MINI PROJECT-1

TOPIC: Compare two digital numbers (two bits) and display the result in a 7-Segment Display.

1.1 INTRODUCTION

Digital systems often require the comparison of numerical values and the clear presentation of the results. This project involves the design of a digital circuit to compare two 2-bit binary numbers and display the outcome on a 7-segment display. This setup is essential in numerous digital electronics applications, such as calculators, digital meters, and other display systems.

The focus of the project is on creating a circuit that accepts two 2-bit binary inputs, A and B. The circuit will compare these inputs to determine if the first number (A) is greater than, less than, or equal to the second number (B). Based on the comparison, the circuit will drive a 7-segment display to indicate the result: 'A' if the first number is greater, 'b' if the second number is greater, and 'E' if the numbers are equal.

To accomplish this, the project uses basic digital logic gates like AND, OR, NOT, and XOR to implement the comparison logic. Instead of utilizing a pre-made 7-segment display driver IC, the project involves designing a custom logic circuit to control the 7-segment display. This custom logic is developed by creating a truth table for each segment of the display and simplifying the logic using Karnaugh maps.

This hands-on project not only reinforces the understanding of digital logic principles but also enhances problem-solving skills by requiring the application of theoretical knowledge to create a functional hardware implementation. The project illustrates how fundamental concepts of digital electronics can be applied to develop practical and effective digital systems.

1.2 LITERATURE SURVEY

The domain of digital comparators is a well-established area in digital electronics, forming the foundation for various computational and display systems. Digital comparators play a crucial role in arithmetic logic units (ALUs), digital signal processors (DSPs), and numerous other applications requiring binary number comparisons. The fundamental task of these devices is to determine whether one binary number is greater than, less than, or equal to another.

Several research papers and textbooks offer comprehensive insights into the design and implementation of digital comparators. For instance, "Digital Design" by M. Morris Mano and

Michael D. Ciletti provides an extensive overview of the principles and types of comparators, including single-bit and multi-bit comparators, along with their practical applications in digital systems [1].

In their paper, Sharma et al. (2019) explore the design and efficiency optimization of digital comparators, emphasizing the importance of speed and power consumption in contemporary digital circuits [2]. They examine various techniques to enhance comparator performance, which is vital for high-speed computing and real-time processing applications.

A study by Patil et al. (2018) focuses on the implementation of comparators using different logic families and their impact on power and delay metrics [3]. This research highlights the significance of selecting suitable logic gates and optimization methods to achieve desired performance levels in specific applications.

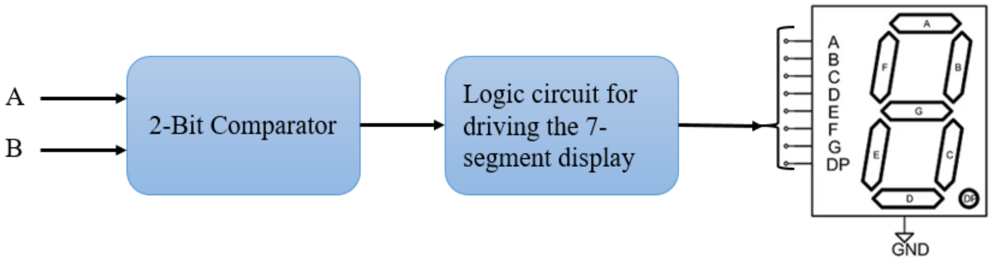
Moreover, a practical approach to teaching digital logic design is presented in "Digital Systems: Principles and Applications" by Ronald J. Tocci and Neal S. Widmer. This textbook emphasizes hands-on projects involving basic logic gates and small-scale integration (SSI) components, akin to the approach taken in this project [4]. It provides valuable insights into the educational benefits of engaging students in practical, hardware-based digital logic design projects.

Previous works have primarily concentrated on more complex multi-bit comparators or different display methods such as light-emitting diodes (LEDs) and liquid crystal displays (LCDs). However, this project aims to simplify the comparison process for educational purposes using basic digital circuits and a 7-segment display. The use of a 7-segment display is particularly advantageous for visualization in educational settings, as it provides a clear and straightforward representation of the comparison results.

The project leverages fundamental concepts of digital logic design, such as the use of truth tables and Karnaugh maps for logic simplification, to implement the comparator and display logic. By building the comparator using basic gates (AND, OR, NOT, XOR), the project offers a deeper understanding of the underlying mechanisms of digital comparison and display systems.

In summary, the literature provides a solid framework for understanding and implementing digital comparators. This project builds on these foundational concepts by applying them to a practical, hands-on design that emphasizes educational value and the core principles of digital electronics.

1.3 BLOCK DIAGRAM

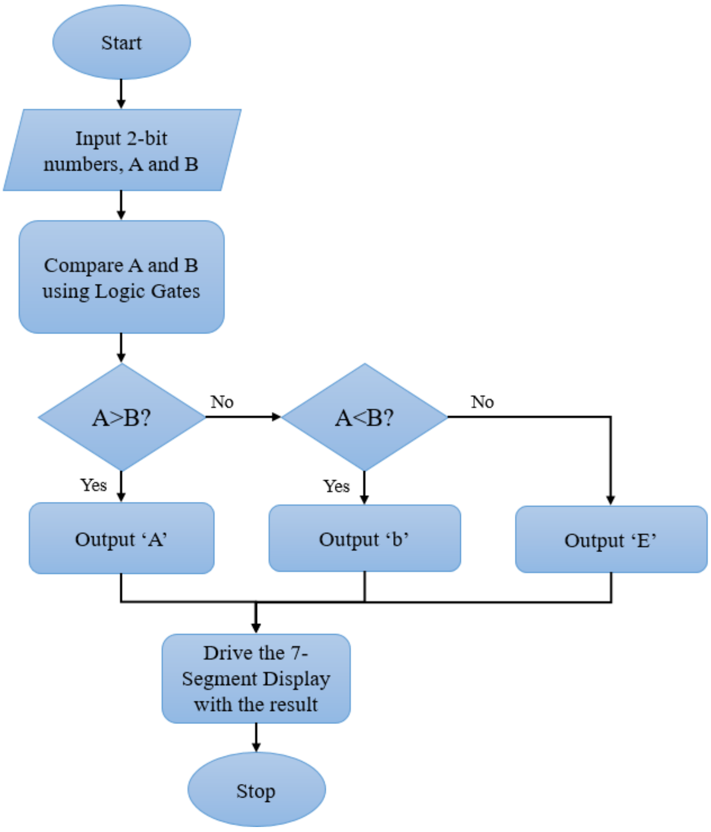


1.4 FLOWCHART/ALGORITHM

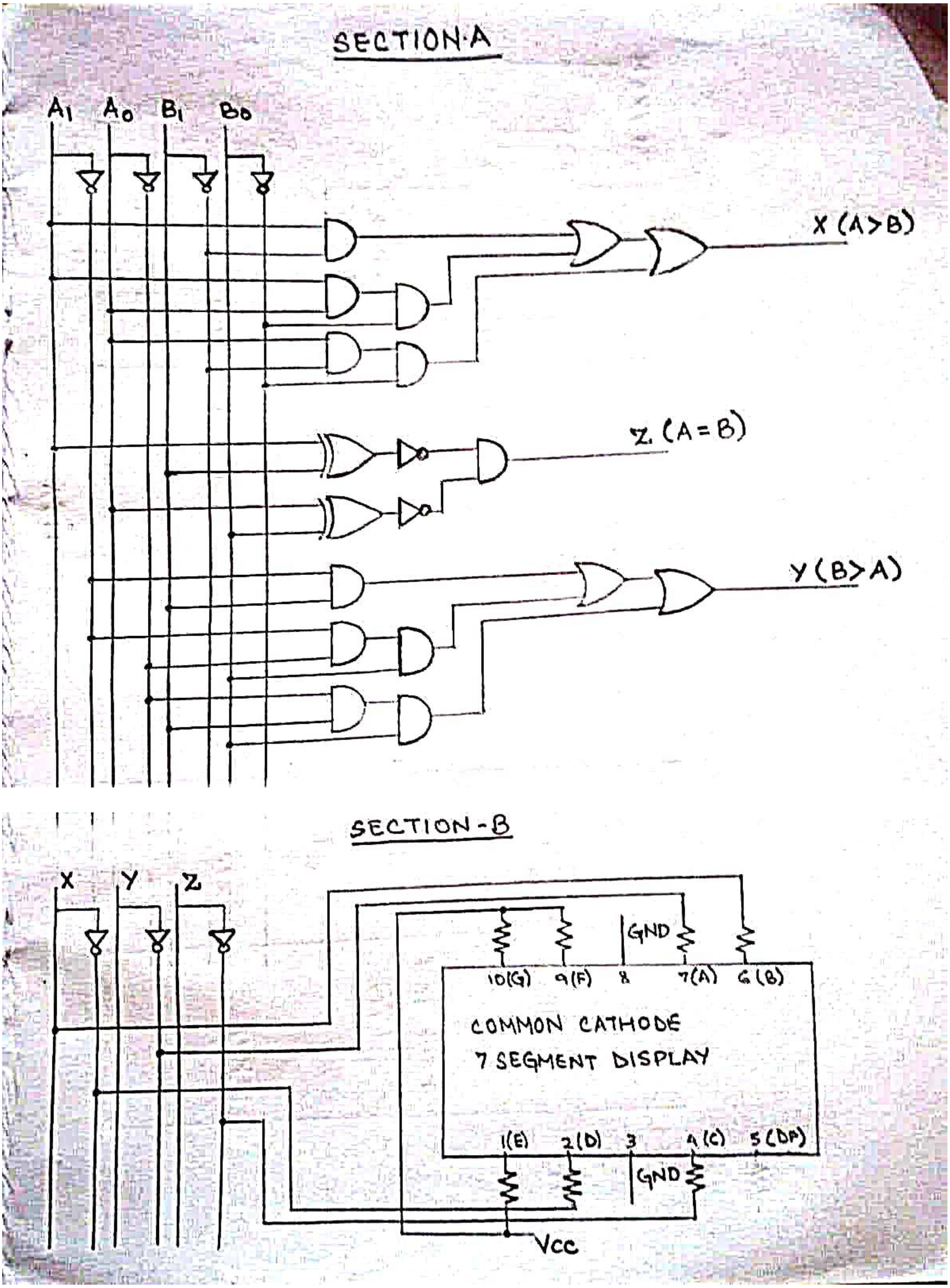
Algorithm:

1. Start
2. Input 2-bit numbers A and B
3. Compare A and B using logic gates
    - If A > B, set output to 'A'
    - If A < B, set output to 'b'
    - If A == B, set output to 'E'
4. Drive the 7-segment display with the result using custom logic
5. End

Flowchart:


Start
Input 2-bit numbers, A and B
Compare A and B using Logic Gates
A>B?
No
A<B?
No
Yes
Yes
Output 'A'
Output 'b'
Output 'E'
Drive the 7-Segment Display with the result
Stop

1.5 CIRCUIT DIAGRAM



1.6 COMPONENT LIST

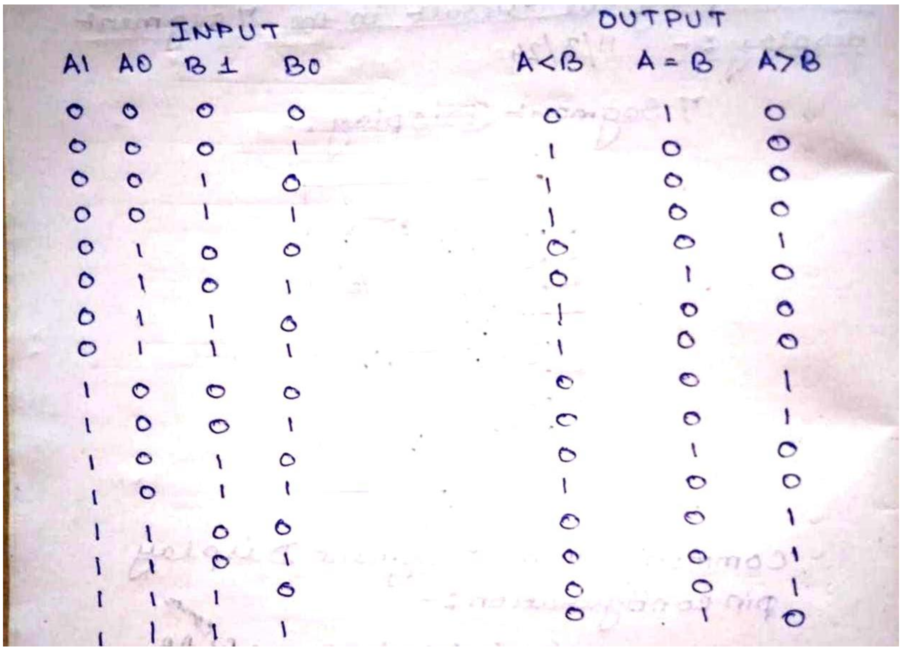
Component	Specification	Quantity
AND Gate	74LS08	3
OR Gate	74LS32	1
NOT Gate	74LS04	2
XOR Gate	74LS86	1
7-Segment Display	Common Cathode	1
Resistors	220Ω	7
Breadboard	-	2

1.7 DETAILED DESCRIPTION

The project entails using fundamental logic gates to compare two 2-bit numbers and designing a custom logic circuit to control the 7-segment display. This section offers a thorough explanation of the design process, the logic implementation, and the method by which the 7-segment display is managed using the outputs from the comparator.

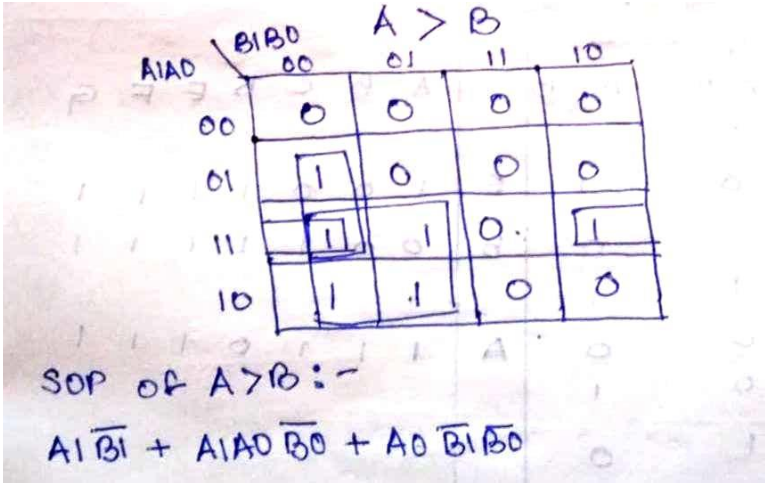
Comparator Design:

1. **Inputs:** The inputs to the comparator are two 2-bit numbers, represented as A (A1, A0) and B (B1, B0).
2. **Truth Table:**



INPUT				OUTPUT		
A1	A0	B1	B0	A<B	A=B	A>B
0	0	0	0	0	1	0
0	0	0	1	1	0	0
0	0	1	0	1	0	0
0	0	1	1	1	0	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	1	0	0
1	0	0	0	0	0	1
1	0	0	1	0	0	1
1	0	1	0	0	1	0
1	0	1	1	1	0	0
1	1	0	0	0	0	1
1	1	0	1	0	0	1
1	1	1	0	0	0	1
1	1	1	1	0	1	0

- **Greater Than (A > B):**
  - For A to be greater than B, the following conditions must be met:
    - A1 > B1, or
    - A1 == B1 and A0 > B0



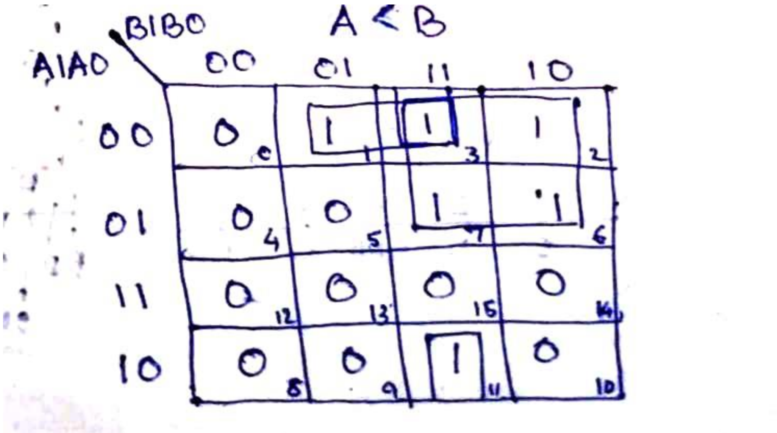
A > B

A1A0 \ B1B0	00	01	11	10
00	0	0	0	0
01	1	0	0	0
11	1	1	0	1
10	1	1	0	0

SOP of A>B :-

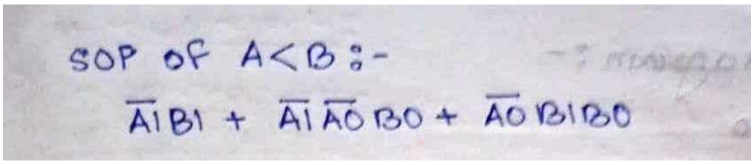
$A1\overline{B1} + A1A0\overline{B0} + A0\overline{B1}\,\overline{B0}$

- **Less Than (A < B):**
  - For A to be less than B, the following conditions must be met:
    - B1 > A1, or
    - B1 == A1 and B0 > A0



A < B

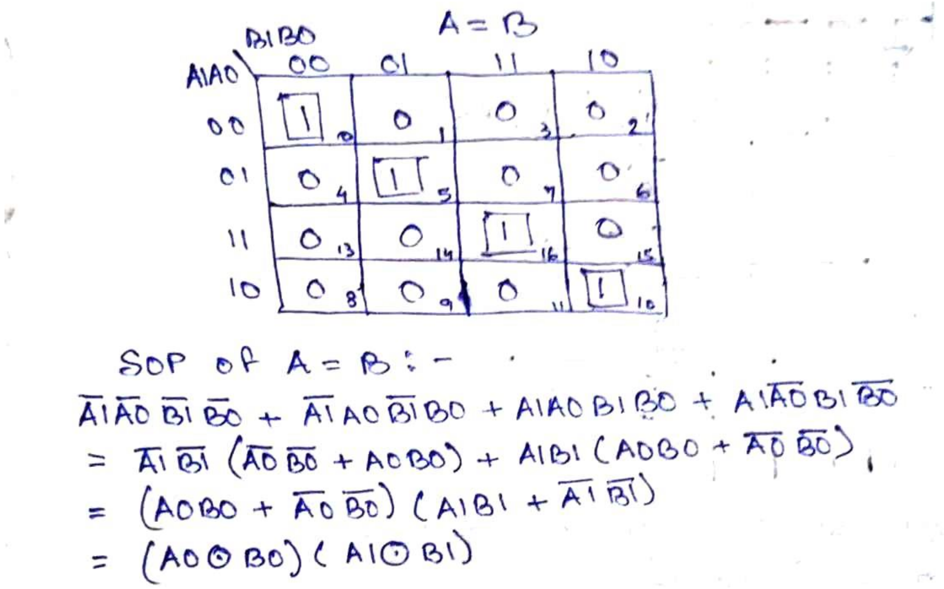
A1A0 \ B1B0	00	01	11	10
00	0_{0}	1_{1}	1_{3}	1_{2}
01	0_{4}	0_{5}	1_{7}	1_{6}
11	0_{12}	0_{13}	0_{15}	0_{14}
10	0_{8}	0_{9}	1_{11}	0_{10}



SOP of A<B :-

$\overline{A1}B1 + \overline{A1}\,\overline{A0}B0 + \overline{A0}B1B0$

- **Equal To (A == B):**
  - For A to be equal to B, the following conditions must be met:
    - A1 == B1 and A0 == B0



A = B

A1A0 \ B1B0	00	01	11	10
00	$\boxed{1}_{0}$	0_{1}	0_{3}	0_{2}
01	0_{4}	$\boxed{1}_{5}$	0_{7}	0_{6}
11	0_{13}	0_{14}	$\boxed{1}_{16}$	0_{15}
10	0_{8}	0_{9}	0_{11}	$\boxed{1}_{10}$

SOP of A = B :-

$$\overline{A1}\,\overline{A0}\,\overline{B1}\,\overline{B0} + \overline{A1}\,A0\,\overline{B1}\,B0 + A1A0B1B0 + A1\overline{A0}\,B1\overline{B0}$$

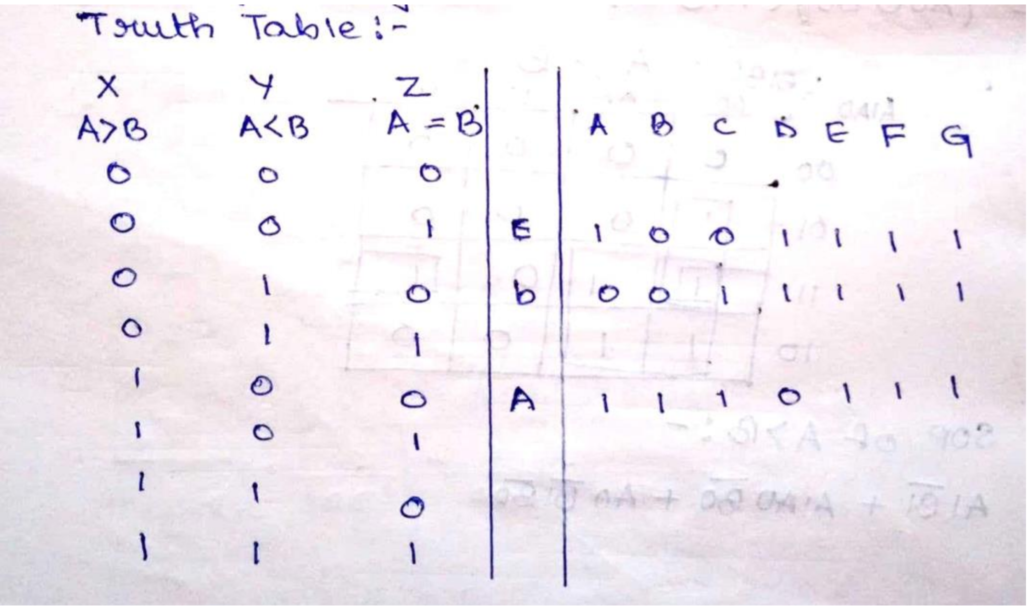
$$= \overline{A1}\,\overline{B1}\,(\overline{A0}\,\overline{B0} + A0B0) + A1B1\,(A0B0 + \overline{A0}\,\overline{B0})$$

$$= (A0B0 + \overline{A0}\,\overline{B0})\,(A1B1 + \overline{A1}\,\overline{B1})$$

$$= (A0 \odot B0)\,(A1 \odot B1)$$

7-Segment Display Logic:

1. **Truth Table and Karnaugh Maps:**
   - The outputs of the comparator are used to drive a 7-segment display. Each segment (a, b, c, d, e, f, g) of the display must be controlled based on the comparison result (A > B, A < B, A == B).
   - A truth table is created for each segment, specifying the desired output for the characters 'A', 'b', and 'E'.
   - Karnaugh maps are used to simplify the logic expressions for each segment.

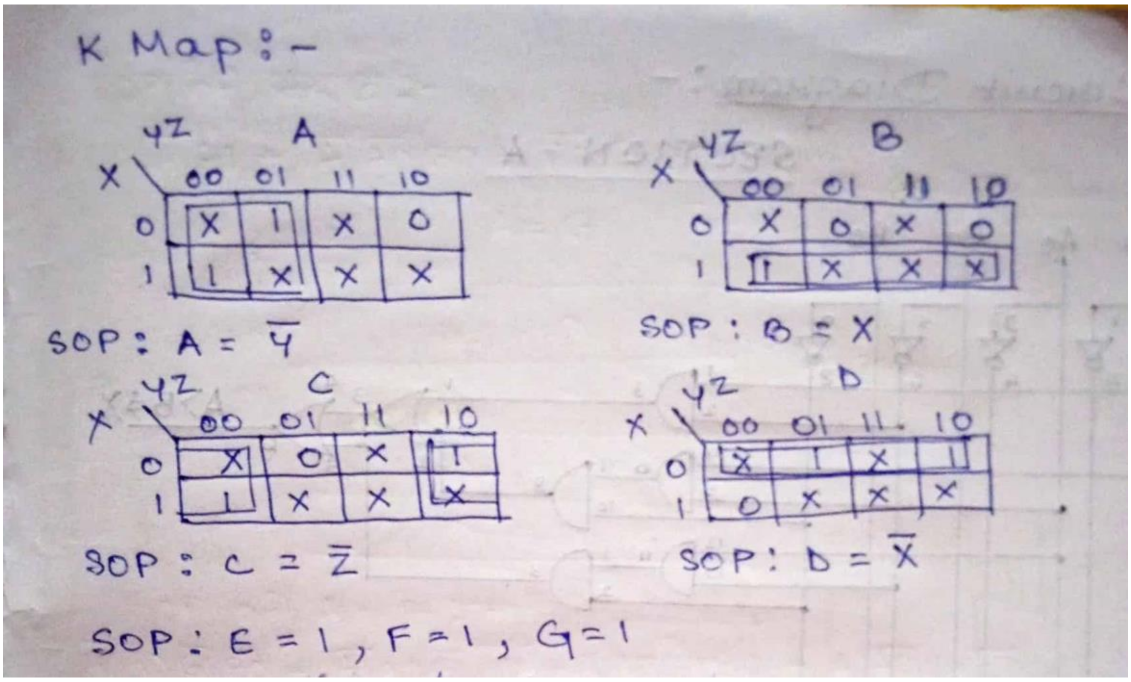


Truth Table :-

X	Y	Z		A	B	C	D	E	F	G
A>B	A<B	A = B								
0	0	0								
0	0	1	E	1	0	0	1	1	1	1
0	1	0	b	0	0	1	1	1	1	1
0	1	1								
1	0	0	A	1	1	1	0	1	1	1
1	0	1								
1	1	0								
1	1	1								

2. **Logic Implementation:**
   - **Segment 'a':** It is lit for the characters 'A' and 'E'.

- **Segment 'b':** It is lit for the characters 'A'.
- **Segment 'c':** It is lit for the characters 'A' and 'b'.
- **Segment 'd':** It is lit for the characters 'b' and 'E'.
- **Segment 'e':** It is lit for all the characters 'A', ‘b’ and 'E'.
- **Segment 'f':** It is lit for all the characters 'A', ‘b’ and 'E'.
- **Segment 'g':** It is lit for all the characters 'A', ‘b’ and 'E'.



K Map:-

A

X \ YZ	00	01	11	10
0	X	1	X	0
1	1	X	X	X

SOP: A = $\overline{Y}$

B

X \ YZ	00	01	11	10
0	X	0	X	0
1	1	X	X	X

SOP: B = X

C

X \ YZ	00	01	11	10
0	X	0	X	1
1	1	X	X	X

SOP: C = $\overline{Z}$

D

X \ YZ	00	01	11	10
0	X	1	X	1
1	0	X	X	X

SOP: D = $\overline{X}$

SOP: E = 1, F = 1, G = 1

3. **Connecting Logic to the 7-Segment Display:**
   - The outputs of the simplified logic expressions for each segment are connected to the corresponding segments of the 7-segment display.
   - This direct control of the display segments using logic gates eliminates the need for a dedicated 7-segment display driver IC.

Circuit Implementation:

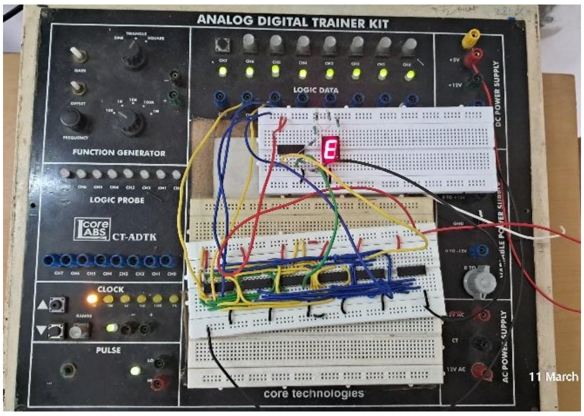
- The comparator and 7-segment display logic circuits are implemented on a breadboard or PCB using basic components such as resistors, wires, and logic gates (e.g., 7408, 7432, 7404, 7486).
- The inputs A and B are provided through switches or a microcontroller, and the comparison result is visually displayed on the 7-segment display.

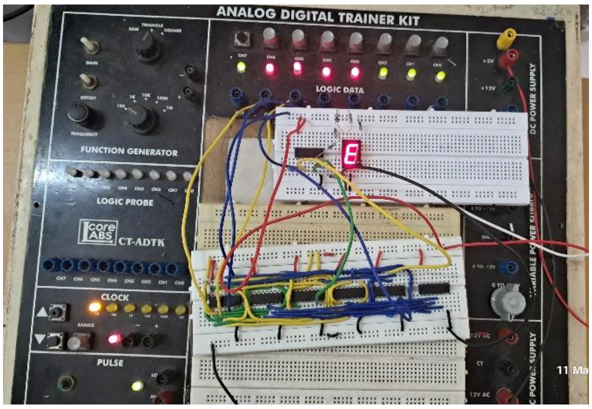
In conclusion, this project involves designing a digital comparator using basic logic gates and implementing custom logic to drive a 7-segment display. The use of truth tables and Karnaugh maps ensures that the logic is optimized and efficient. This project not only enhances understanding of digital logic design but also provides practical experience in constructing and troubleshooting digital circuits.

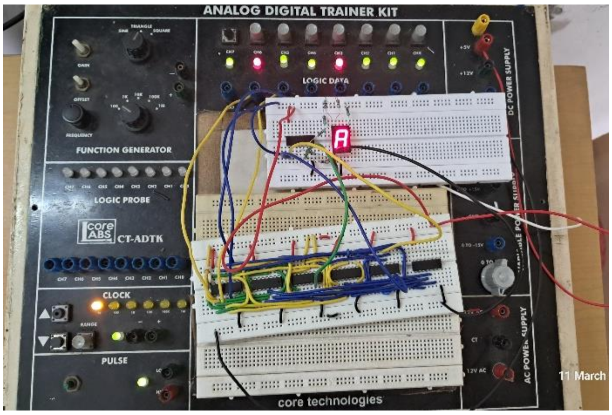
1.8 RESULT AND ANALYSIS

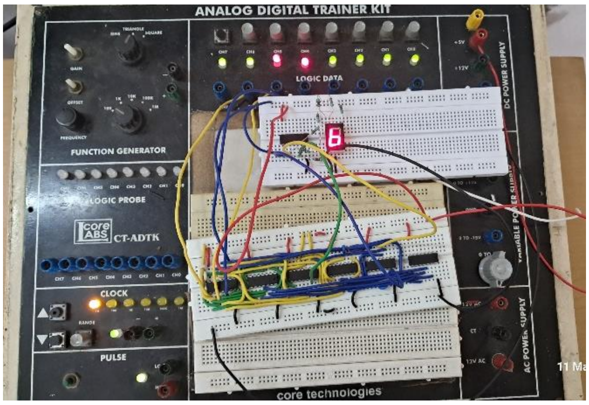
The project involves utilizing basic logic gates to compare two 2-bit numbers and designing a custom logic circuit to drive a 7-segment display. This section offers a comprehensive explanation of the design process, the logic implementation, and the method by which the 7-segment display is controlled using the outputs from the comparator.

The constructed circuit effectively compares two 2-bit numbers and displays the result on a 7-segment display. The circuit was tested with all possible combinations of 2-bit inputs (00, 01, 10, 11) for both A and B, validating the accuracy of the comparison and display logic. When A was greater than B, the display showed 'A'; when B was greater than A, the display showed 'b'; and when A was equal to B, the display showed 'E'.









Testing Procedure:

1. **Input Combinations:** Each combination of the 2-bit numbers A and B was systematically tested. The test cases included:
   - A = 00, B = 00; A = 00, B = 01; A = 00, B = 10; A = 00, B = 11

- A = 01, B = 00; A = 01, B = 01; A = 01, B = 10; A = 01, B = 11
- A = 10, B = 00; A = 10, B = 01; A = 10, B = 10; A = 10, B = 11
- A = 11, B = 00; A = 11, B = 01; A = 11, B = 10; A = 11, B = 11

2. **Comparison Logic Verification:**
   - For each input combination, the comparison logic was verified by checking the outputs of the logic gates.
   - The conditions for A > B, A < B, and A = B were confirmed by monitoring the intermediate signals using a logic analyzer.
3. **Display Output Verification:**
   - The output on the 7-segment display was observed for each input combination.
   - The display correctly showed 'A' when A > B, 'b' when B > A, and 'E' when A = B.

Detailed Results:

1. **A > B Scenarios:**
   - Inputs: A = 10, B = 01; Display Output: 'A'
   - Inputs: A = 11, B = 10; Display Output: 'A'
   - The segments lit for 'A' matched the expected configuration based on the simplified logic expressions.
2. **A < B Scenarios:**
   - Inputs: A = 01, B = 10; Display Output: 'b'
   - Inputs: A = 00, B = 01; Display Output: 'b'
   - The segments lit for 'b' matched the expected configuration based on the simplified logic expressions.
3. **A = B Scenarios:**
   - Inputs: A = 10, B = 10; Display Output: 'E'
   - Inputs: A = 01, B = 01; Display Output: 'E'
   - The segments lit for 'E' matched the expected configuration based on the simplified logic expressions.

Analysis:

The project's success is attributed to its systematic approach to logic design and implementation. By using truth tables and Karnaugh maps, the logic was efficiently simplified, ensuring minimal gate usage while maintaining precise functionality. This project demonstrates the practical application of theoretical concepts in digital electronics, offering a robust learning experience.

In conclusion, the successful implementation and testing of this digital comparator project affirm its educational and practical significance. The project underscores the importance of fundamental digital logic design and lays the groundwork for more complex digital systems. Through this hands-on project, we gained valuable insights into the intricacies of digital comparisons and display mechanisms, enhancing our overall understanding and skills in digital electronics.

1.9 CONCLUSION

This project showcases a practical application of digital logic design by comparing two 2-bit numbers and displaying the result on a 7-segment display. It provides hands-on experience with basic logic gates and the process of designing custom logic circuits using truth tables and Karnaugh maps, which are essential skills in digital electronics.

By undertaking this project, we gained a deeper understanding of the functioning of digital comparators and how to translate logical conditions into physical outputs. The use of basic gates (AND, OR, NOT, XOR) in constructing the comparator circuit offers foundational insights into digital design, reinforcing theoretical concepts through practical application.

Additionally, the project's focus on deriving custom logic for the 7-segment display driver, rather than using a pre-made IC, highlights the importance of problem-solving and critical thinking in electronics. We learned to:

- Construct truth tables to define the logic for each segment of the display.
- Simplify these logical expressions using Karnaugh maps, a fundamental technique for minimizing Boolean functions.
- Implement and troubleshoot these simplified expressions using discrete logic gates.

The successful completion of this project highlights the effective integration of theory and practice. It illustrates how basic digital components can be combined to develop more complex and functional systems. The project also underscores the importance of accuracy in digital design, as even minor mistakes in logic implementation can result in incorrect display outputs.

Furthermore, this project acts as a foundation for more advanced digital systems design. The skills gained here are directly applicable to larger-scale digital design tasks, such as creating ALUs, memory address decoders, and other essential components of microprocessors and microcontrollers. Grasping the basics of digital comparison and display logic also paves the way for exploring programmable logic devices (PLDs) and field-programmable gate arrays (FPGAs), where these principles are implemented on a much larger and more complex scale.

In summary, this project on comparing two digital numbers and showcasing the result on a 7-segment display successfully connects theoretical knowledge with practical application. It fosters crucial skills in logic design, problem-solving, and circuit implementation, rendering it an invaluable educational resource in the domain of digital electronics.

1.10 REFERENCES

1. Mano, M. Morris, and Michael D. Ciletti. "Digital Design." Pearson Education.
2. Sharma, R., Gupta, S., & Mehra, R. (2019). "Design and Optimization of Digital Comparator." Journal of Electronic Design Technology.
3. Patil, R., Desai, S., & Kulkarni, S. (2018). "Implementation of Digital Comparators using Different Logic Families." International Journal of Computer Applications.
4. Tocci, Ronald J., and Neal S. Widmer. "Digital Systems: Principles and Applications." Pearson.
5. https://www.geeksforgeeks.org/magnitude-comparator-in-digital-logic/
6. https://www.alldatasheet.com/datasheet-pdf/pdf/5638/MOTOROLA/74LS04.html
7. https://www.alldatasheet.com/datasheet-pdf/pdf/12619/ONSEMI/74LS08.html
8. https://www.alldatasheet.com/datasheet-pdf/pdf/5707/MOTOROLA/74LS32.html
9. https://www.alldatasheet.com/datasheet-pdf/pdf/46213/SLS/74LS86.html