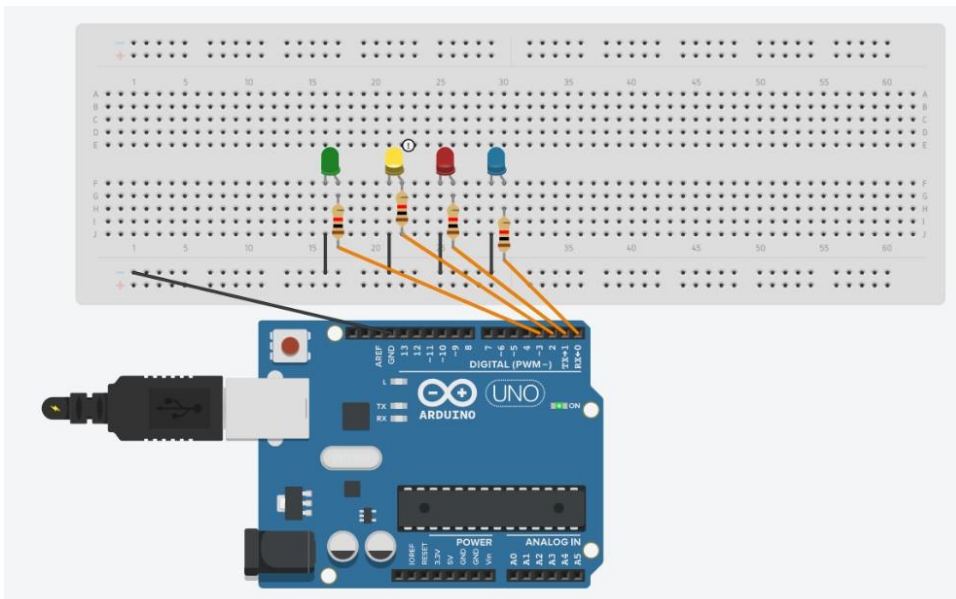
Code:

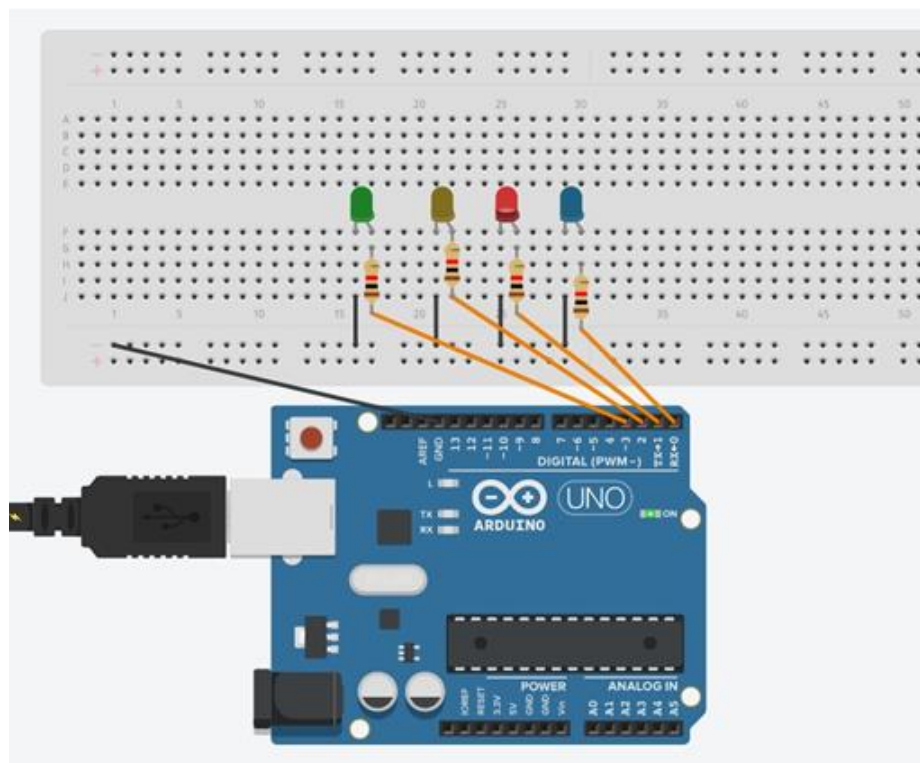
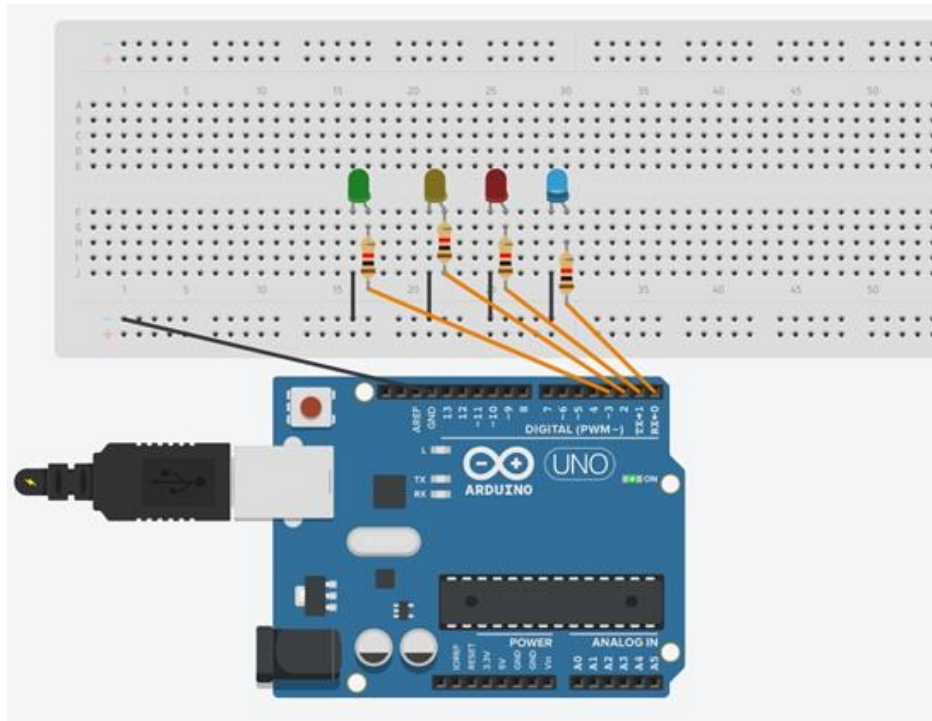
```
void setup()
{
  int i; for(i=0;i<=4;i++)
  {
    pinMode(i,OUTPUT);
  }
}

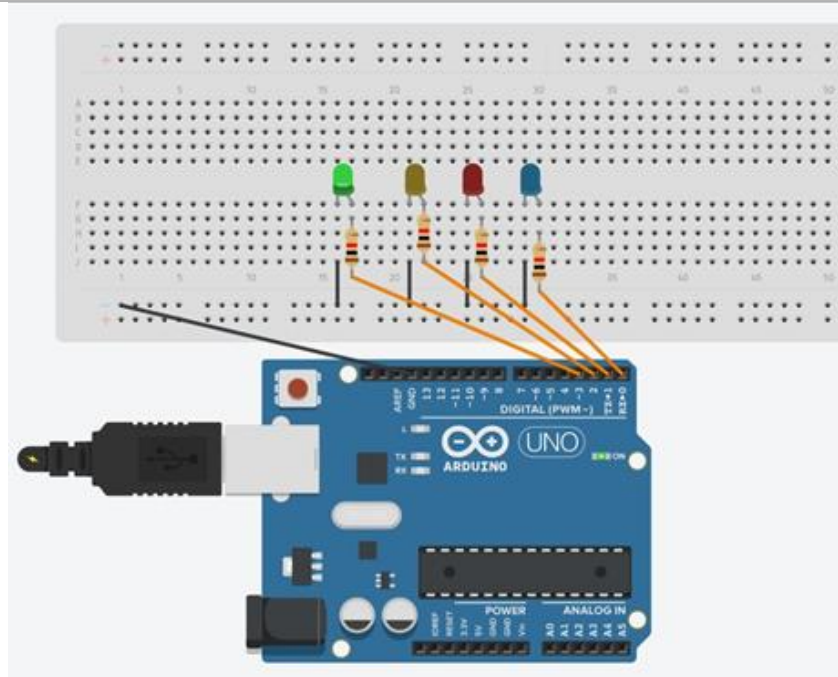
void loop()
{
  int i; for(i=0;i<=4;i++)
  {
    digitalWrite (i,HIGH);delay(100); digitalWrite (i,LOW);delay (20);
  }
}
```



Result/Output:







Sources of error:

1. Due to internal resistance of multimeter. Due to interruption of power supply.
2. Due to wrong connection of circuit.

Learning outcomes (What I have learnt):

1. Learned the application of Arduino Uno IC
2. Designed and learned how to how to control the led flasher
3. Design of circuit using Arduino
4. Verify the circuit by programming.



Evaluation Grid:

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.	Worksheet completion including writing learning objectives/Outcomes. (To be submitted at the end of the day).		10
2.	Post Lab Quiz Result.		5
3.	Student Engagement in Simulation/Demonstration/Performance and Controls/Pre-Lab Questions.		5
	Signature of Faculty (with Date):	Total Marks Obtained:	

