KARNATAK LAW SOCIETY'S

GOGTE INSTITUTE OF TECHNOLOGY

UDYAMBAG, BELAGAVI – 590008

(An Autonomous Institution under Visvesvaraya Technological University, Belagavi)

(Approved By AICTE, New Delhi)

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

COURSE PROJECT : INTERNET OF THINGS

Rachana Kampli	2GI18IS032
Laxmi Nyamagoud	2GI18IS020
Hemanth Tubachi	2GI18IS015
Rohan Kokatanur	2GI18IS066

Guided By: Prof. P.S. Upparamani

2020-2021

CERTIFICATE

This is to certify that **Ms. Rachana Kampli, Ms. Laxmi Nyamagoud, Mr. Hemanth I T, Mr. Rohan Kokatanur** of **Fifth Semester** bearing **USN: 2GI18IS032, 2GI18IS020, 2GI18IS015, 2GI18IS066** has satisfactorily completed the course in Course activity of Internet Of Things. It can be considered as a bonafide work carried out for partial fulfillment of the academic requirement of 5th Semester B.E.(Information Science & Engineering) prescribed by KLS Gogte Institute of Technology, Belagavi during the academic year 2020- 2021.

The report has been approved as it satisfies the academic requirements prescribed for the said degree.

Signature Of The Faculty Member **Signature Of The HOD.**

Date: 09/01/2021

Course Project Report And Ppt Content

- Title
- Problem statement for that the project
- Objectives of Defined Problem statement
- Design / Algorithm/Flowchart/Methodology
- Implementation details/Function/Procedures/Classes and Objects (Language/Tools)
- Working model of the final solution
- Report and Oral Presentation skill

Marks allocation:

	Batch No. :					
1.	Project Title:	Marks Range	USN			
2.	Problem statement (PO2)	0-1				
3.	Objectives of Defined Problem statement (PO1,PO2)	0-2				
4.	Design / Algorithm/Flowchart/Methodology (PO3)	0-3				
5.	Implementation details/Function/Procedures/Classes and Objects (Language/Tools) (PO1,PO3,PO4,PO5)	0-4				
6.	Working model of the final solution (PO3,PO12)	0-5				
7.	Report and Oral presentation skill (PO9,PO10)	0-5				
	Total	20				

*** 20 marks is converted to 10 marks for CGPA calculation**

- **Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.
- **Problem Analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and Engineering sciences.
- **Design/Development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
- **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
- **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
- **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
- **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

- **Individual and team work:** Function effectively as an individual and as a member or leader in diverse teams, and in multidisciplinary settings.
- **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- **Project management and finance:** Demonstrate knowledge and understanding of the engineering management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- **Life-long learning:** Recognize the need for and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

TITLE:

Blinking Of Two LEDs.

PROBLEM STATEMENT:

Develop and Explain with sketch & schematic blinking of 2 separate LEDs, with delay of 5 seconds and 3 seconds respectively, for Arduino**.**

OBJECTIVES:

1. Build and wire a device with two LED's, resistor, and Arduino board.

2. Program the device to " Blinking of 2 separate LEDs, with delay of 5 seconds and 3 seconds".

ABSTRACT:

Blinking of two LEDs with different time intervals using a single device can not be done using traditional functional coding of the arduino. Hence , Object Oriented Programming(class based) has been implemented to blink two LEDs at different intervals.

INTRODUCTION:

The Arduino is a microprocessor. A Microprocessor is a compact programmable computing device with memory, processing, input, and output pins.There are 14 I/O (input/output) pins in your Arduino uno board. These pins are numbered from 0 to 13. They can be configured as either input or output in the sketch we create for arduino. The input/output configuration of pins has to be done inside the **setup()** function.

METHODOLOGY:

CODE:

```
int led1=13;
int led2=12;
unsigned long previousMillis[2]; //[x] = number of leds

void setup()
{
  pinMode(13,OUTPUT);
  pinMode(12,OUTPUT);

}

void loop() {
 BlinkLed(led1, 5000, 0);   //BlinkLed( which led, interval, one of
the stored prevMillis
 BlinkLed(led2, 3000, 1);  //last parameters must be different for
each led
}

void BlinkLed (int led, int interval, int array){

  //(long) can be omitted if you don't plan to blink led for very
long time I think
   if (((long)millis() - previousMillis[array]) >= interval){

    previousMillis[array]= millis(); //stores the millis value in the
selected array

    digitalWrite(led, !digitalRead(led)); //changes led state
  }
 }
```

FLOW CHART:

Step 1: From BlinkWithoutDelay to a single function

Here is a short example of the standard sketch that uses millis() instead of Delay().

```
#define blueLed 3
unsigned long previousMillis = 0;  // stores last time Led blinked
long interval = 100;               // interval at which to blink (milliseconds)
void setup() {                     // set the digital pin as output:
pinMode(blueLed, OUTPUT);
}
void loop() {
if (millis() - previousMillis >= interval) {         //save the last time you blinked LED
previousMillis = millis();
digitalWrite(blueLed, !digitalRead(blueLed));       //change led state
}}
```

And this is how we can compress it in a function, with some limitation yet.

```
#define blueLed 3
void setup() {
pinMode(blueLed, OUTPUT);             //pin3 Output }
void loop() { BlinkBlue(200);          //the led will blink every 200ms
} void BlinkBlue (int interval){
static long prevMill=0;               //prevMill stores last time Led blinked
if (((long)millis() - prevMill) >= interval){
prevMill= millis();
digitalWrite(blueLed, !digitalRead(blueLed)); }
```

With this function we don't need to declare any variables except the led pin and they can have different intervals but we can't use it for more than one led. Every led must have his own function, pretty annoying:

```
#define blueLed 3
#define greenLed 2
void setup() {
pinMode(blueLed, OUTPUT);                              //pin3 Output
pinMode(greenLed, OUTPUT);                             //pin2 Output
}
void loop() {
BlinkGreen(100); BlinkBlue(200);
}
void BlinkBlue (int intervallo){
static long prevMill = 0;
if (((long)millis() - prevMill) >= intervallo){
prevMill = millis(); digitalWrite(blueLed, !digitalRead(blueLed));
} }
void BlinkGreen (int intervallo){
static long prevMill = 0;
if ((millis() - prevMill) >= intervallo){
prevMill = millis();
digitalWrite(greenLed,
!digitalRead(greenLed));
} }
```

In the next step we'll try to write a single function that can be used with multiple leds.

Step 2: A single function for multiple Leds

Finally this is the code for a single function to control multiple Leds.

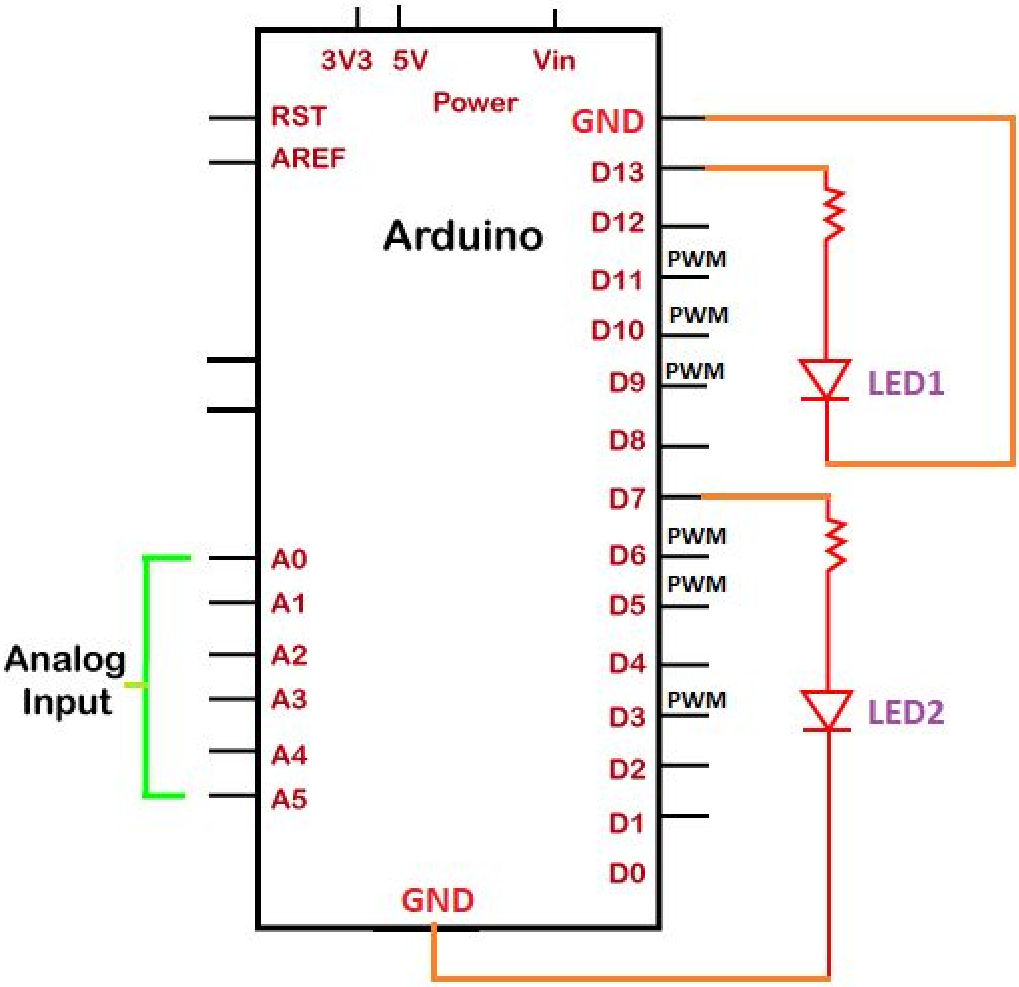
The previousMillis variable has been substituted with an array that stores all the previousMillis for each led.

When you call the function you have to insert 3 arguments: led (pin or variable), interval (in ms), array (must be different for each led).

This way every led can have its own interval, minimal variables required (only the array) and Delay is never used.

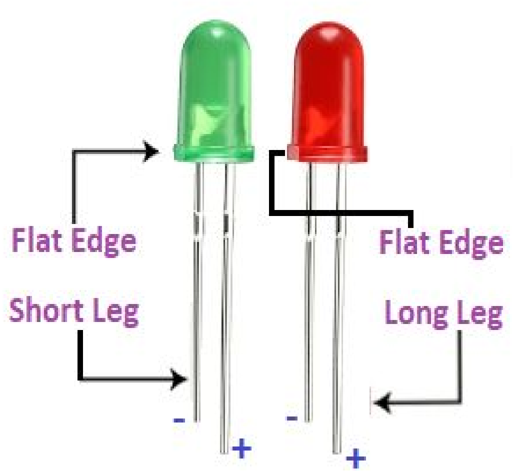
```
//BLINK FUNCTION FOR MULTIPLE LED

#define blueLed 3                    //

#define greenLed 2                   //pin for each led

unsigned long previousMillis[2];     //[x] = number of leds

void setup() {

pinMode(blueLed, OUTPUT);

pinMode(greenLed, OUTPUT); }

void loop() {

BlinkLed(blueLed, 100, 0);           //BlinkLed( which led, interval, one of
the stored prevMillis

BlinkLed(greenLed, 200, 1);          //last parameters must be different for

each led }

void BlinkLed (int led, int interval, int array){

//(long) can be omitted if you don't plan to blink led for very long time I think
if (((long)millis() - previousMillis[array]) >=

interval){

 previousMillis[array]= millis();                    //stores the millis value in the
selected array

digitalWrite(led, !digitalRead(led));                //changes led state

} }
```

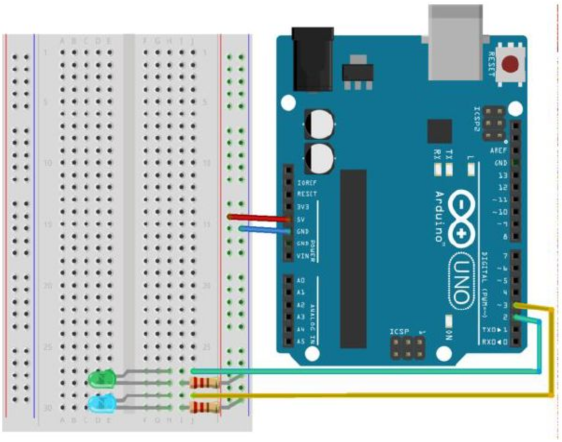
CIRCUIT DIAGRAM:

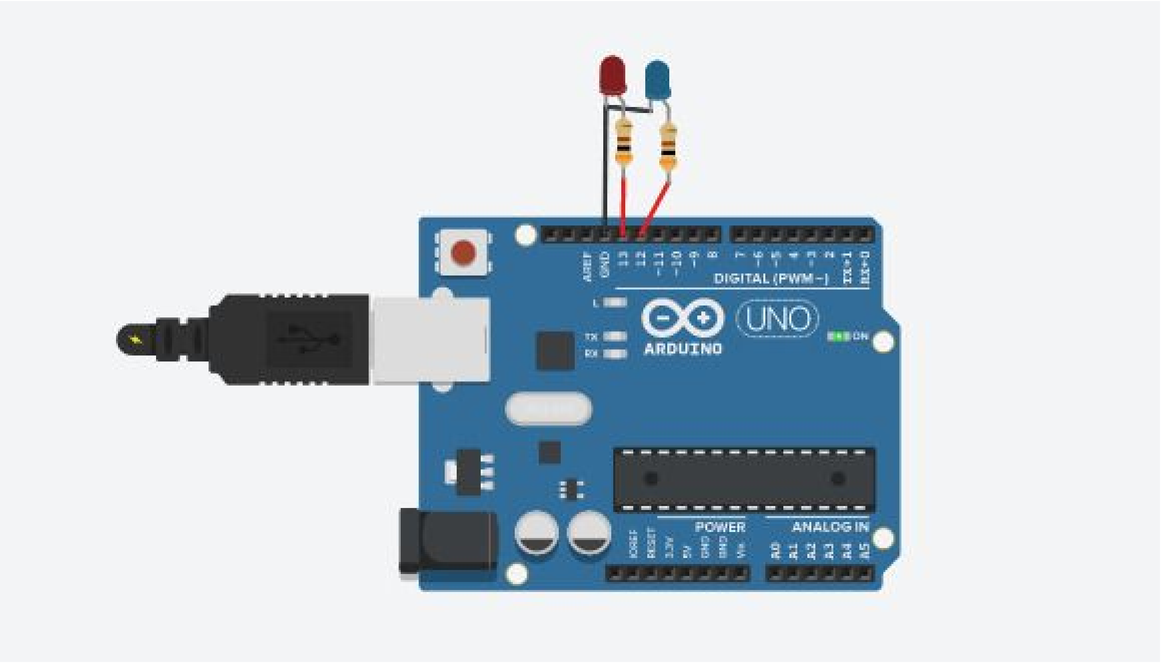


IMPLEMENTATION:

CIRCUIT CONNECTION:







CONCLUSION:

We learnt that blinking of two LEDs with different time intervals using a single device can not be done using traditional functional coding of the arduino. Hence , Object Oriented Programming(class based) has been implemented to blink two LEDs at different intervals.

REFERENCES:

1. http://www.instructables.com/id/Blink-multiple-Leds-1-Function-No-Delay/
2. https://www.javatpoint.com/arduino-blinking-two-led/
3. https://www.tinkercad.com/
4. www.wikipedia.com/
5. www.google.com/