



KIIT, Deemed to be University
School of Computer Engineering
Sensors And Automation [EC28005]

EXPERIMENT -3

Aim:

To successfully interface a 7-segment display with an Arduino microcontroller and write a program to display the digits 0 to 9 sequentially on the display.

Component/Software Used:

Component/Software	Specification
Arduino Uno	-
Bread Board, Cables, Connecting Wires, Laptop/Computer, 7 Segment Display	-
Software(s) Used	Arduino IDE 2.2.1

Theory:

Principle of Working:

The principle of working involves utilizing the Arduino microcontroller to interface with a 7-segment display. A 7-segment display is composed of seven individual

LED segments arranged in a specific pattern to represent numerical digits (0-9) or other characters. The Arduino microcontroller will be programmed to control which segments are illuminated for each digit, thereby displaying the desired digits sequentially. This will be achieved by utilizing the digital output pins of the Arduino to control the segments of the 7-segment display, following a predefined pattern for each digit to be displayed. Additionally, a current-limiting resistor may be used to protect the LEDs and limit the current flowing through the segments.

By understanding the principle of segment arrangement in the 7-segment display and the way Arduino can control it through its digital output pins, we can effectively write a program to display the digits 0 to 9 sequentially on the display.

Program:

```
void setup() {  
  
  pinMode(2,OUTPUT) ;  
  
  pinMode(3,OUTPUT) ;  
  
  pinMode(4,OUTPUT) ;  
  
  pinMode(5,OUTPUT) ;  
  
  pinMode(6,OUTPUT) ;  
  
  pinMode(7,OUTPUT) ;  
  
  pinMode(8,OUTPUT) ;  
  
  pinMode(9,OUTPUT) ;  
  
  void loop() {  
  
    for (int i = 0; i <= 9; i++)  
  
      switch(i) {  
  
        case 0:  
  
          digitalWrite(2,0) ;  
  
          digitalWrite(3,0) ;
```

```
digitalWrite(4,0);  
  
digitalWrite(5,0);  
  
digitalWrite(6,0);  
  
digitalWrite(7,0);  
  
digitalWrite(8,1);  
  
digitalWrite(9,1);  
  
Break;  
  
case 1:  
  
digitalWrite(2,1);  
  
digitalWrite(3,0);  
  
digitalWrite(4,0);  
  
digitalWrite(5,1);  
  
digitalWrite(6,1);  
  
digitalWrite(7,1);  
  
digitalWrite(8,1);  
  
digitalWrite(9,1);  
  
Break;  
  
case 2:  
  
digitalWrite(2,0);  
  
digitalWrite(3,0);  
  
digitalWrite(4,1);  
  
digitalWrite(5,0);  
  
digitalWrite(6,0);  
  
digitalWrite(7,1);  
  
digitalWrite(8,0);  
  
digitalWrite(9,1);  
  
Break;
```

```
case 3:

digitalWrite(2,0);

digitalWrite(3,0);

digitalWrite(4,0);

digitalWrite(5,0);

digitalWrite(6,1);

digitalWrite(7,1);

digitalWrite(8,0);

digitalWrite(9,1);

Break;

case 4:

digitalWrite(2,1);

digitalWrite(3,0);

digitalWrite(4,0);

digitalWrite(5,1);

digitalWrite(6,1);

digitalWrite(7,0);

digitalWrite(8,0);

digitalWrite(9,1);

Break;

case 5:

digitalWrite(2,0);

digitalWrite(3,1);

digitalWrite(4,0);

digitalWrite(5,0);

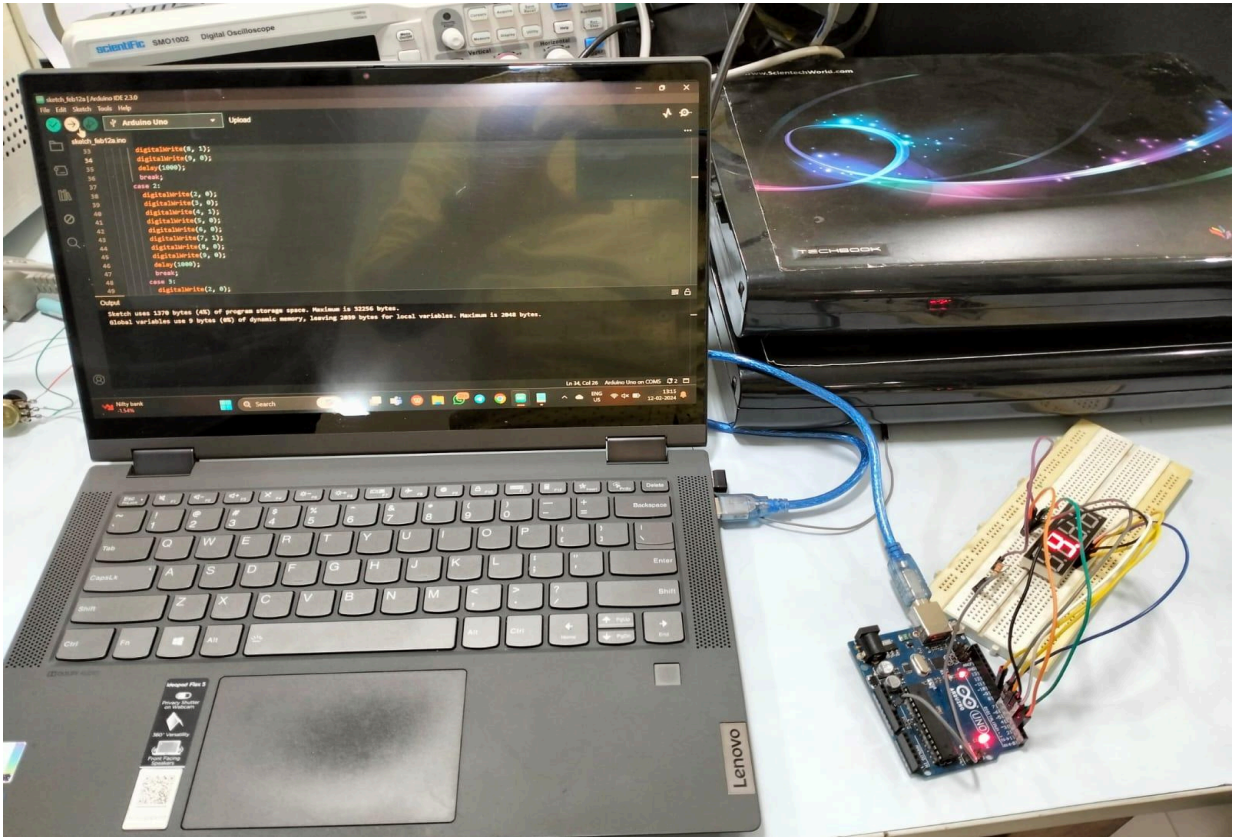
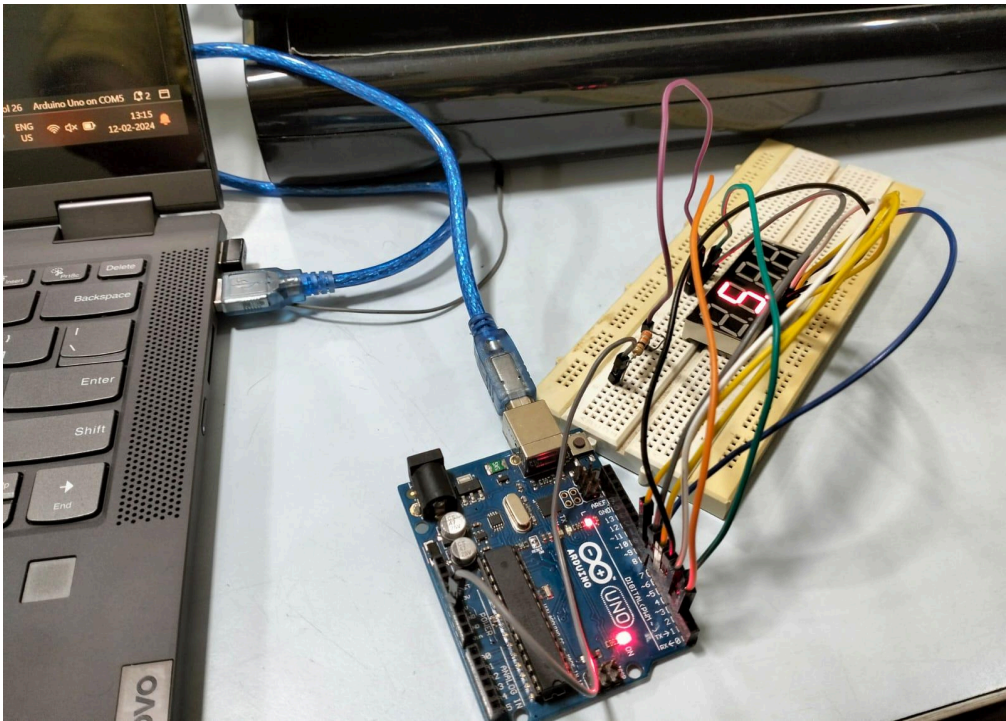
digitalWrite(6,1);

digitalWrite(7,0);
```

```
digitalWrite(8,0);  
  
digitalWrite(9,1);  
  
Break;  
  
case 6:  
  
digitalWrite(2,0);  
  
digitalWrite(3,1);  
  
digitalWrite(4,0);  
  
digitalWrite(5,0);  
  
digitalWrite(6,0);  
  
digitalWrite(7,0);  
  
digitalWrite(8,0);  
  
digitalWrite(9,1);  
  
Break;  
  
case 7:  
  
digitalWrite(2,0);  
  
digitalWrite(3,0);  
  
digitalWrite(4,0);  
  
digitalWrite(5,1);  
  
digitalWrite(6,1);  
  
digitalWrite(7,1);  
  
digitalWrite(8,1);  
  
digitalWrite(9,1);  
  
Break;  
  
case 8:  
  
digitalWrite(2,0);  
  
digitalWrite(3,0);  
  
digitalWrite(4,0);
```

```
digitalWrite(5,0);  
  
digitalWrite(6,0);  
  
digitalWrite(7,0);  
  
digitalWrite(8,0);  
  
digitalWrite(9,1);  
  
Break;  
  
case 9:  
  
digitalWrite(2,0);  
  
digitalWrite(3,0);  
  
digitalWrite(4,0);  
  
digitalWrite(5,0);  
  
digitalWrite(6,1);  
  
digitalWrite(7,0);  
  
digitalWrite(8,0);  
  
digitalWrite(9,1);  
  
Break;  
  
}  
  
}  
  
}
```

Result picture:



Conclusion:

Successfully interfacing a 7-segment display with an Arduino microcontroller and writing a program to display the digits 0 to 9 sequentially on the display requires a solid understanding of the principles of 7-segment displays and the capabilities of the Arduino. By utilizing the digital output pins of the Arduino and considering the arrangement of segments in the 7-segment display, it is possible to achieve the desired display of digits. This experiment provides a practical application of microcontroller programming and circuit interfacing skills, offering a hands-on opportunity to learn about control systems and digital display technologies.

Name: Aditya Bahadur

Roll No: 2205787

Class: CSE 38

Group No: 04