LABORATORY PROJECT REPORT

DIGITAL LOGIC SYSTEM

EXPERIMENT 2

Section 1

Group 2

Semester 1 2025/2026

NO	NAME	MATRIC NO
1.	ADAM NURUDDIN BIN HELMI	2215783
2.	NUR FATIHAH HANNANI BINTI HASMAYADI	2227062
3.	NUR SYAHZMAN AZIQ BIN MOHD SHAHMADI	2310037

Date of submission: 22nd October 2025

Abstract

This experiment focuses on the interfacing and control of a 7-segment display using an Arduino Uno microcontroller. The objective is to understand the fundamental operation of digital logic systems by implementing a simple numerical display system. The methodology involves connecting a common cathode 7-segment display to an Arduino, using digital pins to control individual segments. A pushbutton interface is added to increment and reset the displayed numbers. The experiment demonstrates how to programmatically activate specific segments to display numbers from 0 to 9. Key findings include the practical application of logic gates in display systems and the importance of current-limiting resistors for circuit protection. The results highlight the effectiveness of microcontroller-based digital logic control and offer insights into expanding the system for more advanced applications, such as multiplexed displays. The experiment serves as a foundational exercise for mechatronics students in understanding electronic circuit interfacing and digital logic implementation.

Table of Contents

1.0 Introduction

This experiment focuses on interfacing a common cathode 7-segment display with an Arduino Uno to display numbers from 0 to 9 using digital logic. Pushbuttons are used to manually control the count, allowing for incrementing and resetting the display. The experiment covers key concepts in digital logic systems, such as basic logic gates, electronic circuit interfacing, and IC-based applications. Students will learn how to configure Arduino digital output pins, use resistors for current limiting, and write code to control the display. Additionally, the lab introduces a comparison between 7-segment displays, 12C LCDs, and LED matrix displays, highlighting their differences in coding and applications. This hands-on experiment helps students understand microcontroller programming, circuit design, and display interfacing in embedded systems. This experiment is anticipated to demonstrate basic digital logic control, binary-to-decimal conversion using code, and effective use of Arduino I/O for hardware interfacing.

2.0 Materials and Equipment

2.1 Electronic Components

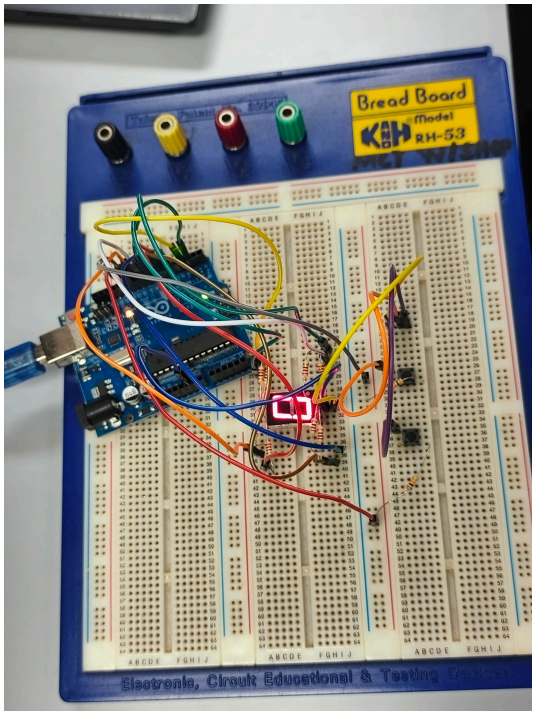
- Arduino Uno
- Common Cathode 7-Segment Display
- 220-Ohm Resistors (7x)
- Jumper Wires
- Breadboard
- Pushbutton

2.2 Equipment and Tools

- Arduino IDE
- USB Cable
- Power Supply (5V via USB or external source)

3.0 Experimental Setup

1. Connect each of the seven segments (a, b, c, d, e, f, g) of the 7-segment display to separate digital pins on the Arduino.
2. Attach the common cathode pin of the display to the ground (GND) pin on the Arduino.
3. For the pushbutton, connect one of its legs to a digital pin on the Arduino, another to the ground (GND) pin, and a third to the 5V pin through a resistor, as illustrated in Figure 1.



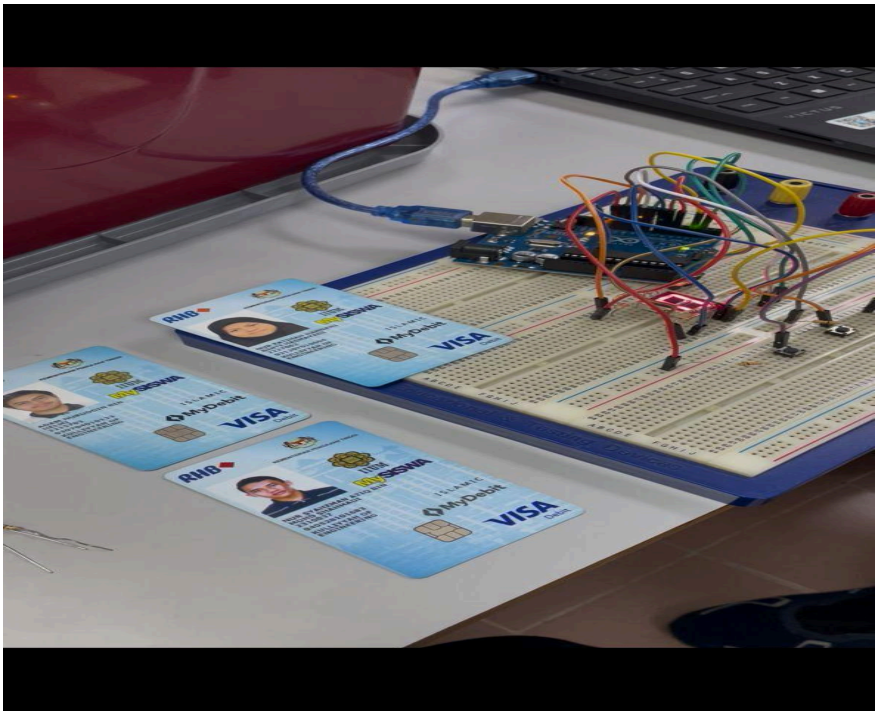


Figure 1.

4.0 Methodology

1. Set up the Arduino Uno
2. Set up the code in Arduino IDE
3. Test the code
4. Code Snippet

4.1 Detail steps

1. Assemble the circuit based on the given setup instructions.

2. Upload the supplied Arduino program to your Arduino Uno board.

3. Launch the Serial Monitor in the Arduino IDE.

4. Press the increment button to raise the count. The 7-segment display should cycle through numbers 0 to 9 in order.

5. Press the reset button to bring the count back to 0.

4.2 Programming codes

Arduino

```
// Define the pins for each segment

const int segmentA = 3;

const int segmentB = 2;

const int segmentC = 8;

const int segmentD = 7;

const int segmentE = 6;

const int segmentF = 10;

const int segmentG = 5;


// Define button pins

const int buttonInc = 11;
```

```
const int buttonReset = 12;


// Variables
int number = 0;
bool lastIncState = HIGH;
bool lastResetState = HIGH;


// 7-segment display digits (for common cathode)
// HIGH = ON, LOW = OFF
const int digits[10][7] = {
  {HIGH, HIGH, HIGH, HIGH, HIGH, HIGH, LOW},   // 0
  {LOW, HIGH, HIGH, LOW, LOW, LOW, LOW},       // 1
  {HIGH, HIGH, LOW, HIGH, HIGH, LOW, HIGH},    // 2
  {HIGH, HIGH, HIGH, HIGH, LOW, LOW, HIGH},    // 3
  {LOW, HIGH, HIGH, LOW, LOW, HIGH, HIGH},     // 4
  {HIGH, LOW, HIGH, HIGH, LOW, HIGH, HIGH},    // 5
  {HIGH, LOW, HIGH, HIGH, HIGH, HIGH, HIGH},   // 6
  {HIGH, HIGH, HIGH, LOW, LOW, LOW, LOW},      // 7
  {HIGH, HIGH, HIGH, HIGH, HIGH, HIGH, HIGH},  // 8
  {HIGH, HIGH, HIGH, HIGH, LOW, HIGH, HIGH}    // 9
};


// Segment pins array
```

```
int segments[] = {segmentA, segmentB, segmentC, segmentD, segmentE, segmentF, segmentG};

void setup() {
  // Initialize 7-segment pins as OUTPUT
  for (int i = 0; i < 7; i++) {
    pinMode(segments[i], OUTPUT);
  }

  // Initialize buttons with internal pull-up
  pinMode(buttonInc, INPUT_PULLUP);
  pinMode(buttonReset, INPUT_PULLUP);

  // Start showing 0
  displayDigit(number);
}

void loop() {
  bool incState = digitalRead(buttonInc);
  bool resetState = digitalRead(buttonReset);

  // Increment button pressed (active LOW)
  if (incState == LOW && lastIncState == HIGH) {
    number++;
```

```
    if (number > 9) number = 0;
    displayDigit(number);
    delay(250); // debounce
  }

  // Reset button pressed (active LOW)
  if (resetState == LOW && lastResetState == HIGH) {
    number = 0;
    displayDigit(number);
    delay(250); // debounce
  }

  lastIncState = incState;
  lastResetState = resetState;
}

// Function to display a digit (0–9)
void displayDigit(int num) {
  for (int i = 0; i < 7; i++) {
    digitalWrite(segments[i], digits[num][i]);
  }
}
```

5.0 Data Collection

Button Press	**Expected Display Output**	**Actual Display Output**
1st	1	1
2nd	2	2
3rd	3	3
4th	4	4
5th	5	5
6th	6	6
7th	7	7
8th	8	8
9th	9	9
10th	0	0

Button Press	**Expected Display Output**	**Actual Display Output**
1st	1	1
Reset button	0	0

6.0 Data Analysis

The data collected from the 7-segment display's response to Arduino input is essential in evaluating the success of the interface and control system. Correct numerical output on the display, corresponding to each button press, indicates that the digital signals sent by the Arduino are controlling each segment correctly. This confirms the correct understanding and implementation of digital logic principles and pin-level control using the microcontroller.

In short, the data directly illustrates how well the system met the objectives of the experiment using the Arduino to control the 7-segment display via user input, serving as practical evidence for understanding and using basic digital electronics and microcontroller programming.

7.0 Results

The experiment successfully demonstrated the interfacing of a common cathode 7-segment display with an Arduino Uno. When the circuit and code were correctly implemented, the display showed numerical values from 0 to 9 sequentially as the increment button was pressed, while the reset button returned the display to zero.

Each number was displayed clearly, and the transitions between digits were smooth and immediate, indicating proper logic control. The use of 220-ohm resistors effectively limited the current, preventing overheating or flickering.

No delays or errors were observed during operation. The system responded promptly to button inputs, confirming that both hardware wiring and Arduino programming were functioning as expected.

8.0 Discussion

The results from our experiment confirm the correct operation of the digital logic system. The Arduino successfully controlled each 7 segment of the display through digital output pins, demonstrating the practical use of binary logic in visual output devices.

Each digit was formed by activating the appropriate combination of segments according to standard logic patterns. The system's accuracy and responsiveness prove that the circuit design and code structure were correctly implemented.

A possible minor issue could arise from loose breadboard connections or debouncing of pushbuttons, which might cause unintended multiple counts. This can be improved by adding a software debounce delay or using hardware filters.This experiment reinforces understanding of how logic gates, microcontroller I/O, and current-limiting resistors work together in digital display applications.

9.0 Conclusion

In conclusion, our experiment on the 7-segment display successfully demonstrated the implementation of digital logic control using an Arduino Uno. The display accurately showed numbers from 0 to 9 based on button inputs, confirming that the circuit wiring, resistor configuration, and programming logic functioned as intended. This confirms the hypothesis that a properly interfaced microcontroller can produce correct numerical outputs through digital logic operations. The experiment provided a practical understanding of how binary logic is applied to real hardware systems. It also emphasized the importance of correct circuit connections, suitable resistor values, and reliable code structure in ensuring stable operation. Overall, the experiment achieved its objectives by reinforcing fundamental concepts in digital electronics and providing a

solid foundation for future projects involving more complex display systems or logic-based microcontroller applications.

10.0 Recommendations

For future iterations of this experiment, it is recommended to pay close attention to all circuit connections to ensure that each component is correctly wired to its respective Arduino pin. Understanding the pin configuration is essential for accurate coding and system functionality. The coding process can also be optimized by using shorter and more efficient code structures, which will help save time and reduce potential errors. It is also advisable to strengthen programming skills through consistent practice rather than relying heavily on AI-generated solutions, which allow for a deeper understanding and greater flexibility in troubleshooting. Additionally, the troubleshooting techniques learned during this experiment should be applied in future projects to quickly identify and correct wiring or logic errors, leading to smoother experimental progress and more reliable results.

11.0 References

- Week 2: Digital Logic System: Basic Logic Gates, Electronic Circuit Interfacing, Basic ALU, 7 segment display, ICs based interfacing application (ver 2) https://drive.google.com/drive/folders/1QecGGN96D76UL2gwBjevFwiEXv2UEniO?usp=sharing

Acknowledgement

Thank you to our lecturer, Dr. Zulkifli Bin Zainal Abidin & Dr.Wahju Sediono, and his teaching assistant for their invaluable guidance and support when completing the tasks. We are deeply appreciate their time and patience to our future academic growth and progress.

Student's Declaration

Certificate of Originality and Authenticity

We hereby certify that we are collectively responsible for the work presented in this report. The content reflects our original work, except where specific references or acknowledgements have been made. No part of this report has been completed or submitted by individuals or sources not identified within.

Furthermore, we confirm that this report is the result of a collaborative effort, with contributions made by all group members. The extent of each individual's contribution is documented within this certificate.

We also hereby certify that we have read and understand the content of the total report and no further improvement on the reports is needed from any of the individual's contributors to the report.

We therefore, agreed unanimously that this report shall be submitted for marking and this final printed report has been verified by us.

Name: ADAM NURUDDIN BIN HELMI	Read	[/]
Matric Number: 2215783	Understand	[/]
Signature: *ADAM*	Agree	[/]

Name: NUR FATIHAH HANNANI BINTI HASMAYADI	Read	[/]
Matric Number: 2227062	Understand	[/]
Signature: *FAT*	Agree	[/]

Name: NUR SYAHZMAN AZIQ BIN MOHD SHAHMADI	Read	[/]
Matric Number: 2310037	Understand	[/]
Signature: *AZIQ*	Agree	[/]