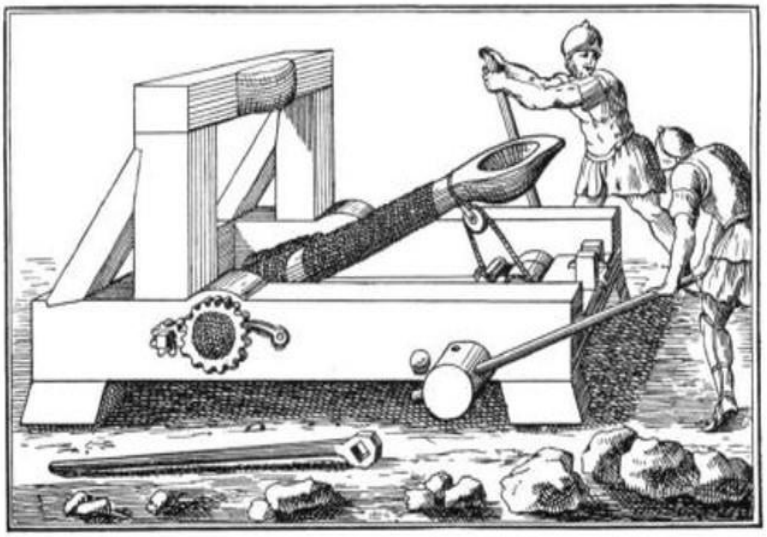
DEPARTMENT
OF
ELECTRONICS AND COMMUNICATION ENGINEERING

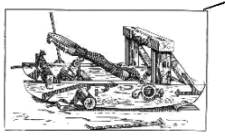
THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

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



INSTRUCTOR INCHARGE

Meharamt Singh 102003241 2CO10



ASSIGNMENT – 2 (A)
STUDY OF IR SENSORS

Exercise 1

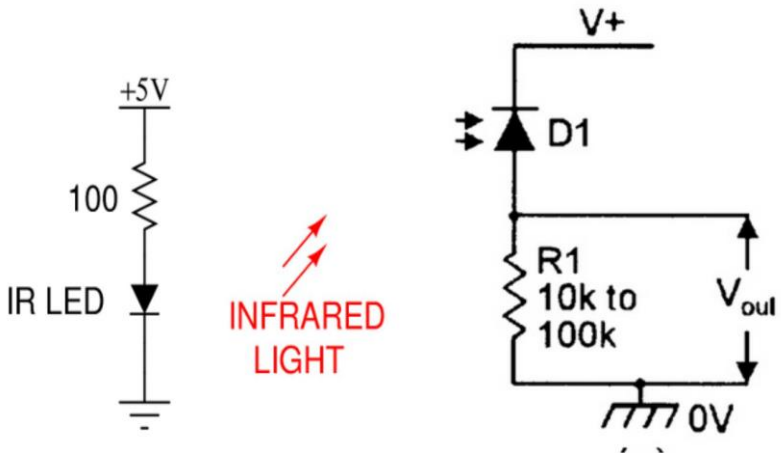
A. Demonstrate the working of IR sensors and receiver and display output using LED.
B. Use the two pair IR sensors of Mangonel to combine the two sensors output into one signal.

Hardware

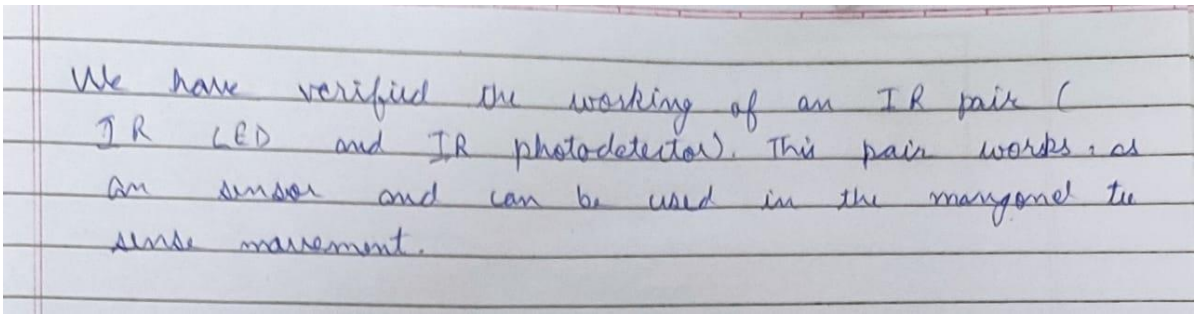
- Bread Board, Power supply
- Resistances and LED
- IR transmitter and Receiver (Photodiode), Single core connecting wires

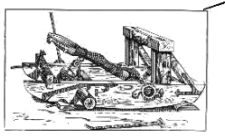
Theory

The figure below shows an IR pair in which IR LED emits infrared light which is received by photo diode D1 and the output voltage across resistor R1 is high. When we block the flow of light then the output voltage becomes low.



Reflections (Conclusions):





ASSIGNMENT – 2(B)

Exercise 1 – To verify the functional table of CD4543

Hardware Required

- Decoder (CD4543)
- Seven Segment Display
- Single core connecting wires

Theory

The decoder (CD4543) is a combinational digital circuit that decodes an 4-bit binary input in the range 0000-1001 (BCD) in to its corresponding decimal level. Example for the binary value 0101 we need to display 5. Hence the decoder will output a HIGH on segments (a, c, d, f and g) with output a LOW on segments (b and e). The latch signal is normally connected to 5V via 10Kohm resistor as per the circuit diagram. This allows the decoder to decode the present binary input (the latch is said to be in a transparent state). When the latch is connected to 0V via the jumper provided its logic state changes to a LOW and the decoder will decode the binary input prior to the latch going low (i.e. the display is frozen when the latch is LOW).

Schematic Diagram

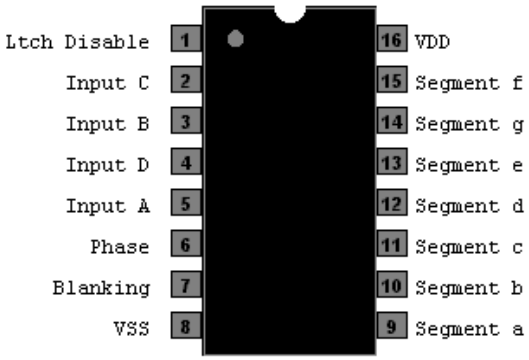
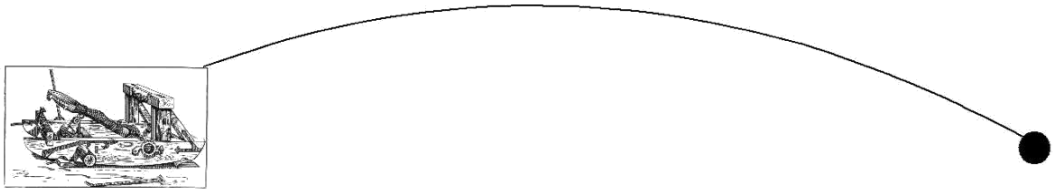


Figure 1: Pin diagram of CD4543

Truth table of CD4543B for Common Cathode Seven Segment Display.

LD	BL	PH	D	C	B	A	a	b	c	d	e	f	g	DISPLAY
1	0	0	0	0	0	0	1	1	1	1	1	1	0	0
1	0	0	0	0	0	1	0	1	1	0	0	0	0	1
1	0	0	0	0	1	0	1	1	0	1	1	0	1	2
1	0	0	0	0	1	1	1	1	1	1	0	0	1	3
1	0	0	0	1	0	0	0	1	1	0	0	1	1	4
1	0	0	0	1	0	1	1	0	1	1	0	1	1	5
1	0	0	0	1	1	0	1	0	1	1	1	1	1	6
1	0	0	0	1	1	1	1	1	1	0	0	0	0	7
1	0	0	1	0	0	0	1	1	1	1	1	1	1	8
1	0	0	1	0	0	1	1	1	1	1	0	1	1	9

Figure 2: Functional table of CD4543



Exercise 2 – BCD (binary coded decimal) to 7 Segment Display

Hardware Required

- Decoder (CD4543)
- Seven Segment Display
- Single core connecting wires
- Arduino Uno

Theory

The decoder (CD4543) is a combinational digital circuit that decodes an 4-bit binary input in the range 0000-1001 (BCD) in to its corresponding decimal level. Example for the binary value 0101 we need to display 5. Hence the decoder will output a HIGH on segments (a, c, d, f and g) with output a LOW on segments (b and e). The latch signal is normally connected to 5V via 10Kohm resistor as per the circuit diagram.
This allows the decoder to decode the present binary input (the latch is said to be in a transparent state). When the latch is connected to 0V via the jumper provided its logic state changes to a LOW and the decoder will decode the binary input prior to the latch going low (i.e. the display is frozen when the latch is LOW).

Schematic Diagram

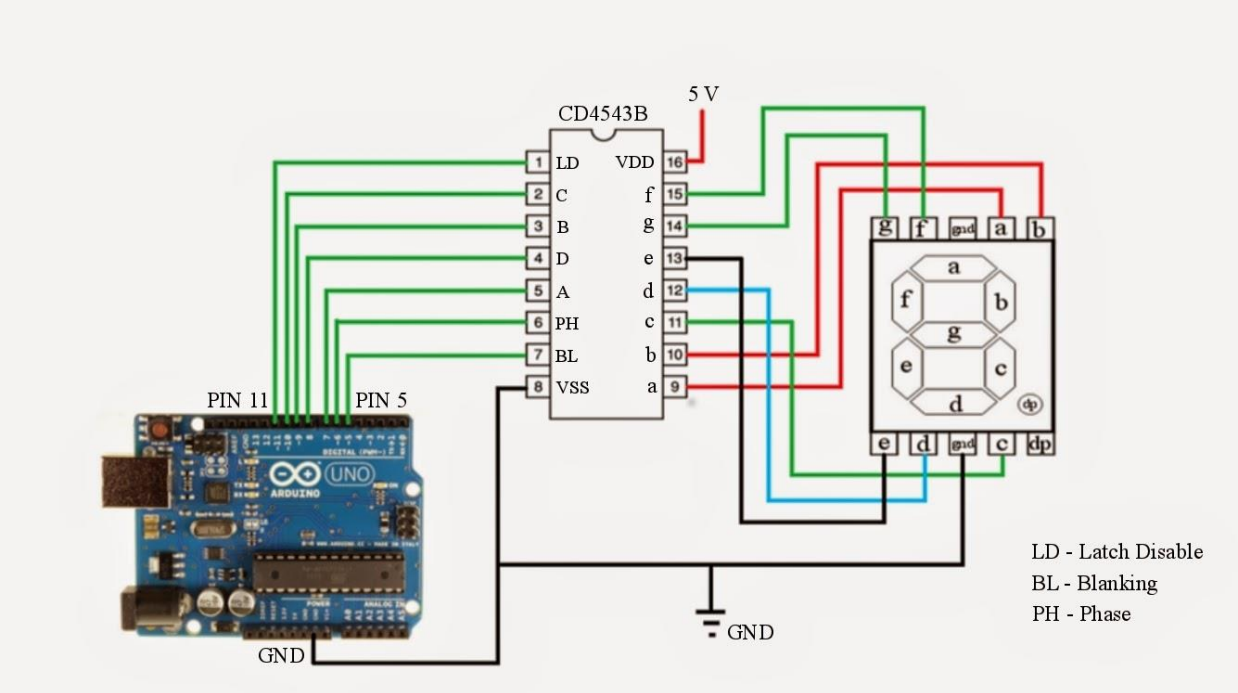
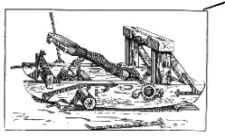
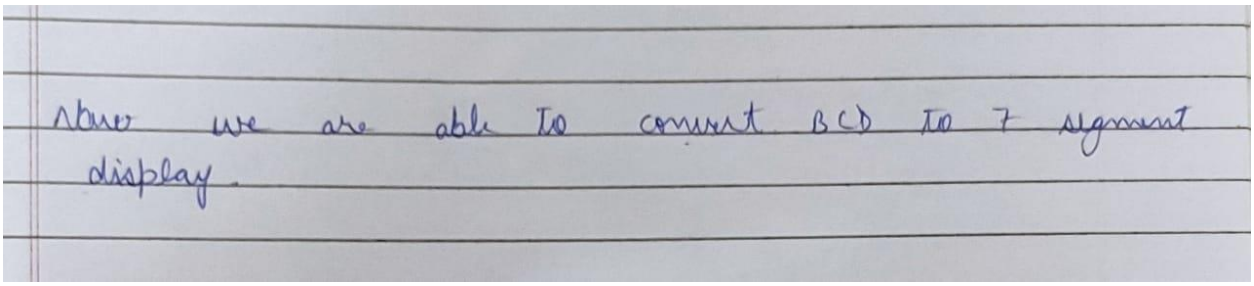


Figure 1: Connection setup for converting BCD input to seven segment ouput.



Reflections (Conclusions):

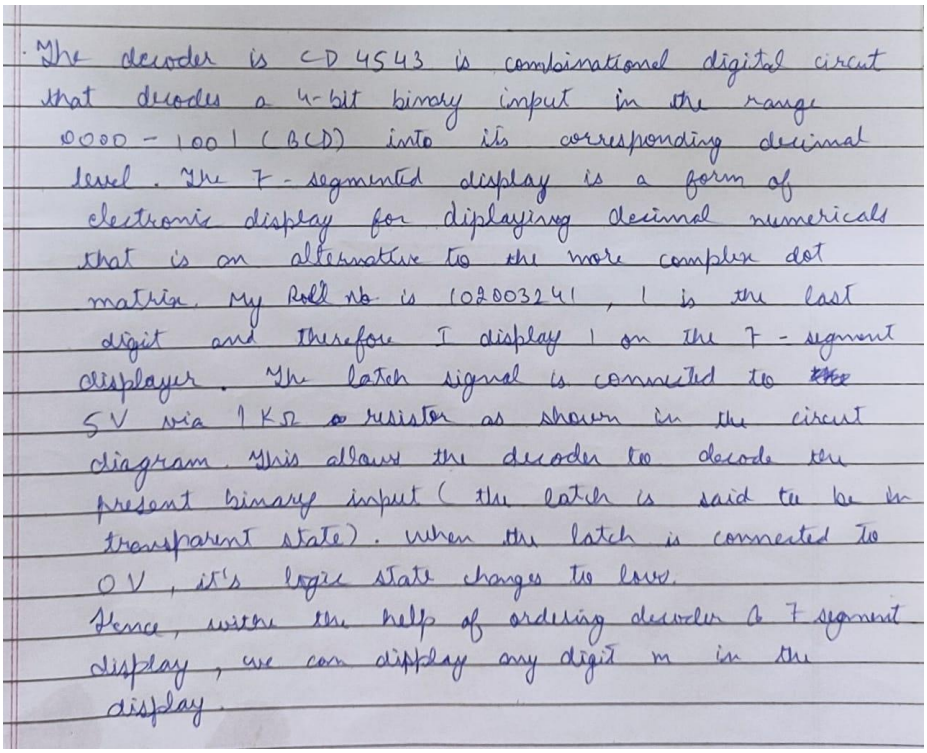


Exercise 3 – Write an Arduino sketch to display the last digit of your Roll Number on the 7-segment display using Tinkercad

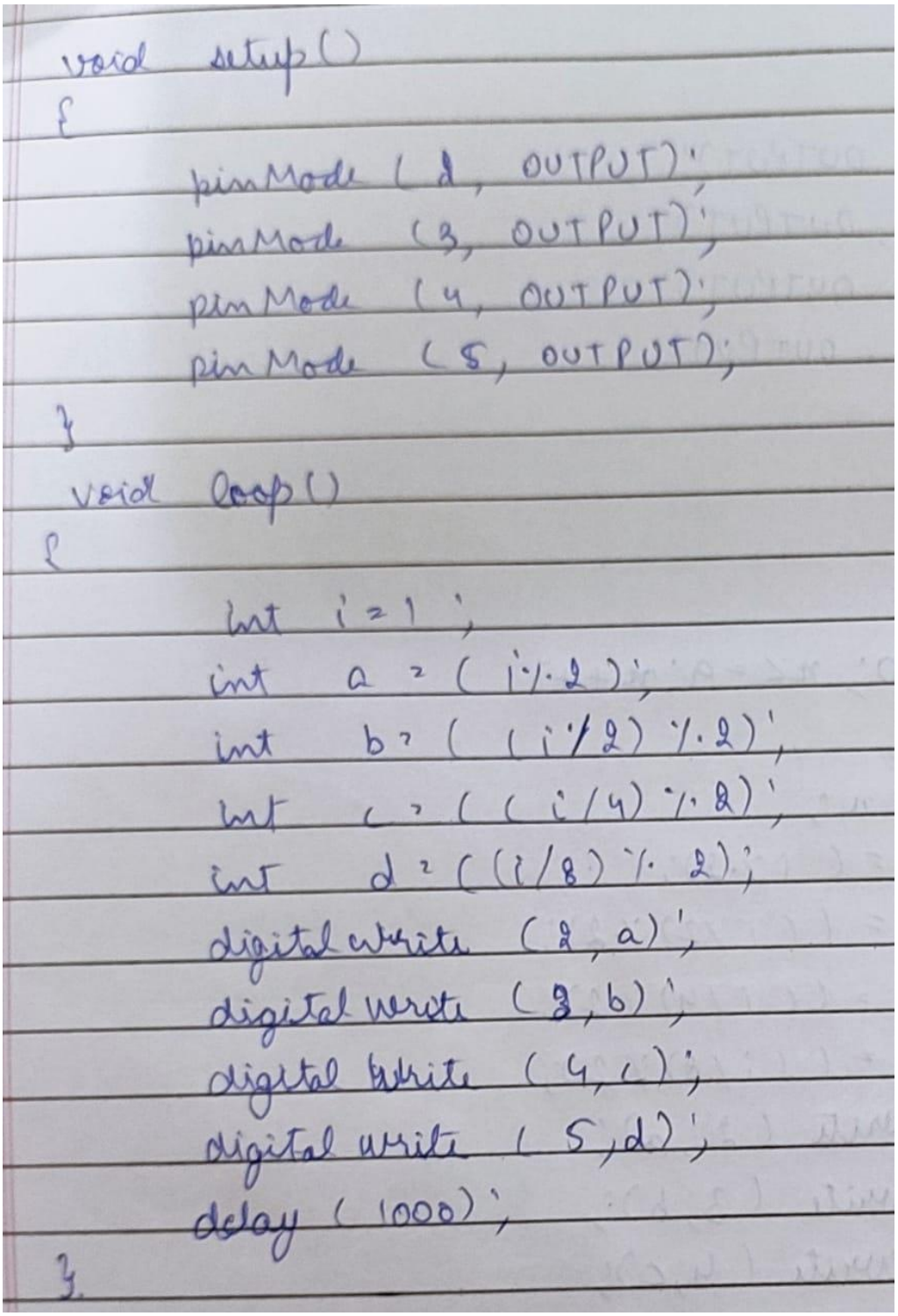
Hardware Required

- Decoder (CD4543)
- Seven Segment Display
- Single core connecting wires
- Arduino Uno
- Tinkercad Software tool (https://www.tinkercad.com/)

Theory (Write the theory as per your understanding during self-effort and lab hours)

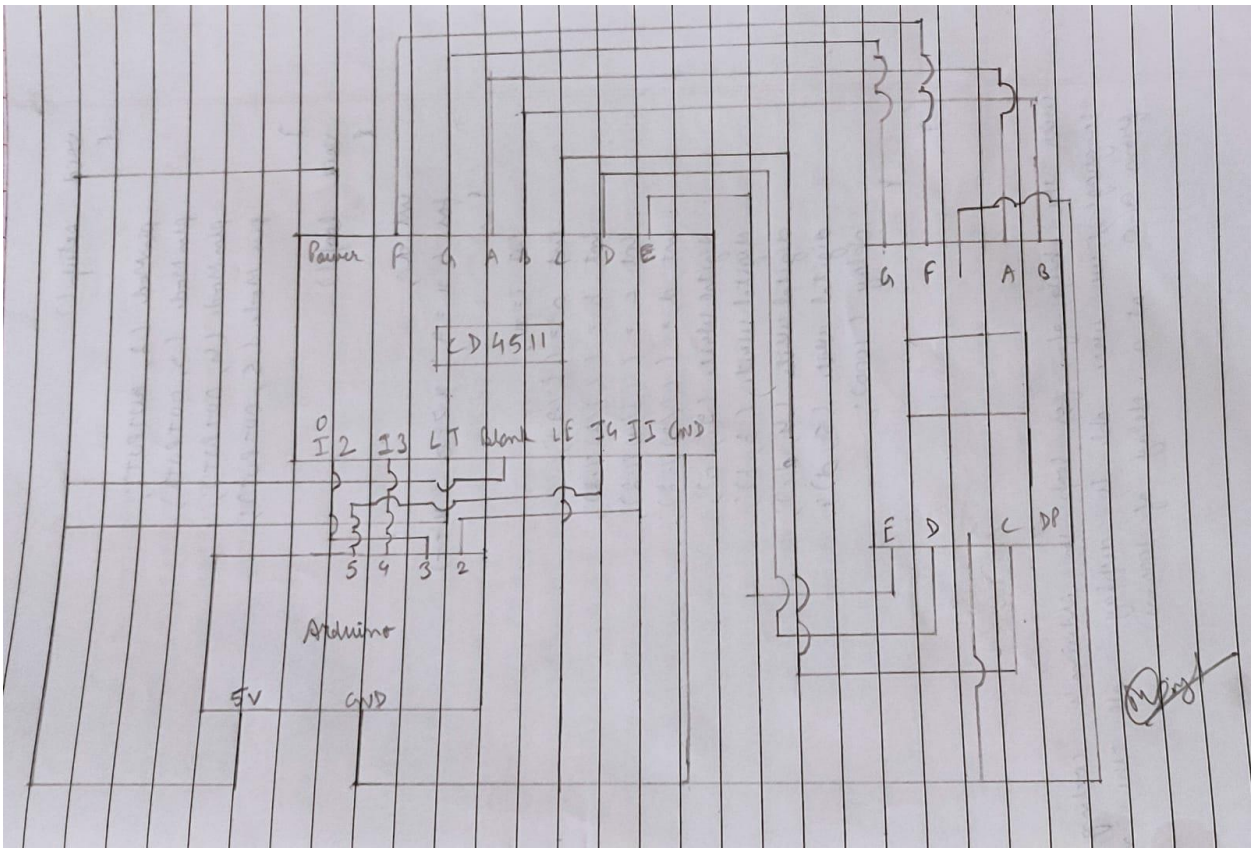


Code:

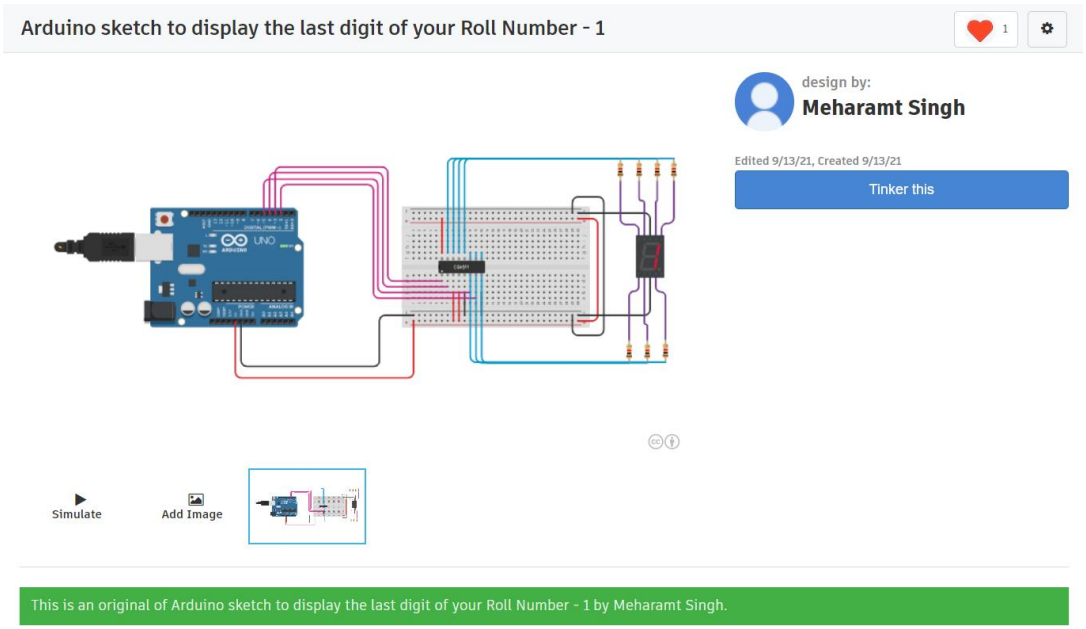


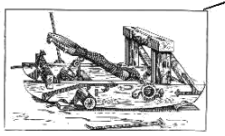
```
void setup()
{
    pinMode (2, OUTPUT);
    pinMode (3, OUTPUT);
    pinMode (4, OUTPUT);
    pinMode (5, OUTPUT);
}
void loop()
{
    int i=1;
    int a = (i%2);
    int b = ((i/2)%2);
    int c = ((i/4)%2);
    int d = ((i/8)%2);
    digitalWrite (2, a);
    digitalWrite (3, b);
    digitalWrite (4, c);
    digitalWrite (5, d);
    delay (1000);
}
```

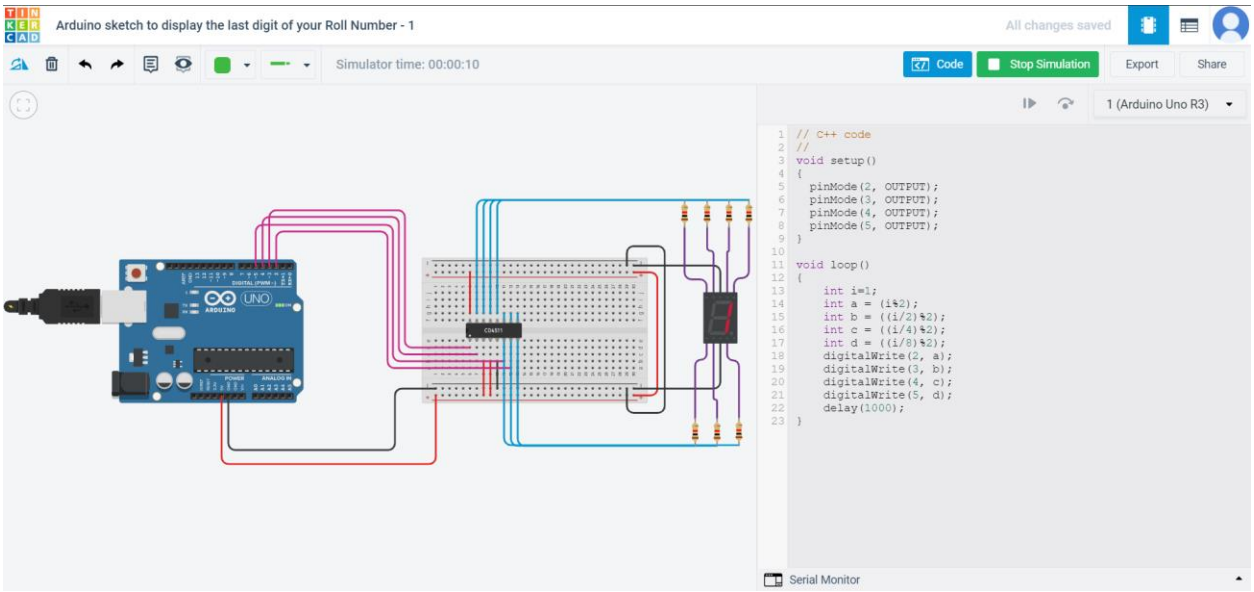
Experimental connection diagram:



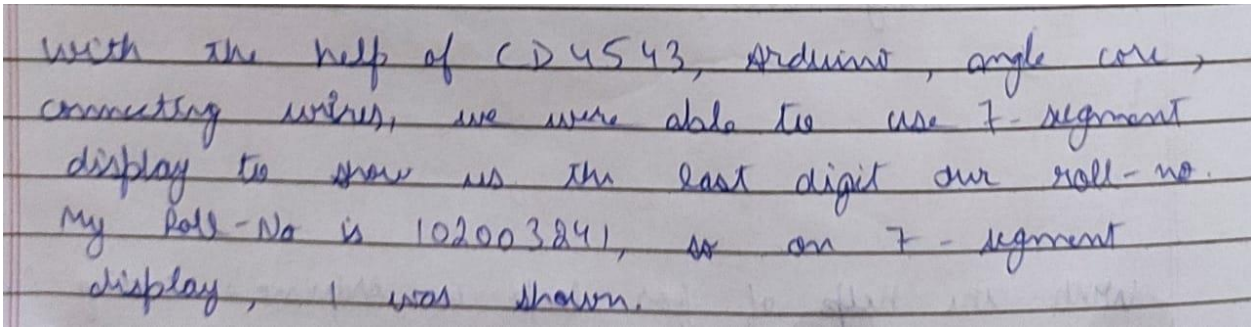
Tinkercad Snapshots:

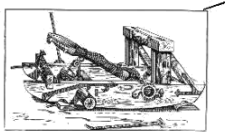






Reflections:

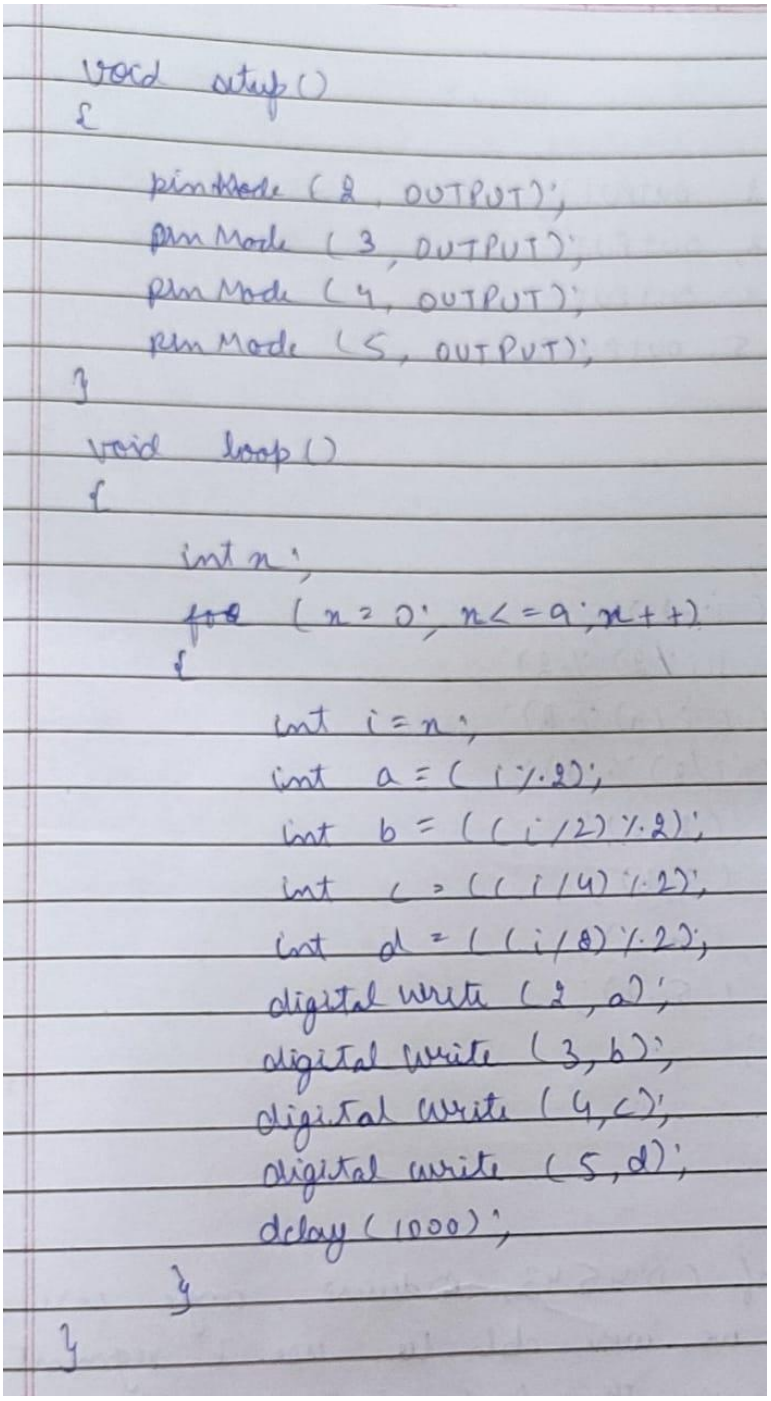




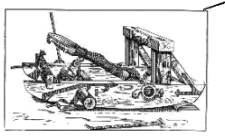
Assignment Tasks:

- Using Tinkercad, write an Arduino sketch to make an up counter which counts from 0 to 9 & repeat it infinitely.

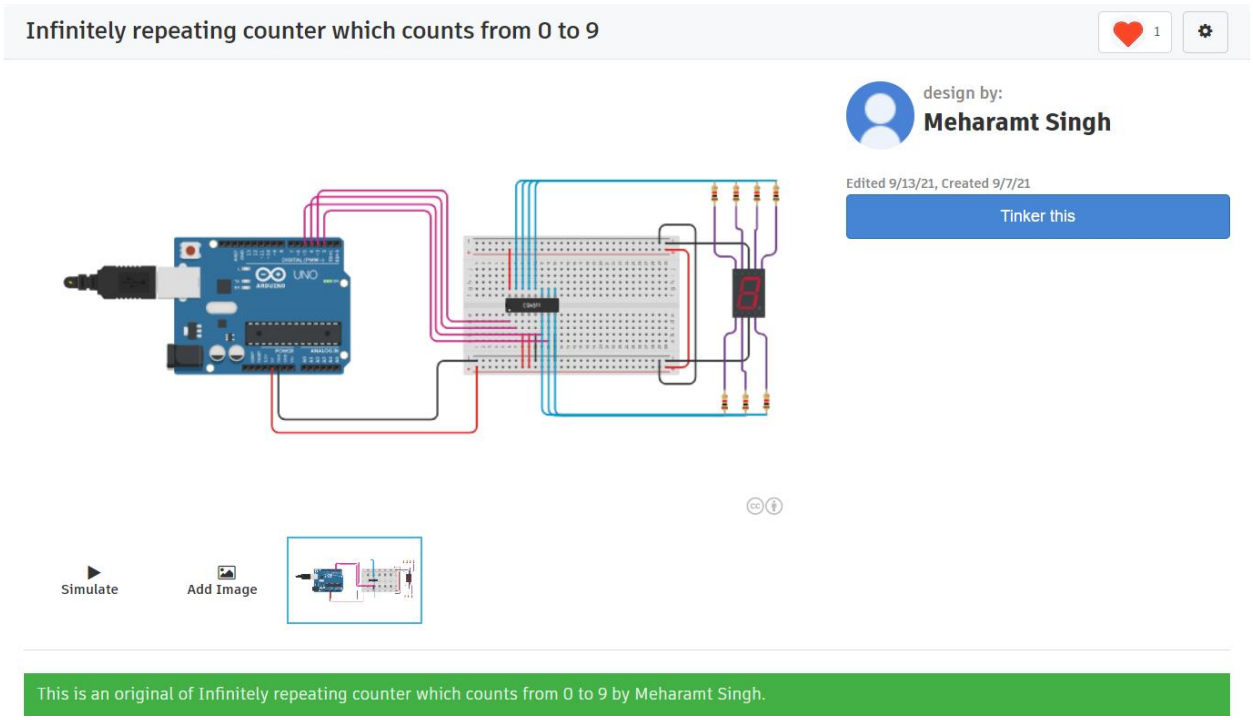
Code:

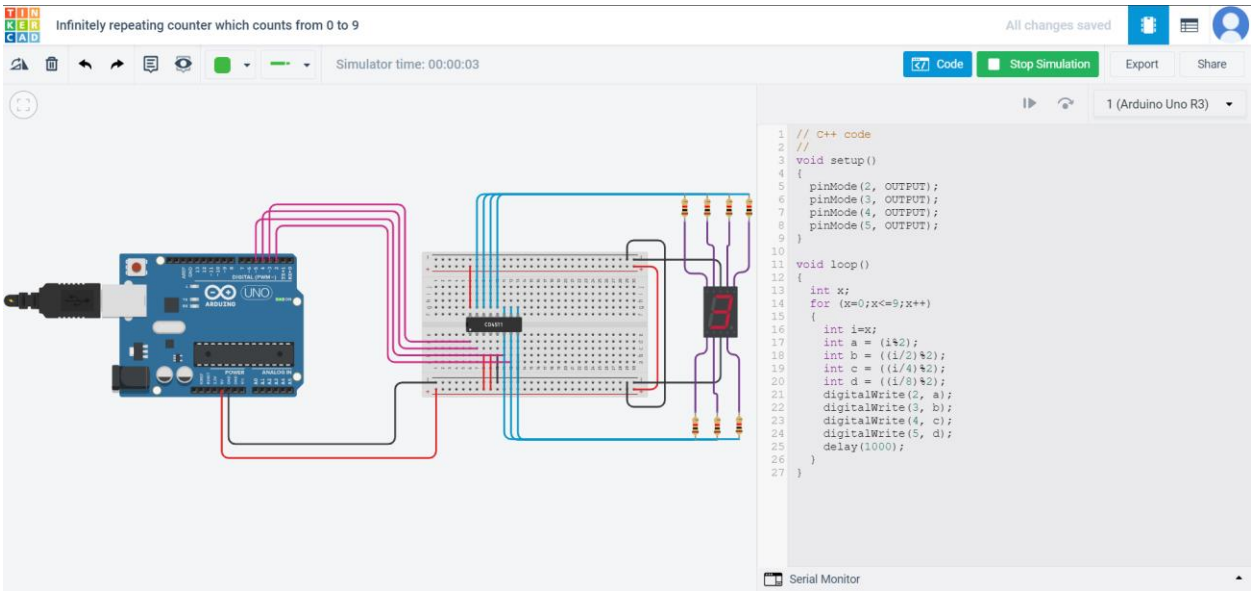


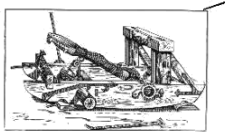
```
void setup()
{
    pinMode (2, OUTPUT);
    pinMode (3, OUTPUT);
    pinMode (4, OUTPUT);
    pinMode (5, OUTPUT);
}
void loop()
{
    int n;
    for (n=0; n<=9; n++)
    {
        int i=n;
        int a=(i%2);
        int b=((i/2)%2);
        int c=((i/4)%2);
        int d=((i/8)%2);
        digitalWrite (2, a);
        digitalWrite (3, b);
        digitalWrite (4, c);
        digitalWrite (5, d);
        delay (1000);
    }
}
```



Tinkercad Snapshots:







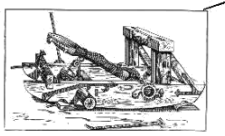
Reflections (Conclusions):

With the help of for loop in Arduino's coding language, we were able to diplay all the numbers from 0-9 at a delay of 1000 ms.

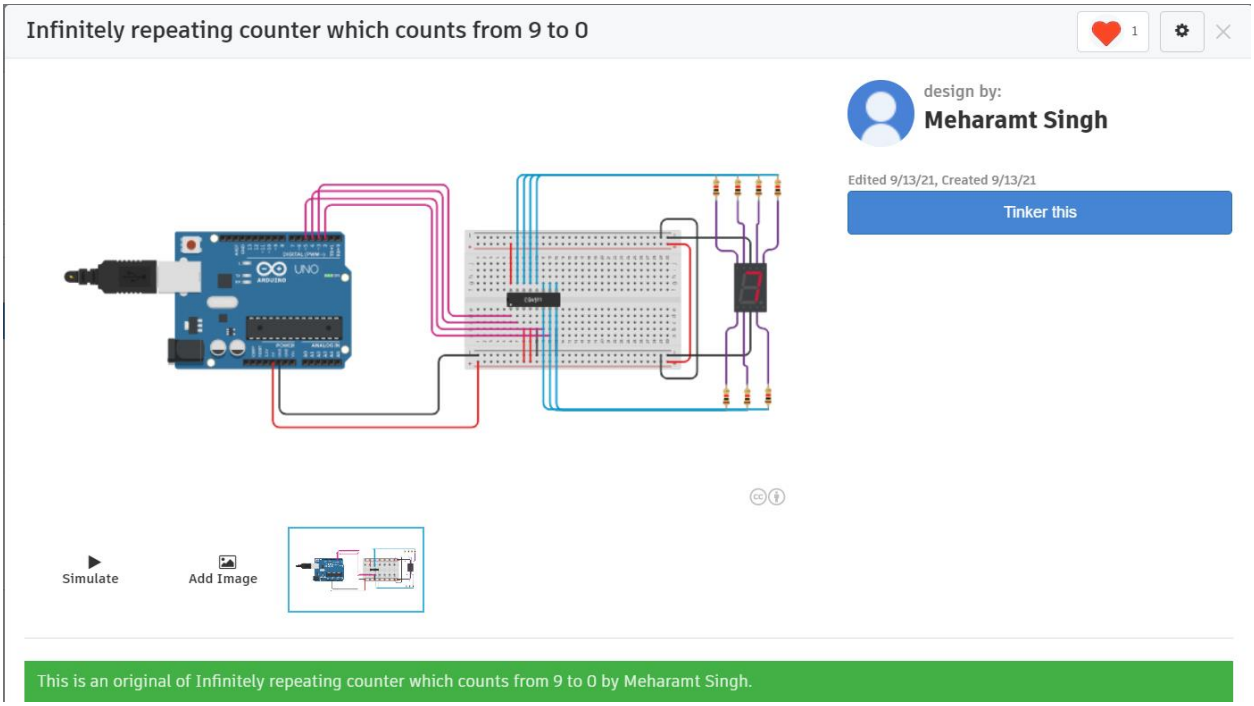
- Using Tinkercad, write an Arduino sketch to make an up counter which counts from 9 to 0 & repeat it infinitely.

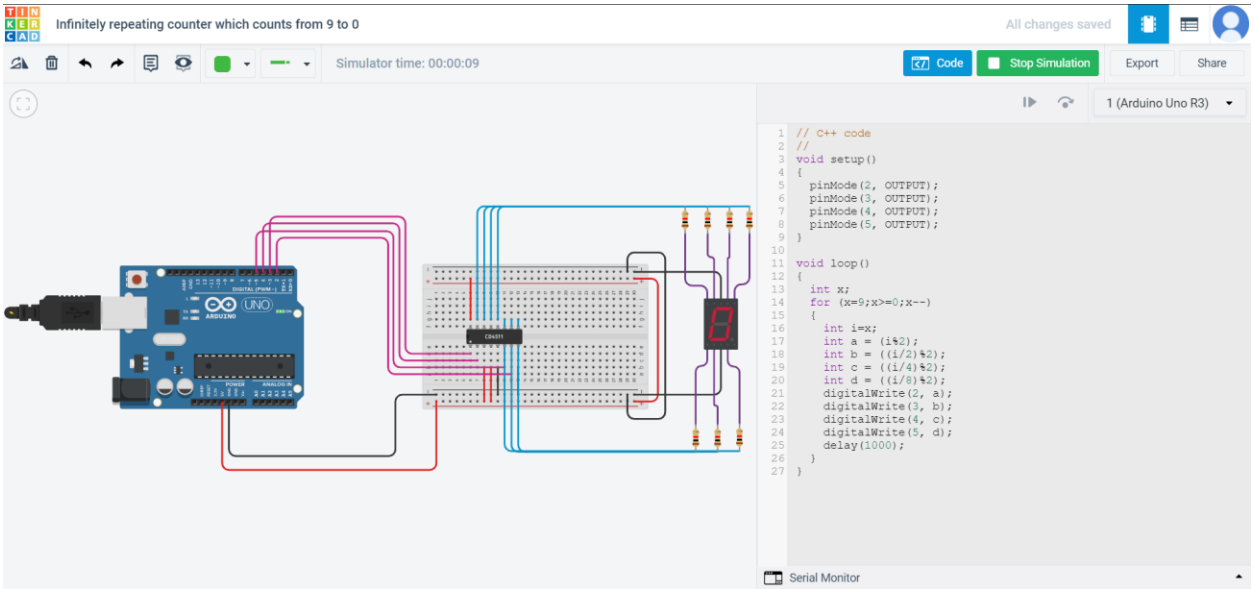
Code:

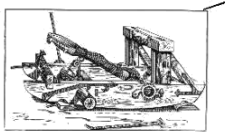
```
void setup()
{
    pinMode(2, OUTPUT);
    pinMode(3, OUTPUT);
    pinMode(4, OUTPUT);
    pinMode(5, OUTPUT);
}
void loop()
{
  int x;
  for(x=9; x>=0; x--)
  {
    int i=x;
    int a = (i%2);
    int b = ((i/2)%2);
    int c = ((i/4)%2);
    int d = ((i/8)%2);
    digitalWrite(2, a);
    digitalWrite(3, b);
    digitalWrite(4, c);
    digitalWrite(5, d);
    delay(1000);
  }
}
```



Tinkercad Snapshots:







Reflections (Conclusions):