

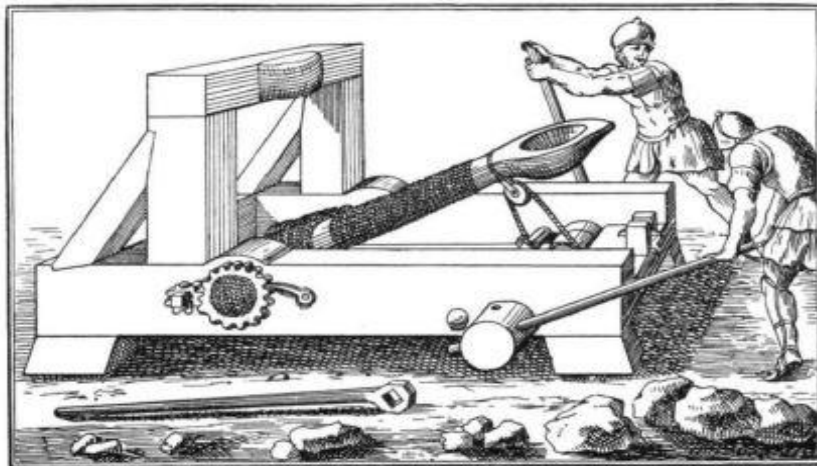


**DEPARTMENT  
OF  
ELECTRONICS AND COMMUNICATION ENGINEERING**



**THAPAR INSTITUTE**  
OF ENGINEERING & TECHNOLOGY  
(Deemed to be University)

**Handout/Assignment-1**  
for  
**Engineering Design Project-I (UTA013)**



**INSTRUCTOR INCHARGE**



---

## **ASSIGNMENT - 1**

### **INPUT / OUTPUT INTERFACE DESIGN**

#### **Exercise 1 – Blink**

1. To blink the LED with an Arduino for 1 sec 'on' time and 2 seconds 'off' time (both, on breadboard and using Tinkercad)

#### **Hardware/Software Required**

- Arduino Board
- Tinkercad Software tool (<https://www.tinkercad.com/>)
- Breadboard
- LED and Resistors

#### **Circuit description**

To build the circuit, attach a 100-ohm resistor to either leg of the diode. Attach the leg of the LED connected to the flat edge of the body (the negative leg, called the cathode) to ground. Connect the remaining leg (the positive leg, called the anode) to pin 13. Then plug your Arduino board into your computer, start the Arduino program, and upload the code.

#### **Tinkercad**

Tinkercad is a free, online collection of software tools for 3D design, electronics, and coding

#### **Getting Started:**

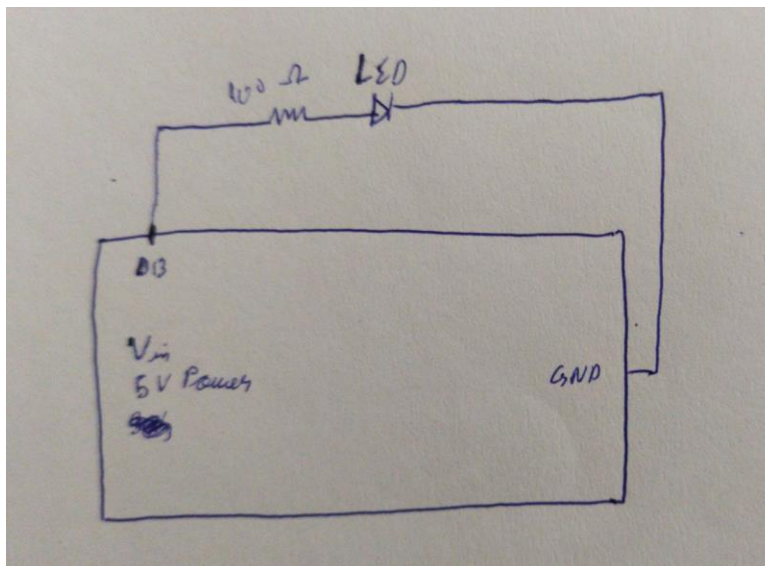
1. Visit <https://www.tinkercad.com/>
2. Sign in through your google account (Thapar Email ID only)
3. On the Dashboard, select **Circuits** from the drop box and click on **Create new Circuit**

#### **Code**



Ex - 1  
void setup()  
{  
  pinMode(13, OUTPUT);  
}  
void loop()  
{  
  digitalWrite(13, HIGH);  
  delay(1000);  
  digitalWrite(13, LOW);  
  delay(1000);  
}

### Schematic



### Tinkercad's Snapshot



### Time Based Blinking LED

Like 0

design by:  
**NISHANT GUPTA**

Edited 8/30/20, Created 8/28/20

Tinker this

Time Based Blinking LED

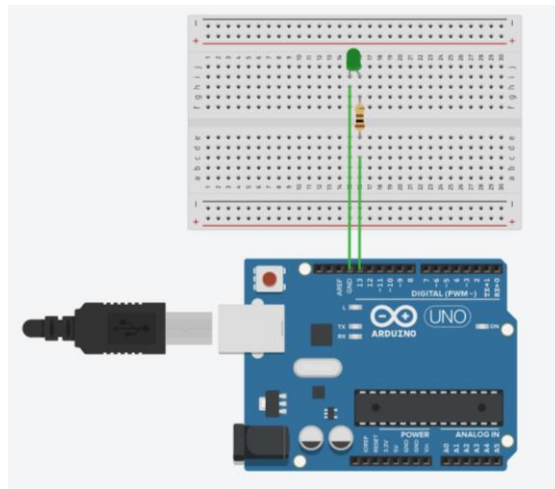
```
1 void setup()
2 {
3   pinMode(13, OUTPUT);
4 }
5
6 void loop()
7 {
8   digitalWrite(13, HIGH);
9   delay(1000); // Wait for 1000 milliseconds
10  digitalWrite(13, LOW);
11  delay(1000); // Wait for 1000 milliseconds
12 }
```

Serial Monitor

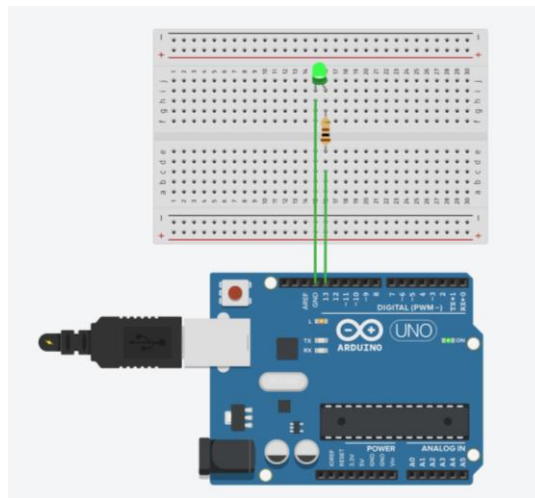


---

LED OFF



LED ON



**Reflections(Conclusions):**



a) Usage of build in function called `pin Mode()` helped to tell Arduino whether the pin is INPUT or OUTPUT

b) When a pin used as an OUTPUT digital write is written in code to command it to HIGH or LOW

## Exercise 2 – PushButton

On a breadboard, connect Pushbutton to turn on the built-in LED on pin 13 i.e. when you press the button LED should glow otherwise it should be in off state.

### Hardware Required

- Arduino Board
- momentary button or switch
- 10K ohm resistor
- breadboard
- hook-up wire

### Circuit description

Connect three wires to the Arduino board. The first two, red and black, connect to the two long vertical rows on the side of the breadboard to provide access to the 5 volt supply and ground. The third wire goes from digital pin 10 to one leg of the pushbutton. The same leg of the button connects through a pull-down resistor (here 10 KOhms) to ground. The other leg of the button connects to the 5 volt supply.

When the pushbutton is open (unpressed) there is no connection between the two legs of the pushbutton, so the pin is connected to ground (through the pull-down resistor) and we read a LOW. When the button is closed (pressed), it makes a connection between its two legs, connecting the pin to 5 volts, so that we read a HIGH.

You can also wire this circuit the opposite way, with a pullup resistor keeping the input HIGH, and going LOW when the button is pressed. If so, the behaviour of the sketch will be reversed, with the LED normally on and turning off when you press the button.

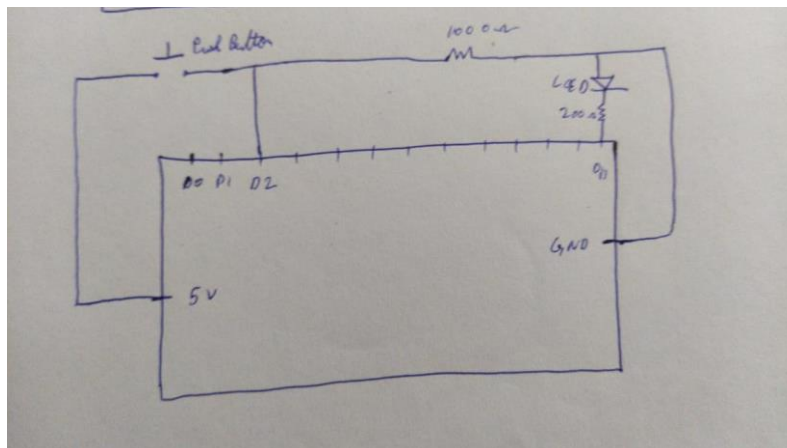
If you disconnect the digital I/O pin from everything, the LED may blink erratically. This is because the input is "floating" - that is, it will randomly return either HIGH or LOW. That's why you need a pull-up or pull-down resistor in the circuit.



## Code


```
Ex-2  
void setup()  
{  
  pinMode(13, OUTPUT)  
  pinMode(2, INPUT)  
}  
void loop()  
{  
  int x = digitalRead(2);  
  if (x == 1){  
    digitalWrite(13, HIGH);  
  }  
  else {  
    digitalWrite(13, LOW);  
  }  
}
```

## Schematic



## Tinkercad's Snapshot





Push Button

```

1 void setup()
2 {
3   pinMode(13, OUTPUT);
4   pinMode(2, INPUT);
5 }
6
7 void loop()
8 {
9   int x = digitalRead(2);
10  if (x==1)
11    digitalWrite(13,HIGH);
12  }
13  else{
14    digitalWrite(13,LOW);
15  }
16 }

```

Push Button

design by: **NISHANT GUPTA**

Edited 8/28/20, Created 8/28/20

Tinker this

Simulate Add Image

### Reflections(Conclusions):

(a) Push button is used to turn ON and OFF the led. when pushed it turns LED ON else LED remains OFF

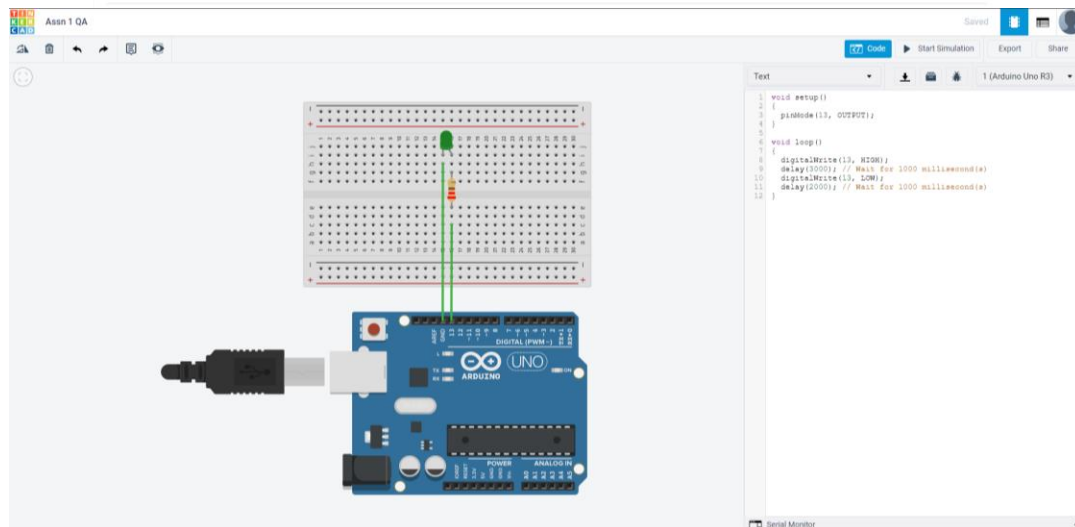
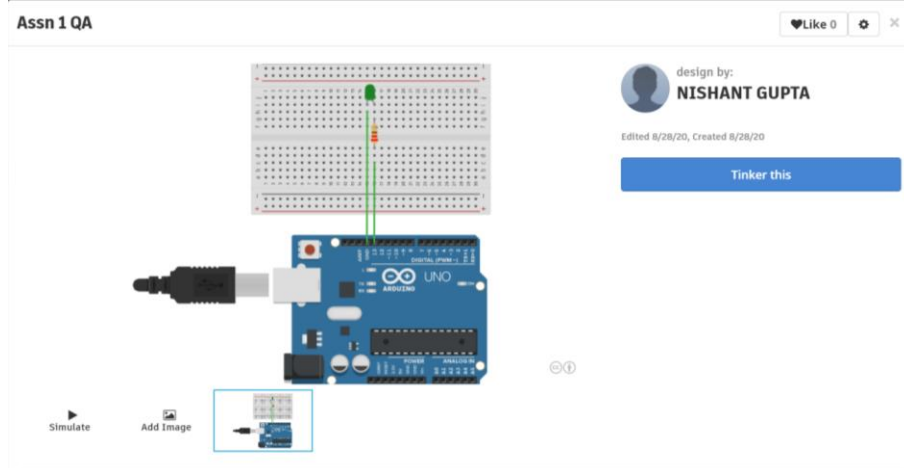
(b) It is similar method to turn ON/OFF a device





## Assignment Tasks

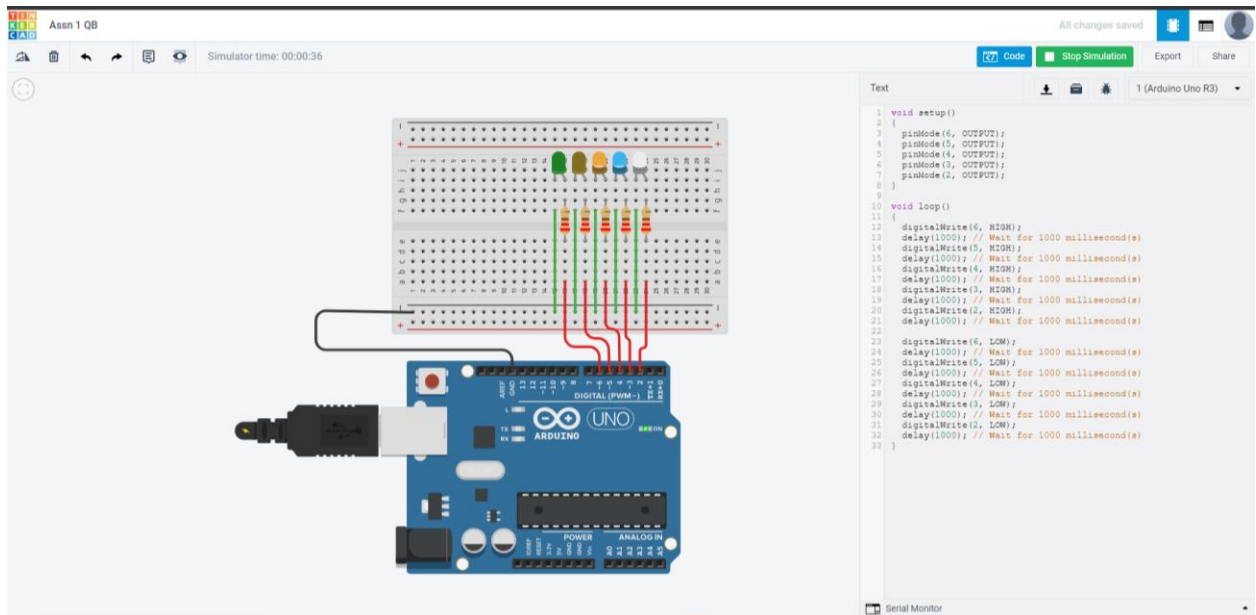
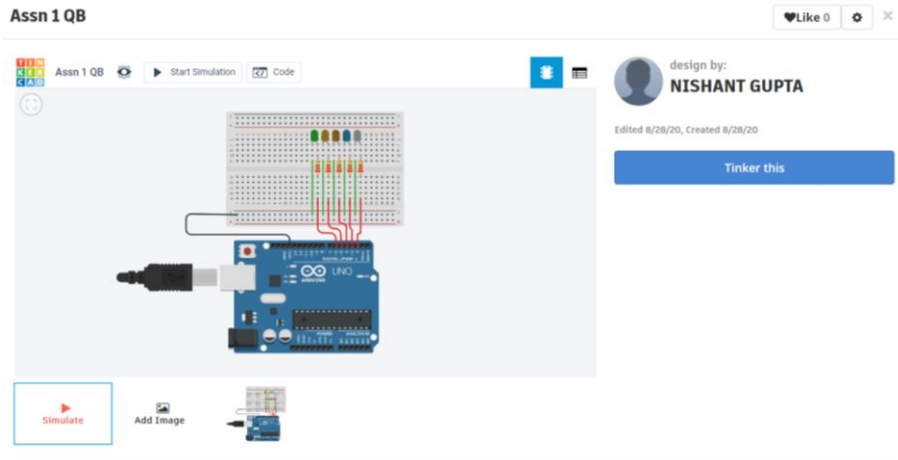
- A. Using Tinkercad, change the amount of time the LED is 'ON' to the sum of last two digits of your Roll Number, leaving the amount of time the LED is 'OFF' to 2 second. (For example, if your Roll No. is 1019\*\*\*27, the LED must be kept on for 9 second).





- B. Using Tinkercad, hook up 5 LEDs to pins 2 through 6 (with resistors). Modify the code to turn on each one in order and then extinguish them in order.

(**HINT**: hook them up one additional LED at a time and make sure the new one works before you add the next one.)





- C. Using Tinkercad, perform Exercise 2 i.e., Connect Pushbutton to turn on the built-in LED on pin 13 i.e. when you press the button LED should glow otherwise it should be in off state.

**Note:** Each Student must attach a snapshot of the circuit and code created by him/her using 'Share' icon available on Tinkercad only.

The top screenshot shows a Tinkercad workspace titled "Assn 1 QC". It features an Arduino Uno connected to a breadboard. A pushbutton is connected to pin 7 (INPUT) and a red LED is connected to pin 13 (OUTPUT). The circuit is simulated, and the LED is shown glowing. The user's name "NISHANT GUPTA" is visible in the top right corner.

The bottom screenshot shows the same circuit diagram with the code editor open on the right side. The code is as follows:

```
1 void setup()
2 {
3   pinMode(13, OUTPUT);
4   pinMode(7, INPUT);
5 }
6
7 void loop()
8 {
9   int x = digitalRead(7);
10  if (x==1){
11    digitalWrite(13, HIGH);
12  }
13  else{
14    digitalWrite(13, LOW);
15  }
16 }
```