UNIVERSITY
OF WOLLONGONG
IN DUBAI

Assignment Cover Sheet

Student Name	Student number	Student submitting work
[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	
[redacted]	[redacted]	

Subject code and name	ECTE233 – Digital Hardware
Subject Coordinator	Dr. Obada Al Khatib
Title of Assignment	ECTE233 Project Report
Date and time due	[redacted]
Total number of pages	[redacted]
Total number of words	[redacted]

Student declaration and acknowledgment

By submitting this assignment online, the submitting student declares on behalf of the team that:

1. All team members have read the subject outline for this subject, and this assessment item meets the requirements of the subject detailed therein.
2. This assessment is entirely our work, except where we have included fully documented references to the work of others. The material in this assessment item has yet to be submitted for assessment.
3. Acknowledgement of source information is by the guidelines or referencing style specified in the subject outline.
4. All team members know the late submission policy and penalty.
5. The submitting student undertakes to communicate all feedback with the other team members.

4-bit Custom Arithmetic Logic Unit

ECTE233 Project Report

ECTE233 – Digital Hardware

University of Wollongong in Dubai
School of Engineering

Table of Contents

Table of Figures

Introduction

The rapid evolution of digital electronics has made arithmetic logic units (ALUs) fundamental components in modern computing systems. This project focuses on the design and implementation of a custom 4-bit ALU using fundamental logic gates, providing hands-on experience with digital circuit design principles and Boolean algebra applications.

The proposed ALU implements four distinct operations controlled by two operation selection bits (OP1:0), processing two 4-bit inputs (A3:0 and B3:0) to produce a 4-bit output (S3:0) along with an overflow detection bit (OF). The design is restricted to basic logic gates (AND, OR, NOT, XOR, NAND, NOR, and XNOR).

This report encompasses the complete design process, from initial specification analysis through implementation and testing. The design methodology incorporates truth table development, Karnaugh map simplification, and systematic testing procedures to verify functionality.

Design

The ALU has 10 input bits and 5 output bits. The specific operation to be performed is determined by two operation bits (OP1:0). The nibbles A3:0 and B3:0 are the other inputs with which operations are performed. The result of the operation, represented by nibble S3:0 is generated using 4 output bits. To indicate if an overflow has occurred during the operation, a single output bit (OF) is utilized. The top-level diagram is shown in Figure 1.

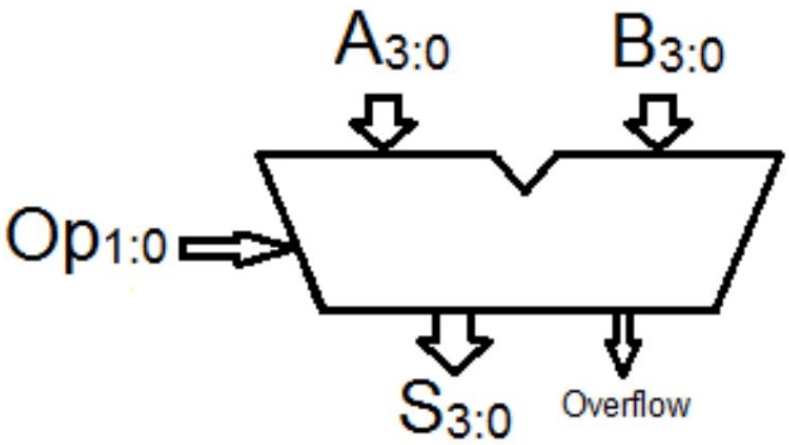


Figure 1 – Top-level diagram of the ALU

Inputs

The input interface for the 4-bit ALU utilizes single pole, double throw (SPDT) switches, configured to provide robust and reliable input selection. Each of the ten switches is designed with one terminal connected to the voltage source and the other to ground, enabling precise binary input control. The first four switches correspond to the A input nibble. Similarly, the next four switches define the B input nibble, providing independent control over the operands. The final two switches serve as control lines, determining the specific logical operation to be executed by the ALU. This switch-based input mechanism offers an intuitive method of input selection, with each switch's position directly translating to a logical high or low state. The design ensures clear differentiation between states, minimizing ambiguity in input representation and providing users with a direct, interactive means of manipulating the ALU's computational parameters.

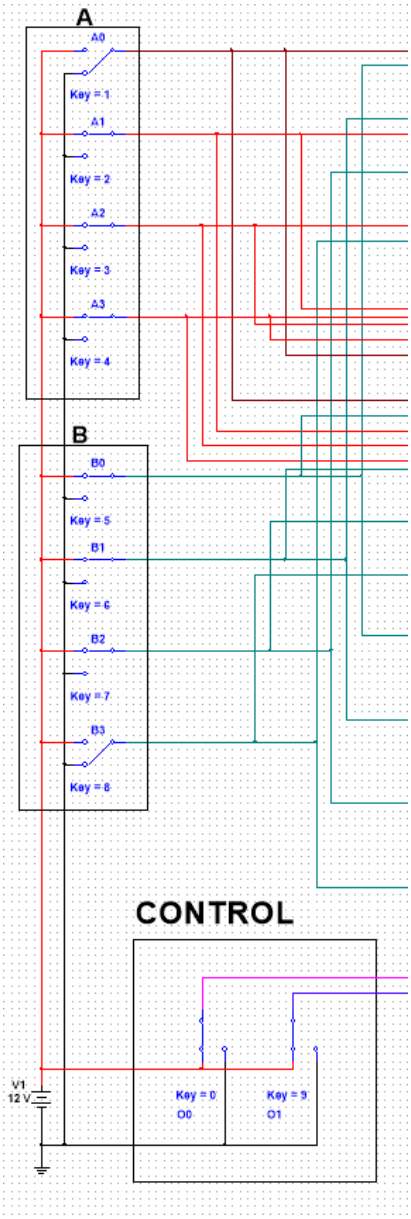


Figure 2 – 10 input switches representing A, B, and OP

Multiplexer

The multiplexer (MUX) illustrates a modular design approach, constructed using a configuration of four AND gates, two NOT gates, and one OR gate to achieve accurate signal selection. The modular architecture allows for independent routing of signals, with each component playing a specific, well-defined role in the signal selection process.

By inverting the control signals using NOT gates and then utilizing these complemented and original signals as inputs to the AND gates, the circuit creates a flexible logic network capable of routing the appropriate input to the output, based on control bits. Each AND gate receives three inputs: one from a specific input line (the data) and two from control lines, ensuring that only the selected input is activated, while all others remain disabled.

The 4 AND gates are connected to an OR gate, creating a compact and efficient signal routing mechanism. This modular design enables precise control, where the control bits determine which input is passed through to the output by selectively enabling the corresponding AND gate while keeping all other gates in a disabled state. This approach not only enhances the circuit's flexibility but also simplifies potential future modifications or expansions of the ALU's functionality.

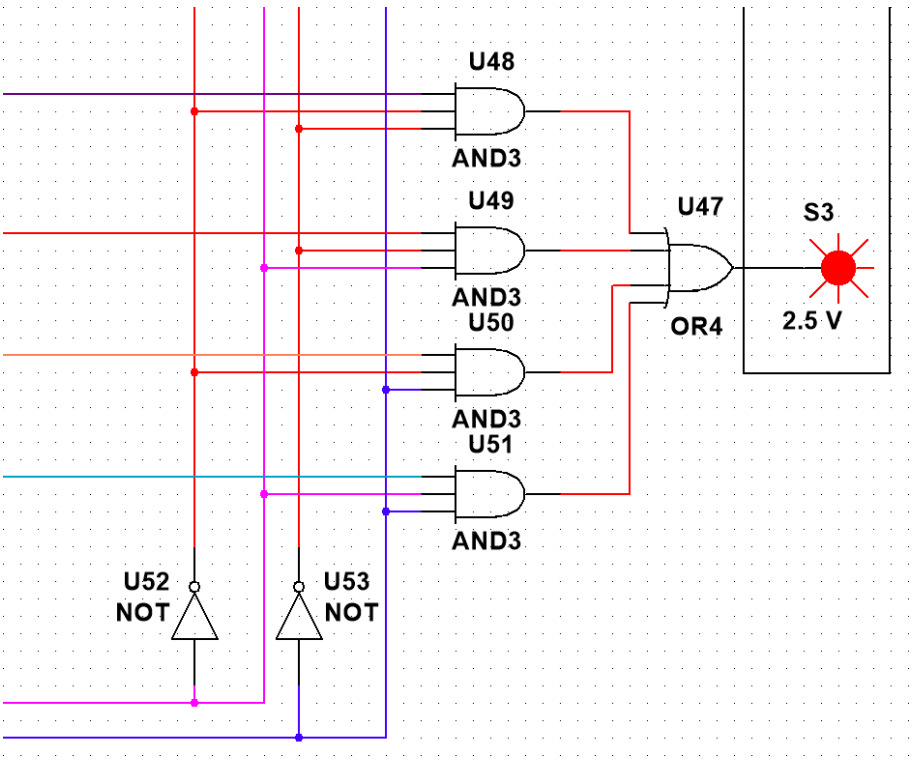


Figure 3 – Gate Implementation of the multiplexer

As seen in Figure 3, the AND gates receive three inputs – two control lines, and one data input from the operations that have been performed in the ALU.

Operation 0 – A AND B

The A AND B operation was implemented using four individual AND gates, one for each corresponding bit of inputs A and B. Each AND gate receives the corresponding bits of inputs A and B, producing an output S that directly represents the bitwise AND of those bits. The outputs of these AND gates are routed directly to the required multiplexer inputs. The implementation of the AND gates can be seen in Figure 4.

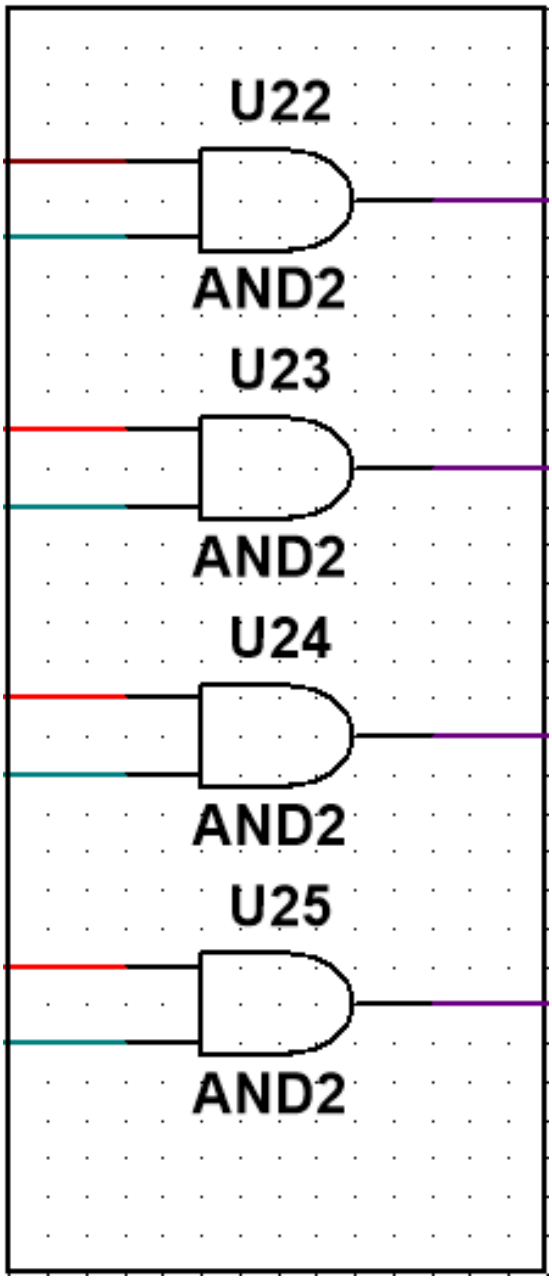


Figure 4 –Implementation of A AND B

Operation 1 – A1, A0, A3, A2

The input shuffling operation S = A1, A0, A3, A2 is implemented using a straightforward wire connection strategy, leveraging the multiplexers for each output bit. The routing network is constructed by directly connecting the input signals A1, A0, A3, and A2 to specific input ports of the multiplexers. Each 4-to-1 multiplexer is configured to select the appropriate input signal based on a fixed routing pattern, where the selection is hardwired to achieve the desired bit rearrangement. The multiplexer selection lines are statically configured to route A1 to S3, A0 to S2, A3 to S1, and A2 to S0, ensuring a deterministic signal path. This approach provides a direct mechanism for signal reordering without requiring complex logic or additional computational overhead.

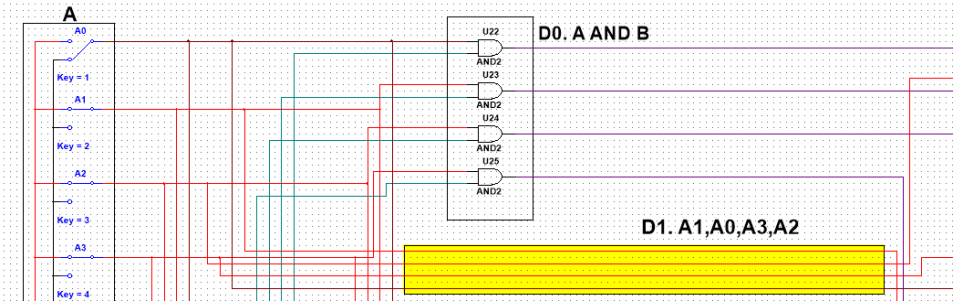


Figure 5 –Connection of the wires from the switches to the multiplexers

Operation 2 – A == B

The A == B comparison circuit utilizes XNOR gates to perform bit-wise equality checking, with an AND gate consolidating the results. Each corresponding bit from inputs A and B is connected to an XNOR gate, producing a high output only when bits match. The four XNOR outputs are then fed into an AND gate, which generates a high signal when all bits are identical. This design provides a straightforward and efficient method of comparing 4-bit inputs.

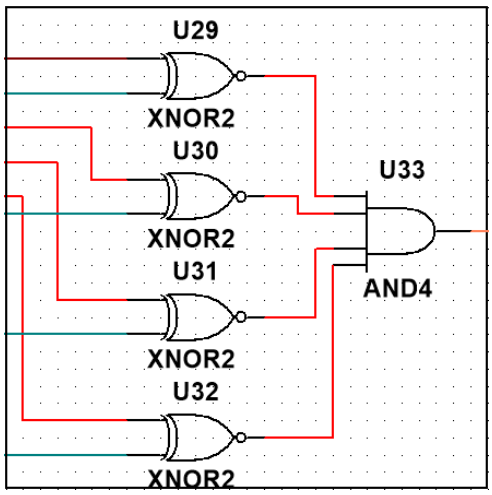


Figure 6 – Implementation of A == B using gates

Operation 3 – A + B

The 4-bit binary addition circuit is constructed using a cascaded half adder network that implements sequential binary addition through fundamental logic gates. Each half adder module comprises XOR and AND gates to compute the sum and carry-out for individual bit positions. The addition begins with the least significant bit, where a half adder processes A0 and B0, generating the least significant sum bit and a potential carry. Subsequent half adders in the chain process A1 and B1, A2 and B2, and A3 and B3, with each stage receiving the carry-out from the previous bit position. This interconnected design allows for sequential bit-wise addition, with the carry signal propagating through the higher-order bits. An additional logic block monitors the most significant bit's carry-out to detect potential overflow conditions, ensuring comprehensive handling of arithmetic operations across the 4-bit input range. An additional AND gate was incorporated into the overflow detection circuit, using two control signals to selectively enable the overflow indication based on the current operation.

During initial testing a critical design flaw was discovered, where the overflow bit would persist between different arithmetic operations, potentially providing false overflow indications. This unintended behaviour could lead to incorrect system interpretation of computational results across varying operational modes. To mitigate this issue, the last AND gate was added, allowing the overflow to display an output only during binary addition.

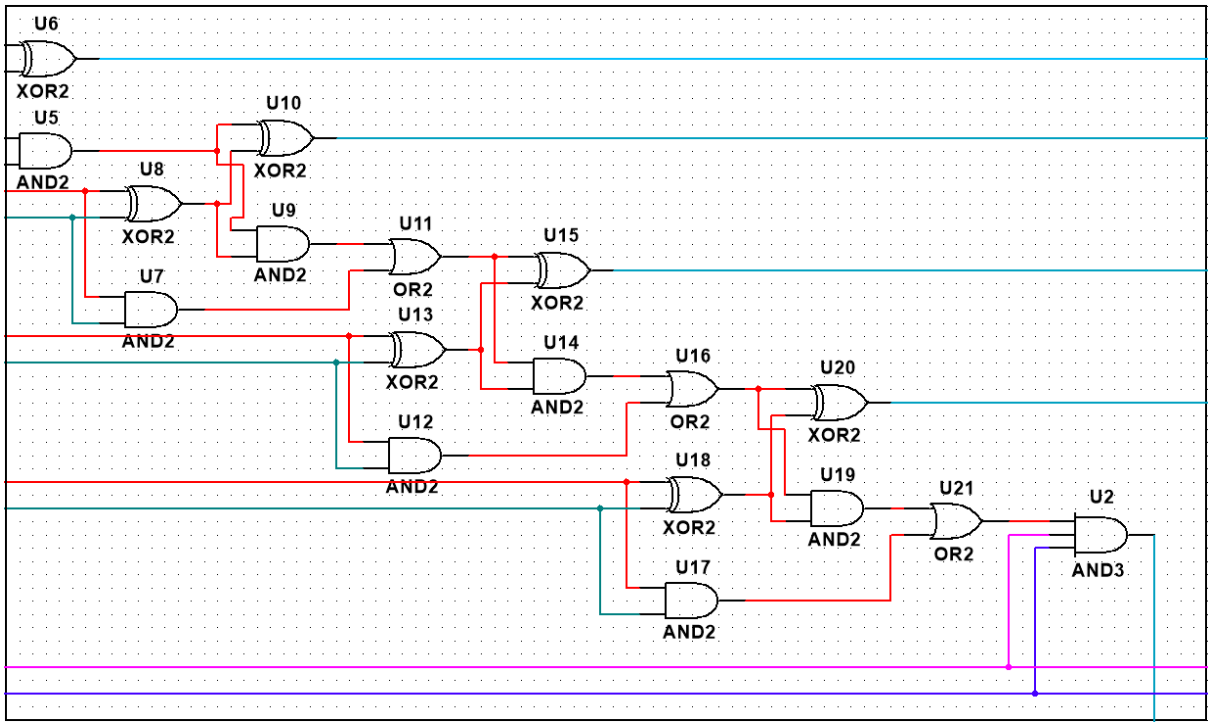


Figure 7 – 4-bit addition using Half Adders

Final Design

By systematically breaking down complex functions into basic logic gates, the final ALU design exhibits a highly modular and scalable structure that effectively executes four different operations. The distinct operational blocks—AND operation, bit shuffling, equality comparison, and binary addition—all demonstrate modularity since they are made to function as separate components that easily interface with the multiplexer-based selection system. In addition to making debugging and verification easier, this method makes it possible to expand to support more operations or broader bit widths with little change to the architecture.

By carefully considering signal routing and strategically reusing components, the design prioritizes efficiency, especially in the multiplexer implementation that acts as the main control hub. By adding extra control logic to avoid erroneous overflow indications between activities, the overflow detection system is a prime example of the design process's attention to detail. The explicit division of control and data pathways improves maintainability, and the design's consistent application of fundamental logic gates (AND, OR, NOT, XOR, and XNOR) guarantees dependability while preserving simplicity. The strong input interface made possible by SPDT switches and the systematic approach to circuit design produce an ALU architecture that is both adaptable and expandable, making it easy to modify for greater bit widths or more sophisticated applications.

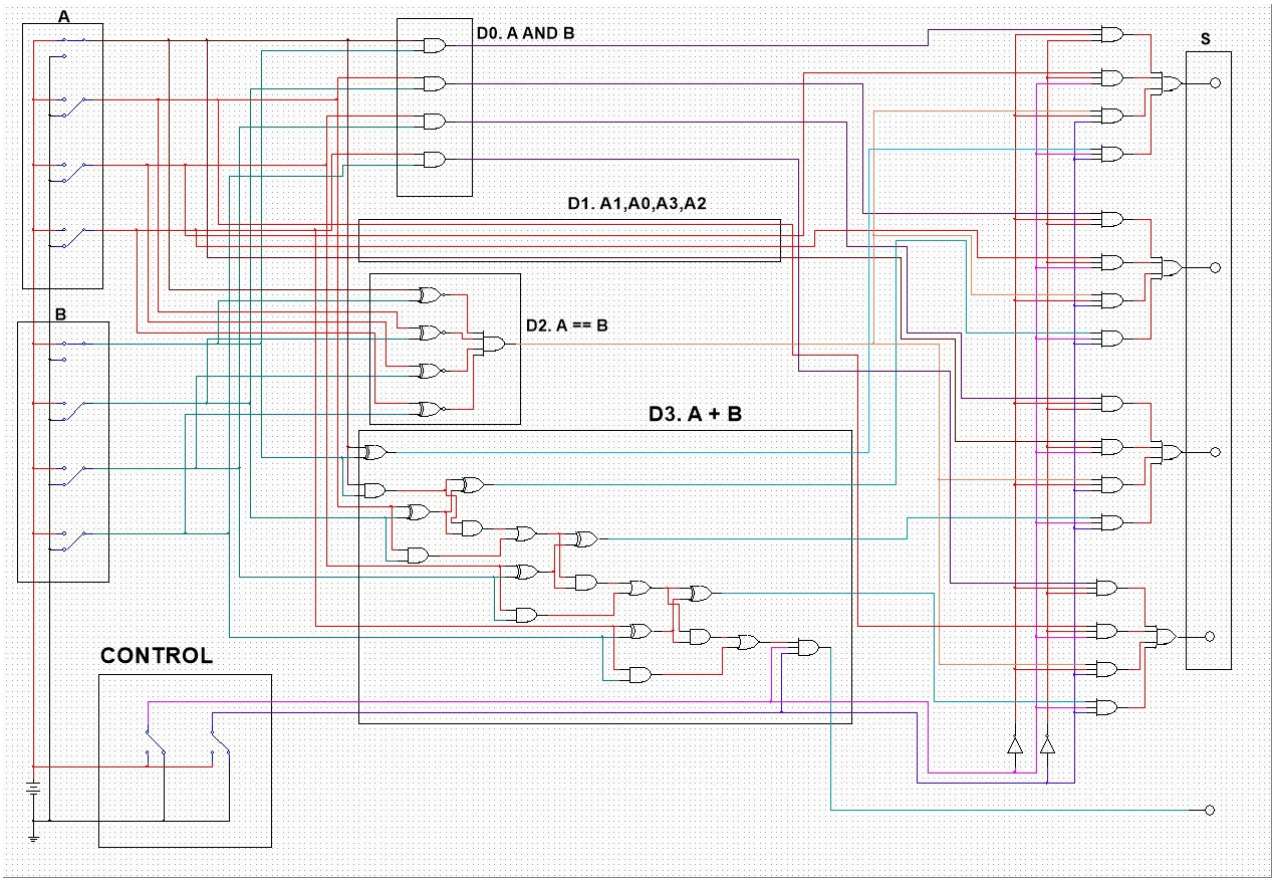


Figure 8 – The final design of the ALU

Truth Tables & Karnaugh Maps

The ALU's truth tables are extensive (2^8 = 256 rows) due to 8 inputs, making Karnaugh mapping impractical. However, by identifying patterns in the truth tables, we can analyse a subset to derive meaningful equations.

The following are the truth tables for the first operation

A_n	B_n	S_n
0	0	0
0	1	0
1	0	0
1	1	1

A \ B	0	1
0	0	0
1	0	1

$S_n = A_nB_n$ (where n = 0, 1, 2 or 3)

The K-Map for these truth tables will be the same, hence their functions will be the same.

For the second operation (A1, A0, A3, A2), the truth table is much shorter due to the introduction of X's (don't care's). The input of B doesn't matter in this case.

A3	A2	A1	A0	B3	B2	B1	B0	S3	S2	S1	S0
0	0	0	0	X	X	X	X	0	0	0	0
0	0	0	1	X	X	X	X	0	1	0	0
0	0	1	0	X	X	X	X	1	0	0	0
0	0	1	1	X	X	X	X	1	1	0	0
0	1	0	0	X	X	X	X	0	0	0	1
0	1	0	1	X	X	X	X	0	1	0	1
0	1	1	0	X	X	X	X	1	0	0	1
0	1	1	1	X	X	X	X	1	1	0	1
1	0	0	0	X	X	X	X	0	0	1	0
1	0	0	1	X	X	X	X	0	1	1	0
1	0	1	0	X	X	X	X	1	0	1	0
1	0	1	1	X	X	X	X	1	1	1	0
1	1	0	0	X	X	X	X	0	0	1	1
1	1	0	1	X	X	X	X	0	1	1	1
1	1	1	0	X	X	X	X	1	0	1	1
1	1	1	1	X	X	X	X	1	1	1	1

A3A2 \ A1A0	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	0	0	1	1
10	0	0	1	1

S3 = A1

A3A2 \ A1A0	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	1	1	1	1
10	1	1	1	1

S2 = A3

A3A2 \ A1A0	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	0	1	1	0
10	0	1	1	0

S2 = A0

A3A2 \ A1A0	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	1	1	1	1
10	0	0	0	0

S0 = A2

Like the first operation, the third operation (A == B) has a huge truth table, and by observation, the relevant subset is used for ascertaining the equation and K-map.

Again, the K-Map for these truth tables will be the same.

A_n	B_n	S_n
0	0	1
0	1	0
1	0	0
1	1	1

A \ B	0	1
0	1	0
1	0	1

$S_n = A_nB_n + A_n'B_n' = A_n \equiv B_n = A_n$ XNOR B_n
(where n = 0, 1, 2 or 3)

For the last operation, A + B, the circuit has been implemented using Half Adders. The truth table for one of the half adders is as follows:

A_n	B_n	Sum_n	$Carry_n$
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

A \ B	0	1
0	0	1
1	1	0

$S_n = A_n \oplus B_n$

(where n = 0, 1, 2 or 3)

$C_n = A_nB_n$

A \ B	0	1
0	0	0
1	0	1

Testing Procedure & Results

To ensure the accuracy of the ALU designed, extensive testing was done. Multiple tests were generated for each operation of the ALU, to ensure the results were accurate. This included edge cases, random numbers, and patterns. Multiple iterations of testing for each test case were conducted to ensure consistency in results. The circuit consistently produces the same output for the same input.

The testing methodology employed in this set of test cases follows a structured and systematic approach to ensure the ALU operates as expected. Each test case is designed to exercise a specific OPCODE (operation code) and a range of input values, including edge cases and boundary conditions.

For each test case, the methodology specifies the input operands, the expected output, and the actual output obtained from the ALU. This comprehensive approach allows for a thorough verification of the ALU's functionality and identification of any discrepancies between the expected and actual results. By executing multiple trials for each test case, the testing procedure ensures the consistency and reliability of the ALU's performance.

It is important to note that the switches are in the order of least to most significant bit, top to bottom. Hence, the topmost probe is S0 and the lowermost is the overflow (OF) bit.

The test cases have been displayed below.

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
00	A AND B	0000	0000	0000	0	0000	0

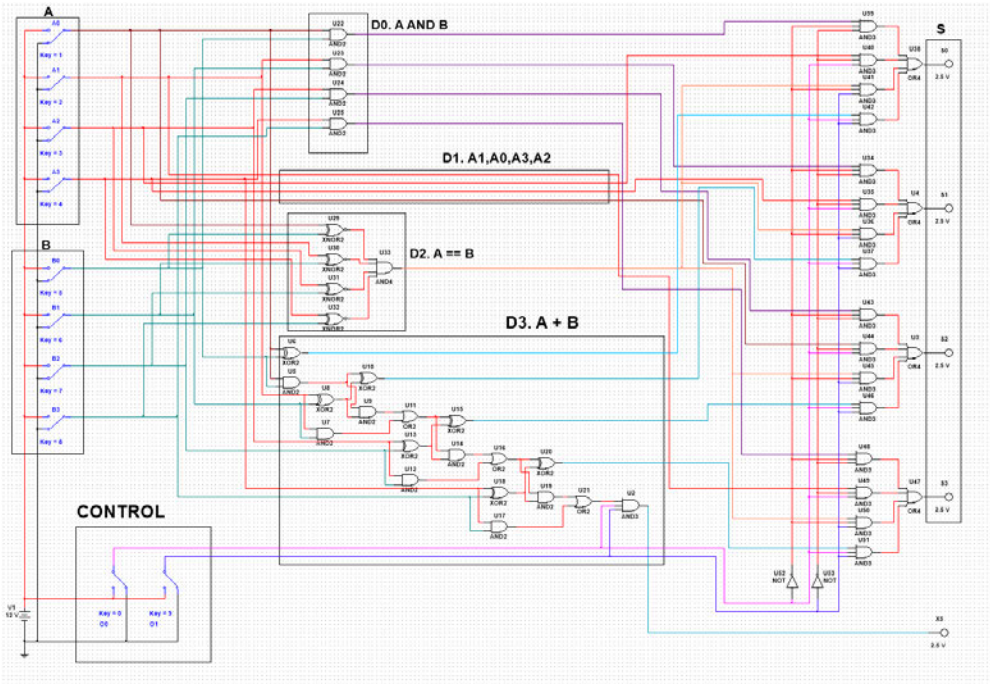


Figure 9 – ALU Test Case 1

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
00	A AND B	1111	1111	1111	0	1111	0

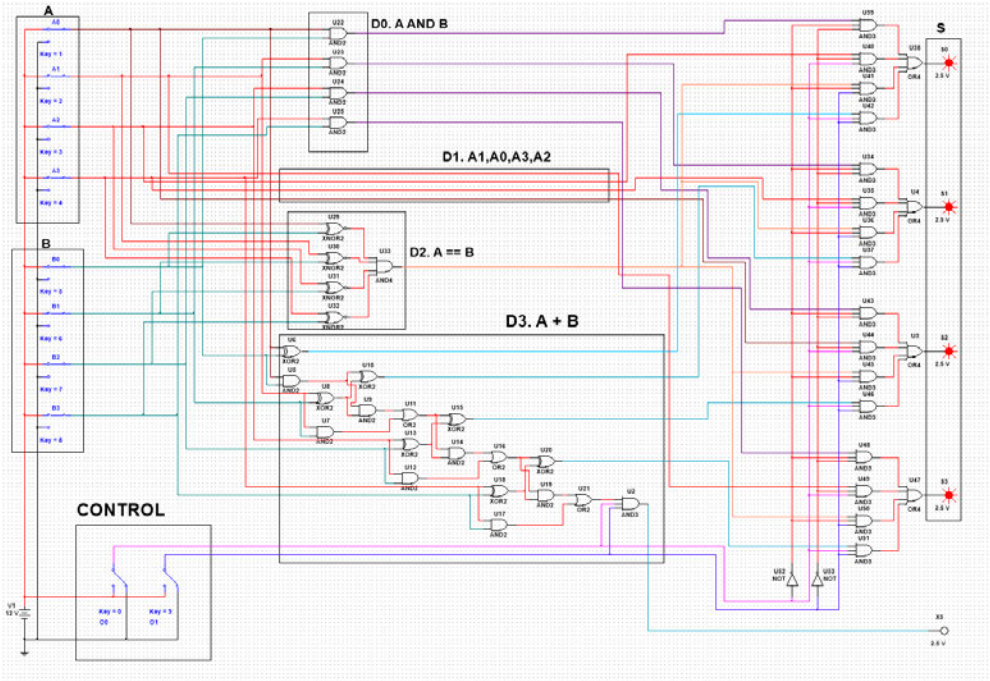


Figure 10 – ALU Test Case 2

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
00	A AND B	1010	1010	1010	0	1010	0

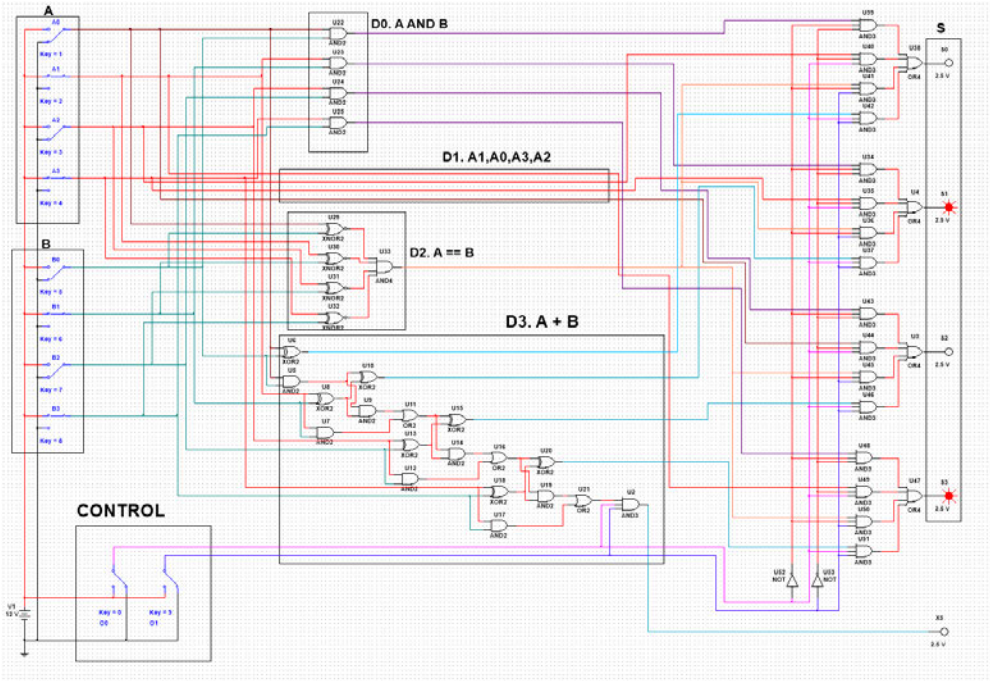


Figure 11 – ALU Test Case 3

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
00	A AND B	1111	0000	0000	0	0000	0

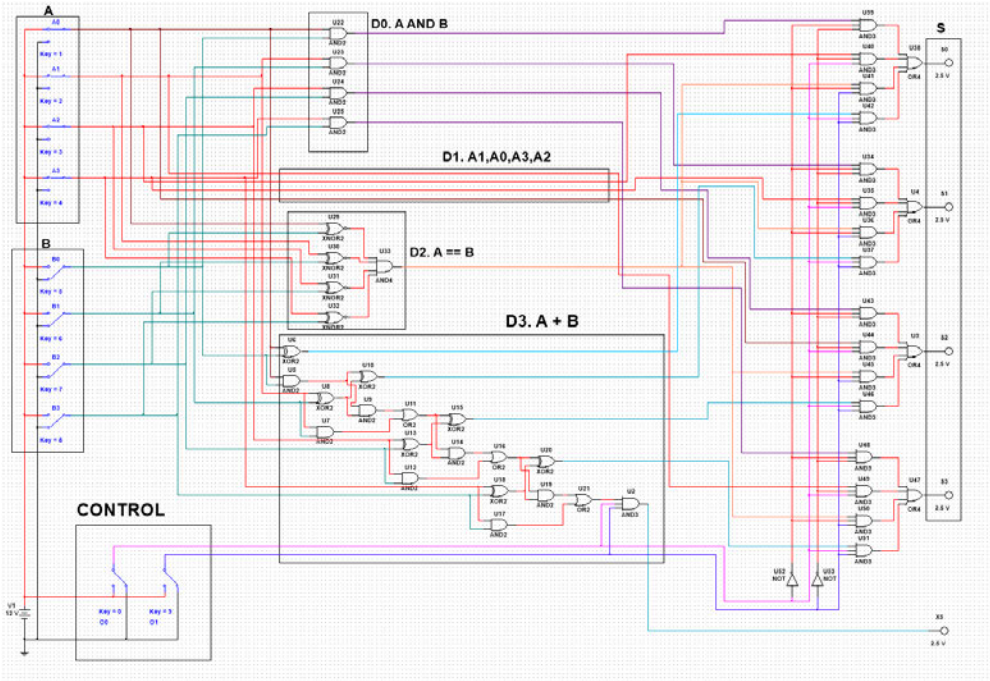


Figure 12 – ALU Test Case 4

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
00	A AND B	1111	0001	0001	0	0001	0

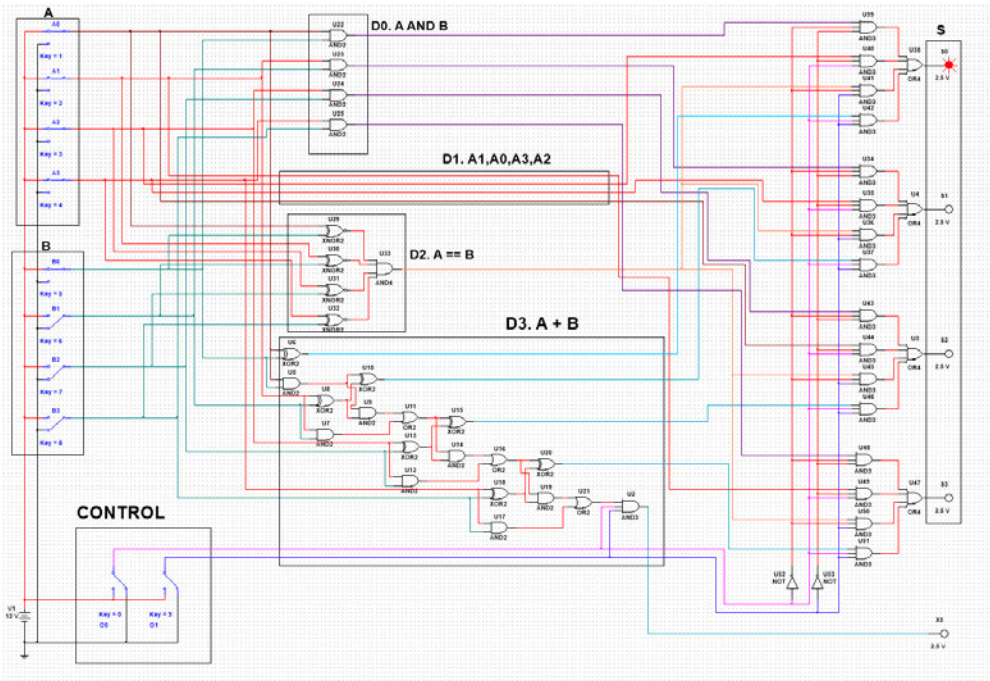


Figure 13 – ALU Test Case 5

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
00	A AND B	0101	0111	0101	0	0101	0

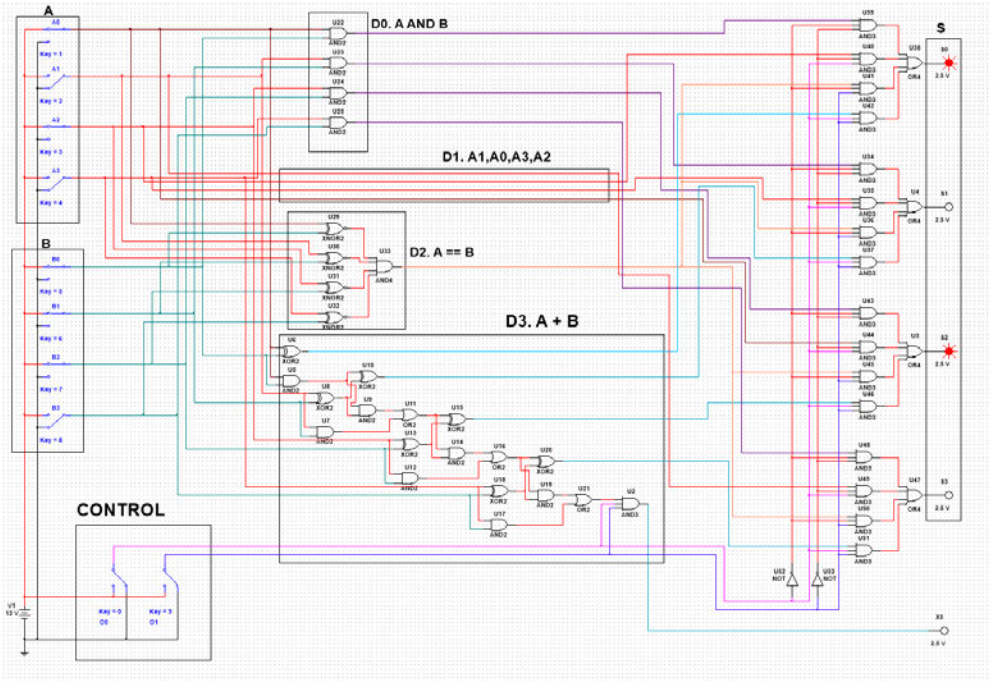


Figure 14 – ALU Test Case 6

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
00	A AND B	0011	0101	0001	0	0001	0

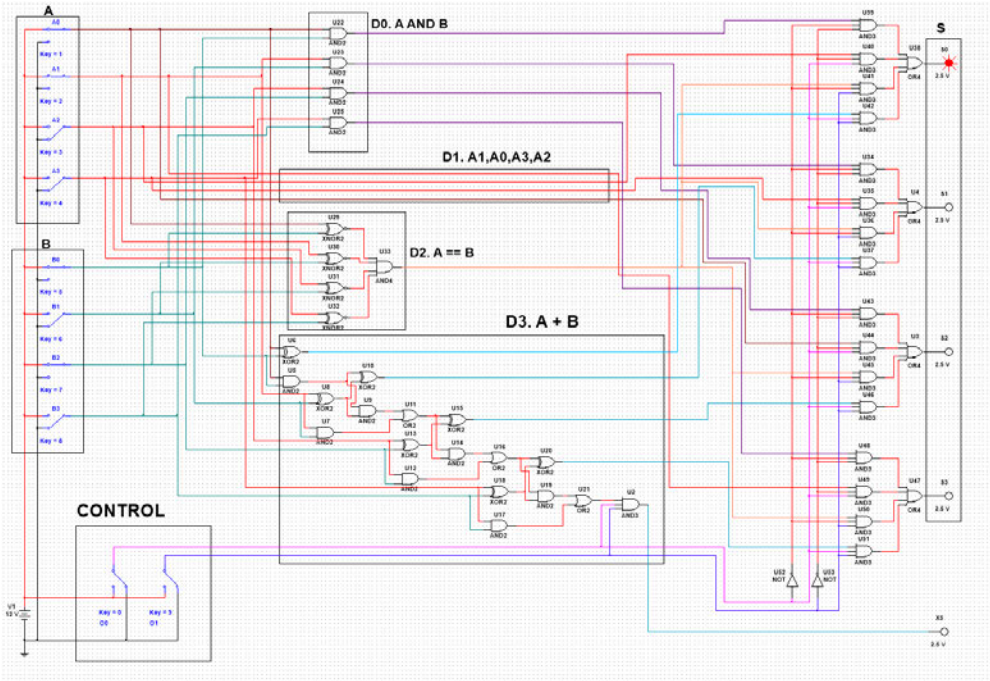


Figure 15 – ALU Test Case 7

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
00	A AND B	1001	1001	1001	0	1001	0

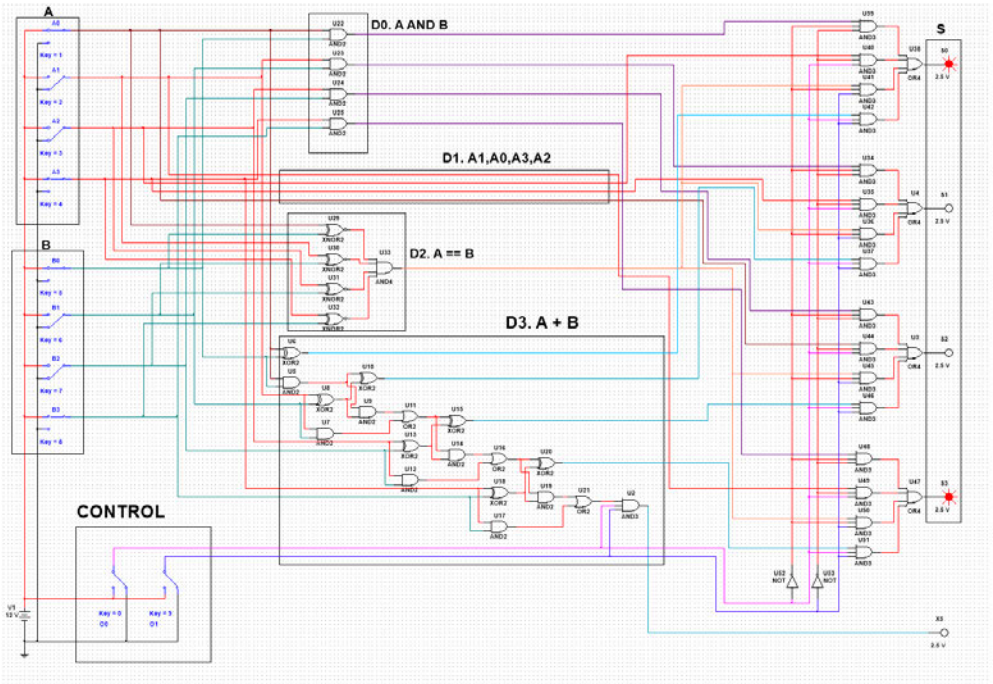


Figure 16 – ALU Test Case 8

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
00	A AND B	0111	1110	0110	0	0110	0

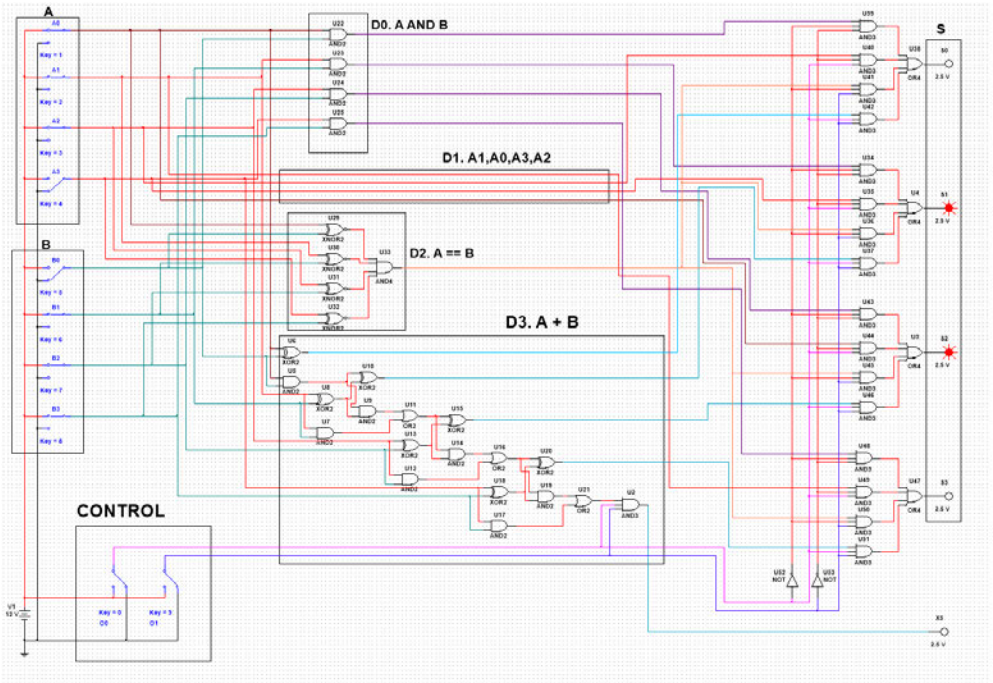


Figure 17 – ALU Test Case 9

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
00	A AND B	0011	1111	0011	0	0011	0

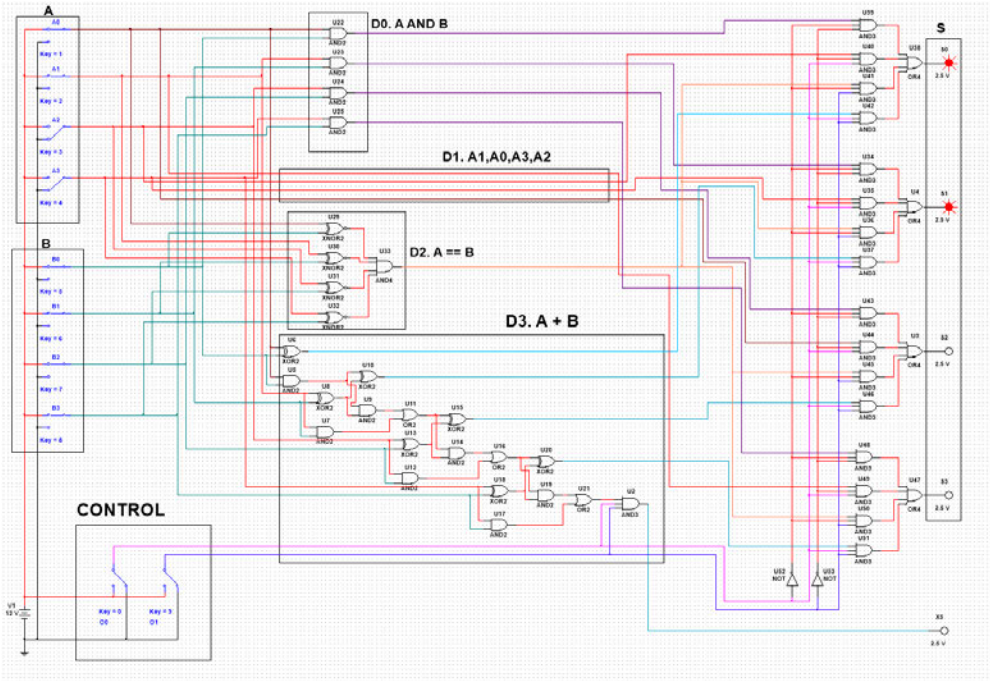


Figure 18 – ALU Test Case 10

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
01	A1,A0,A3,A2	0000	XXXX	0000	0	0000	0

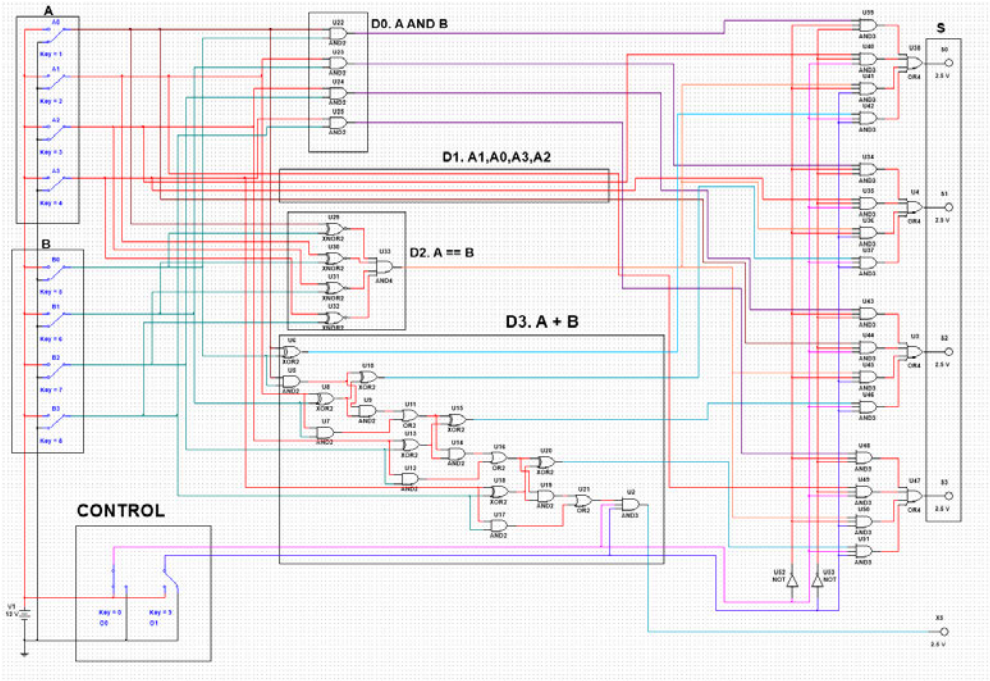


Figure 19 – ALU Test Case 11

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
01	A1,A0,A3,A2	1111	XXXX	1111	0	1111	0

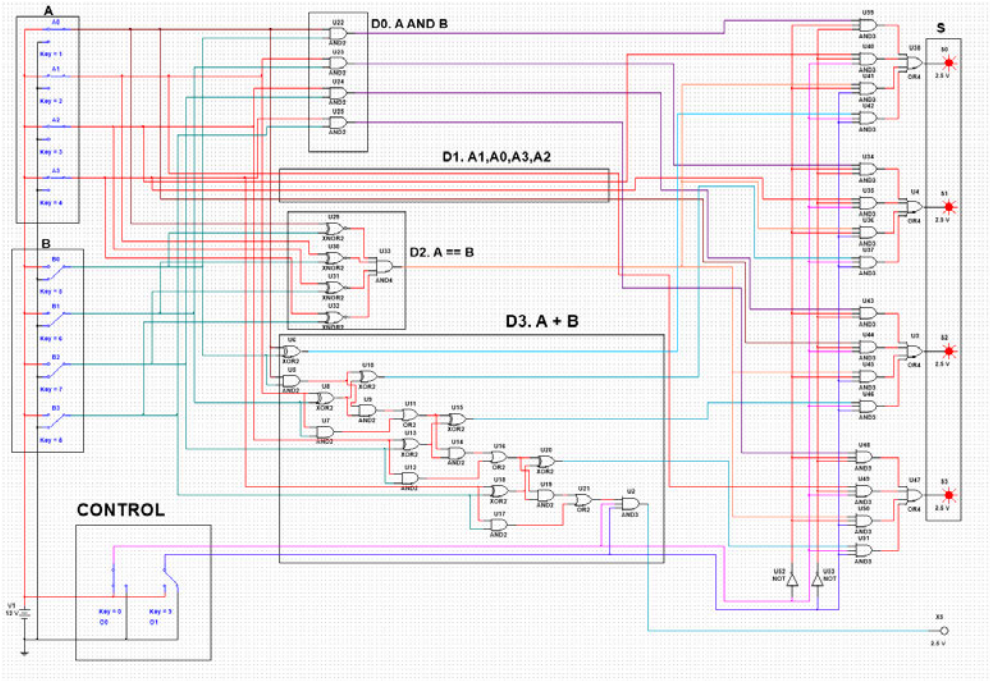


Figure 20 – ALU Test Case 12

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
01	A1,A0,A3,A2	0010	XXXX	1000	0	1000	0

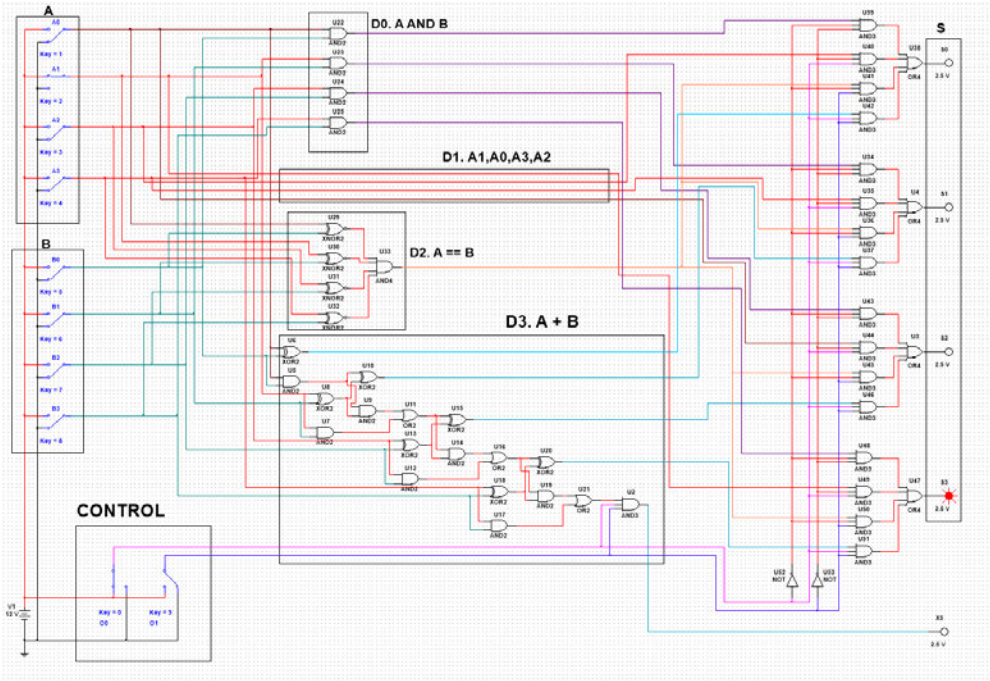


Figure 21 – ALU Test Case 13

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
01	A1,A0,A3,A2	0001	XXXX	0100	0	0100	0

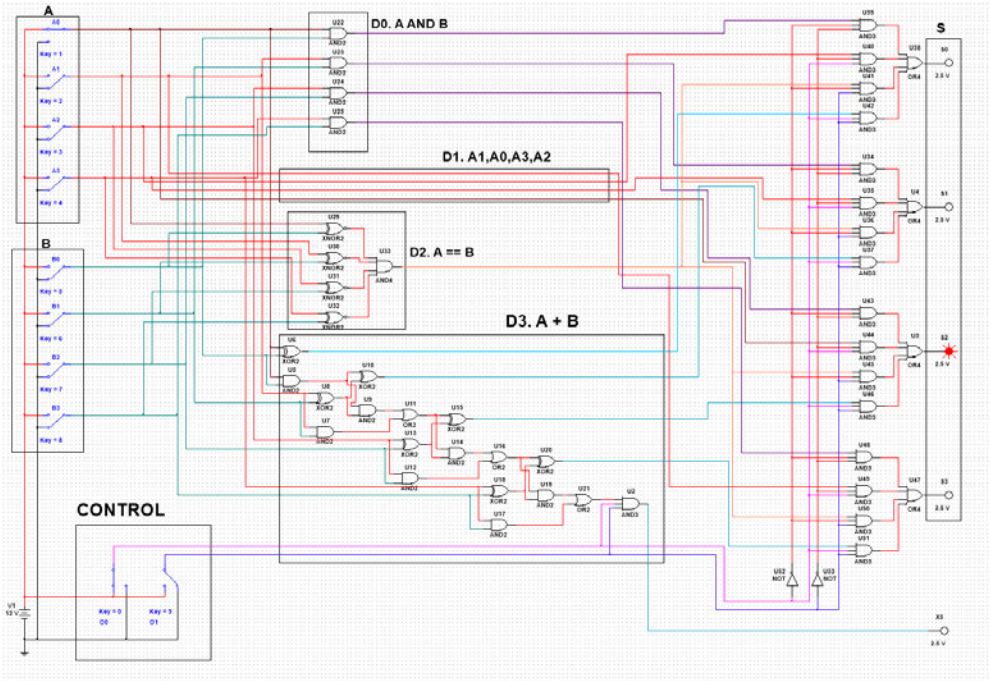


Figure 22 – ALU Test Case 14

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
01	A1,A0,A3,A2	1000	XXXX	0010	0	0010	0

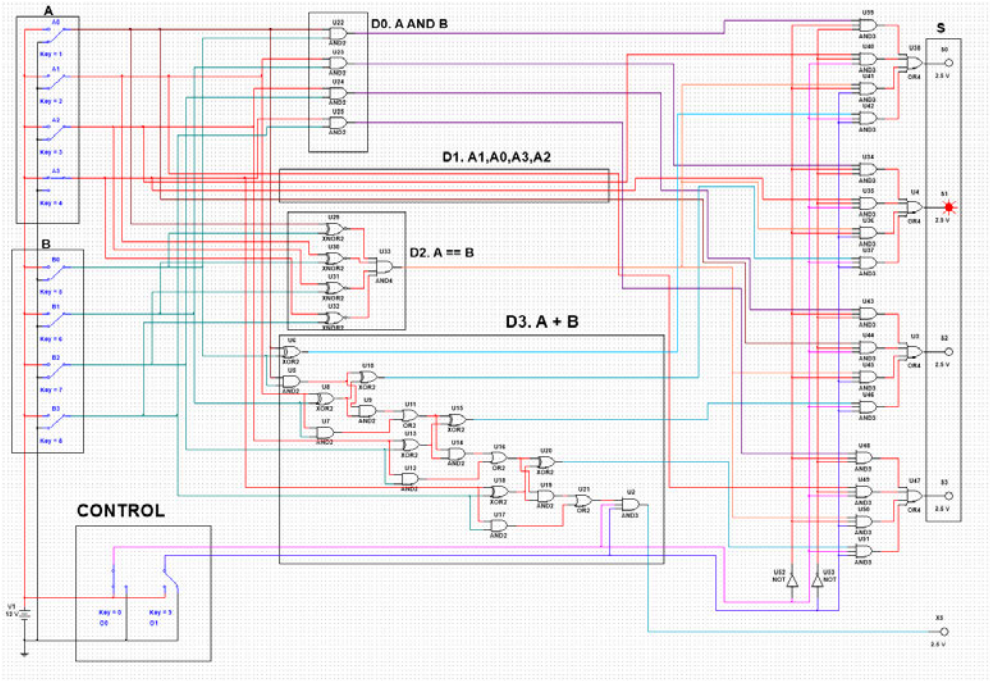


Figure 23 – ALU Test Case 15

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
01	A1,A0,A3,A2	0100	XXXX	0001	0	0001	0

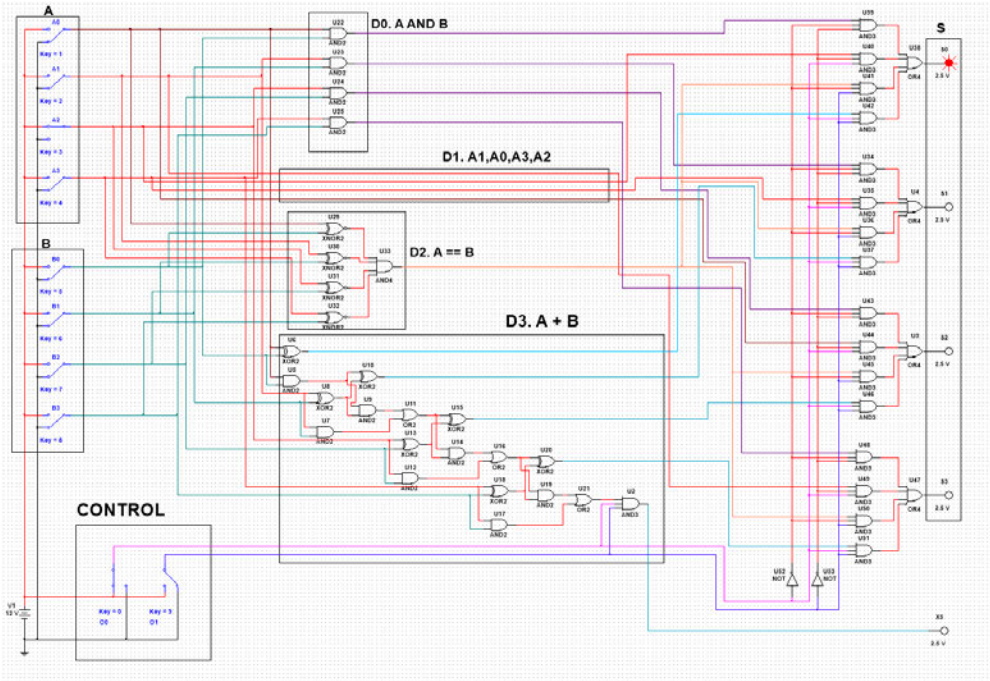


Figure 24 – ALU Test Case 16

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
10	A == B	0000	0000	1111	0	1111	0

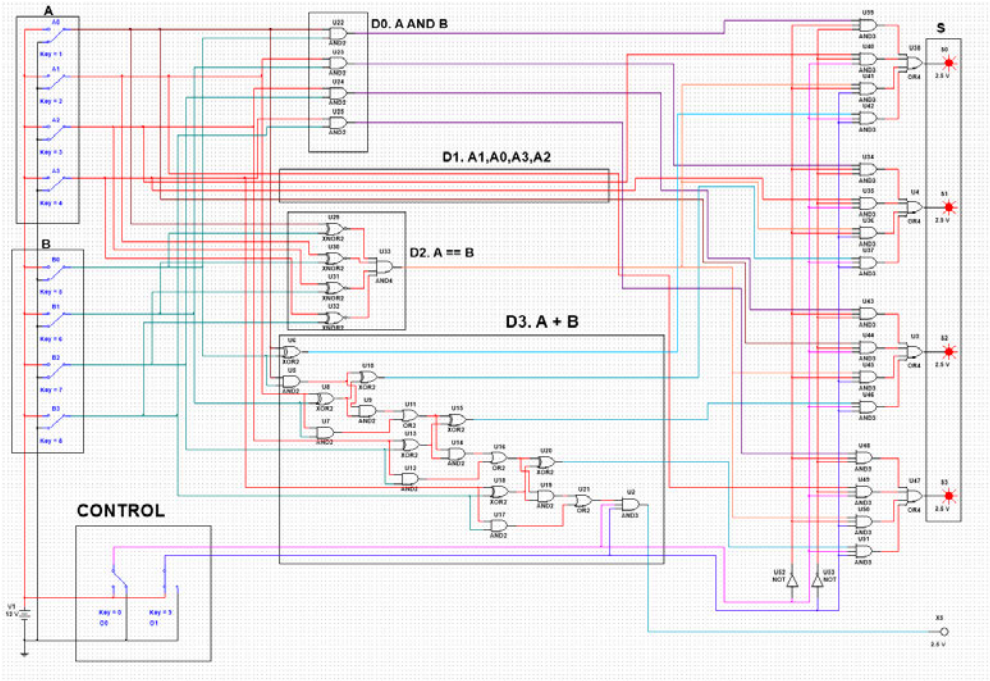


Figure 25 – ALU Test Case 17

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
10	A == B	1111	1111	1111	0	1111	0

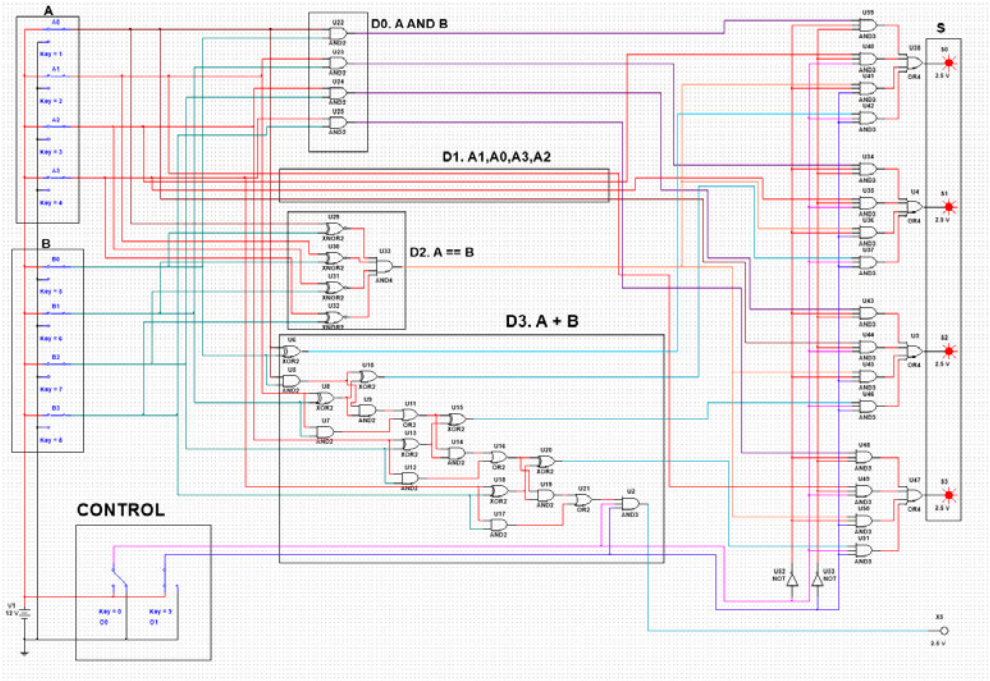


Figure 26 – ALU Test Case 18

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
10	A == B	1010	1010	1111	0	1111	0

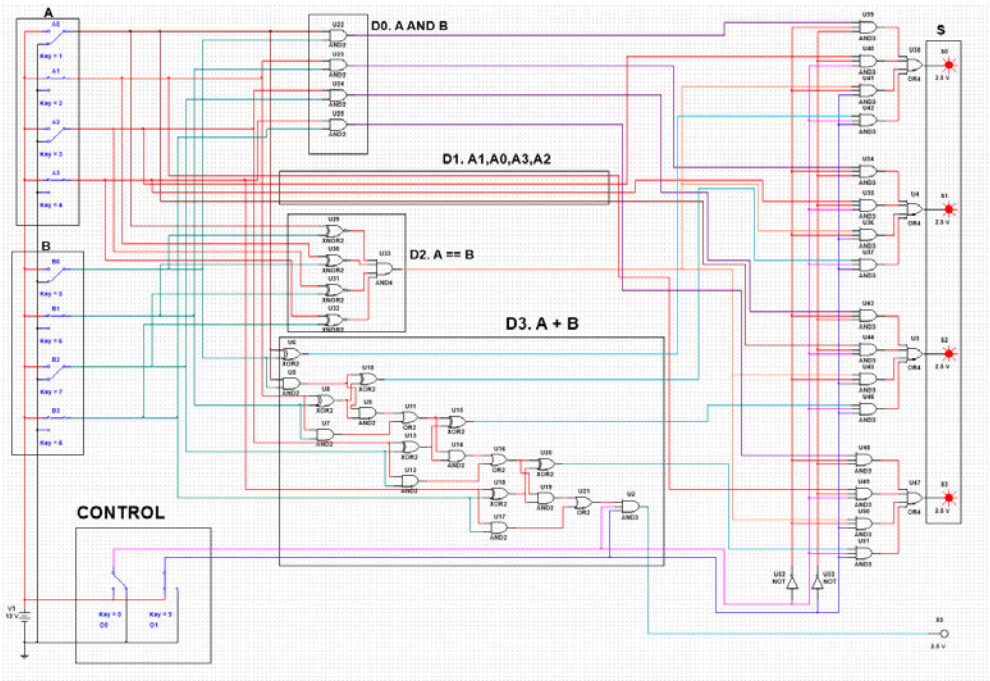


Figure 27 – ALU Test Case 19

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
10	A == B	0101	0101	1111	0	1111	0

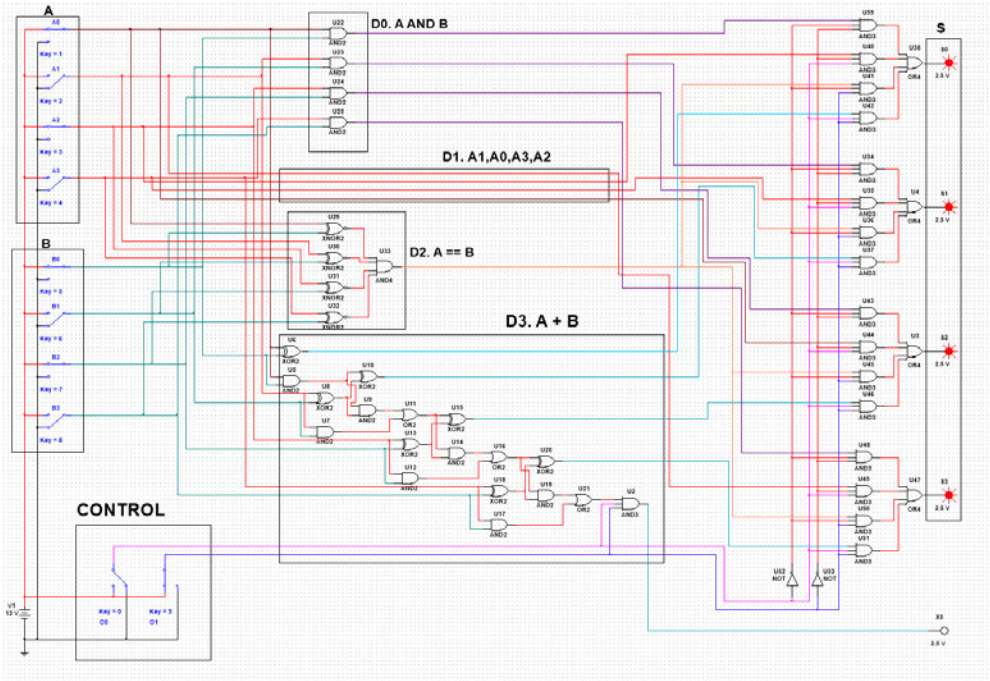


Figure 28 – ALU Test Case 20

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
10	A == B	1100	1100	1111	0	1111	0

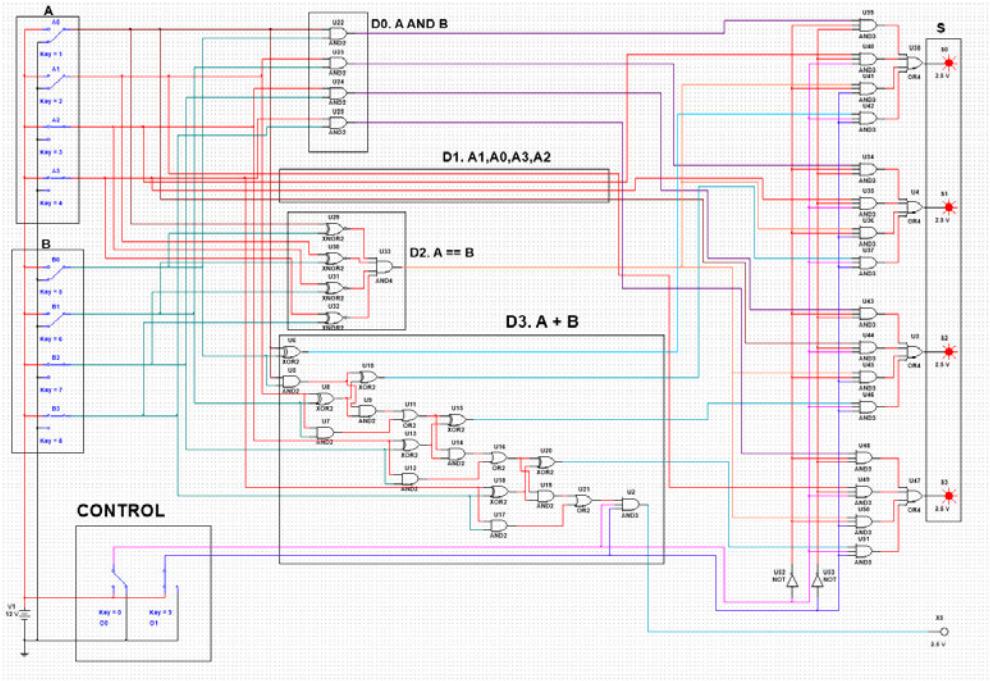


Figure 29 – ALU Test Case 21

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
10	A == B	0011	0011	1111	0	1111	0

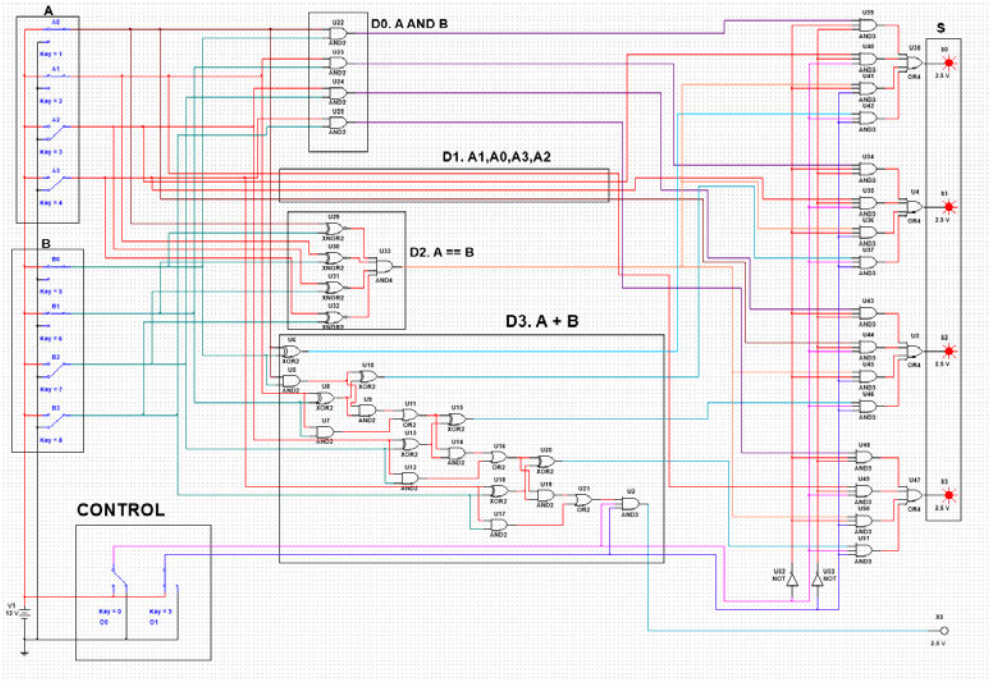


Figure 30 – ALU Test Case 22

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
10	A == B	0000	0001	0000	0	0000	0

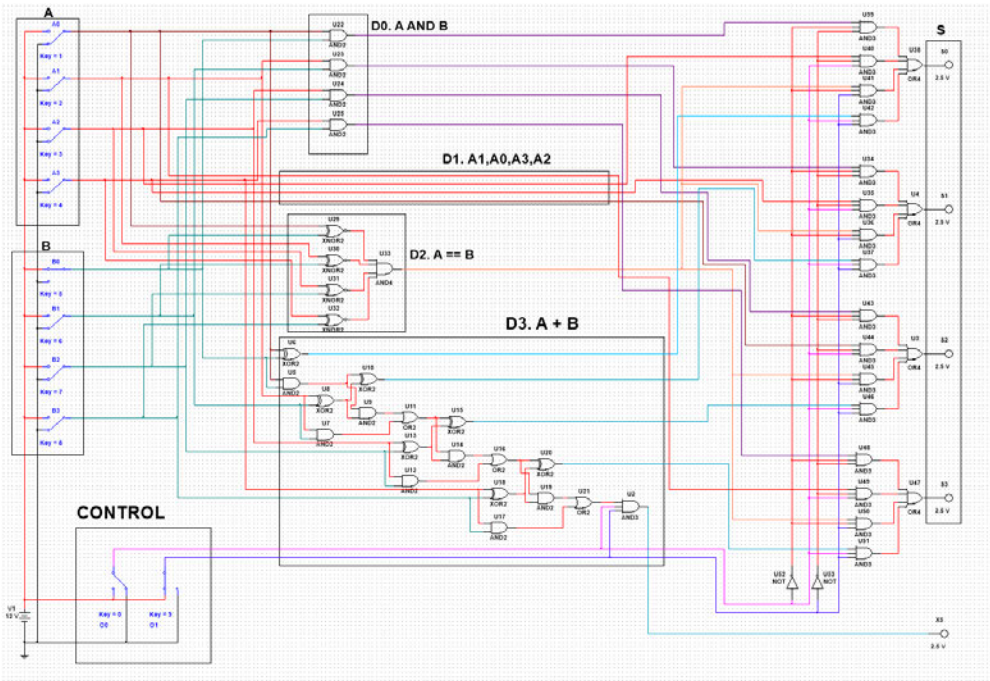


Figure 31 – ALU Test Case 23

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
10	A == B	1000	0000	0000	0	0000	0

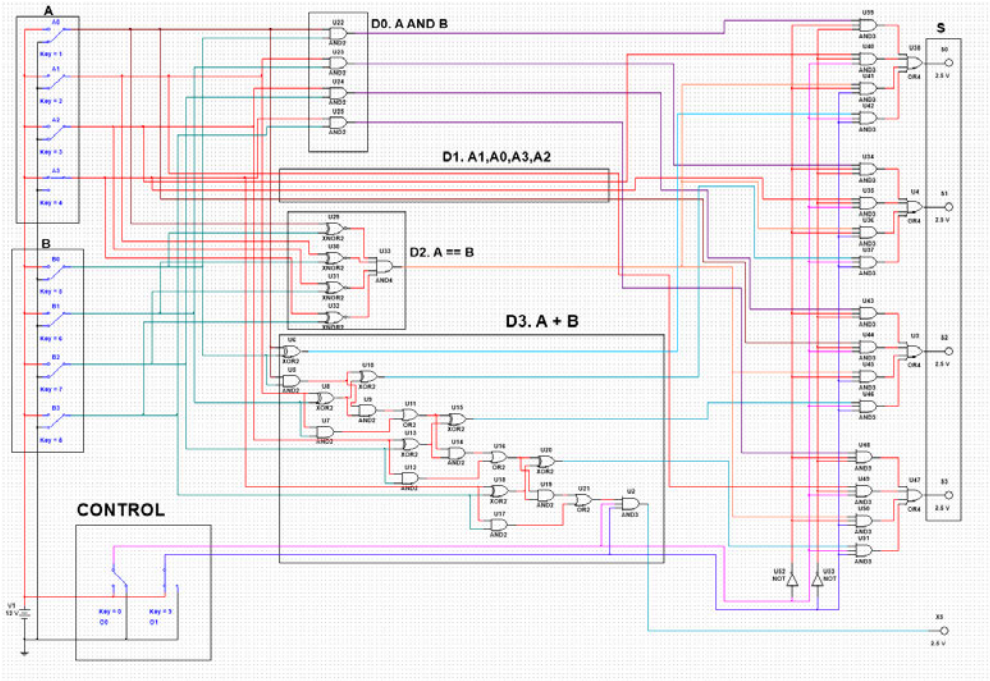


Figure 32 – ALU Test Case 24

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
10	A == B	1111	1110	0000	0	0000	0

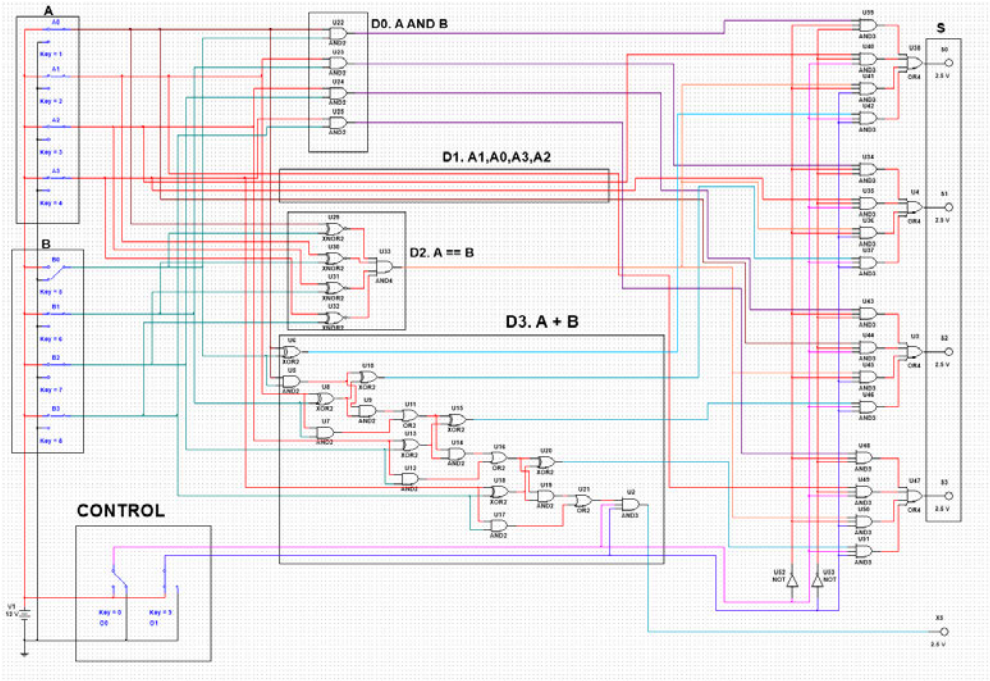


Figure 33 – ALU Test Case 25

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
10	A == B	1001	1001	1111	0	1111	0

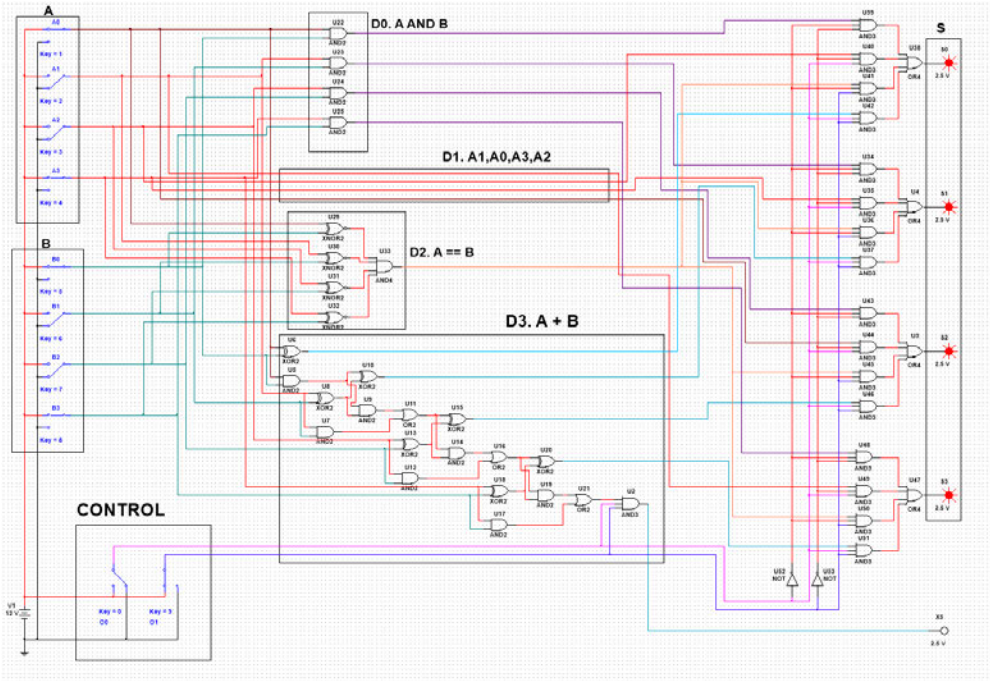


Figure 34 – ALU Test Case 26

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
11	A + B	0000	0000	0000	0	0000	0

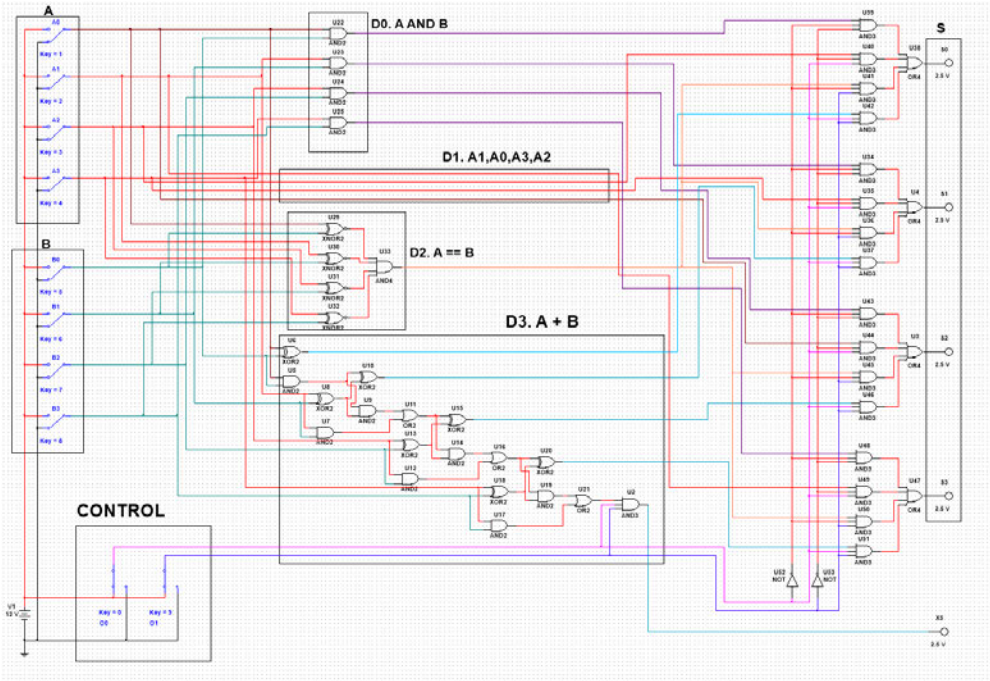


Figure 35 – ALU Test Case 27

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
11	A + B	0000	0001	0001	0	0001	0

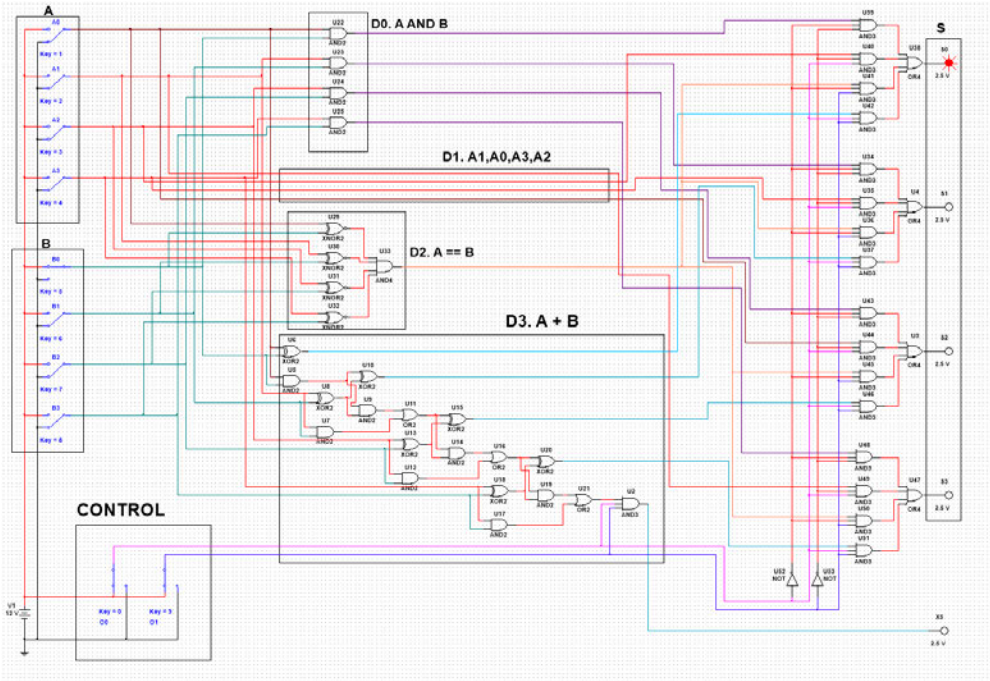


Figure 36 – ALU Test Case 28

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
11	A + B	0001	0001	0010	0	0010	0

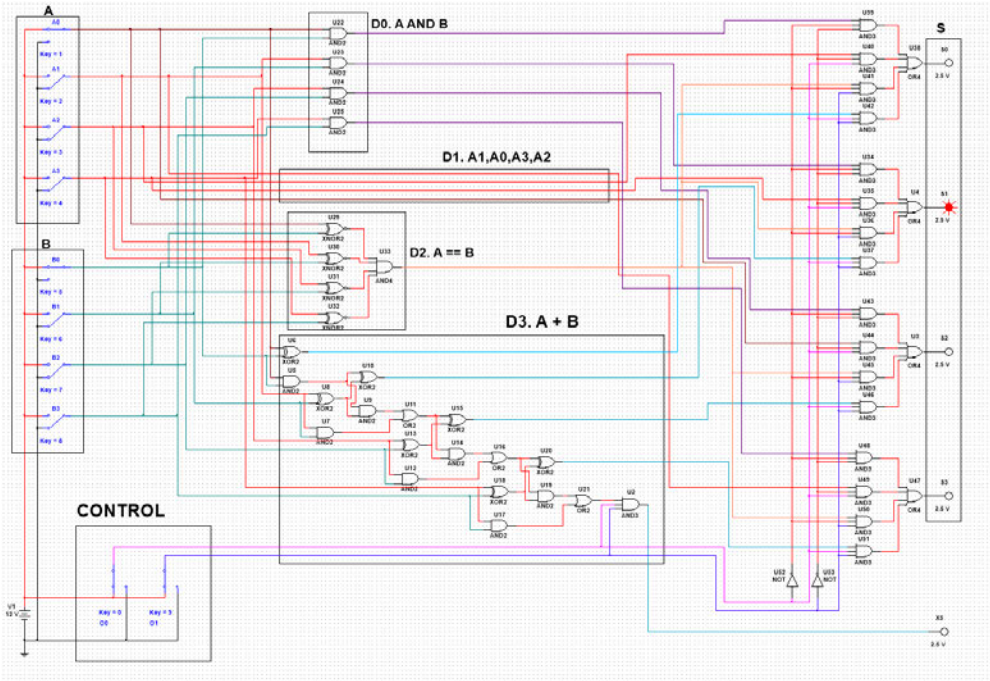


Figure 37 – ALU Test Case 29

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
11	A + B	1000	1000	0000	1	0000	1

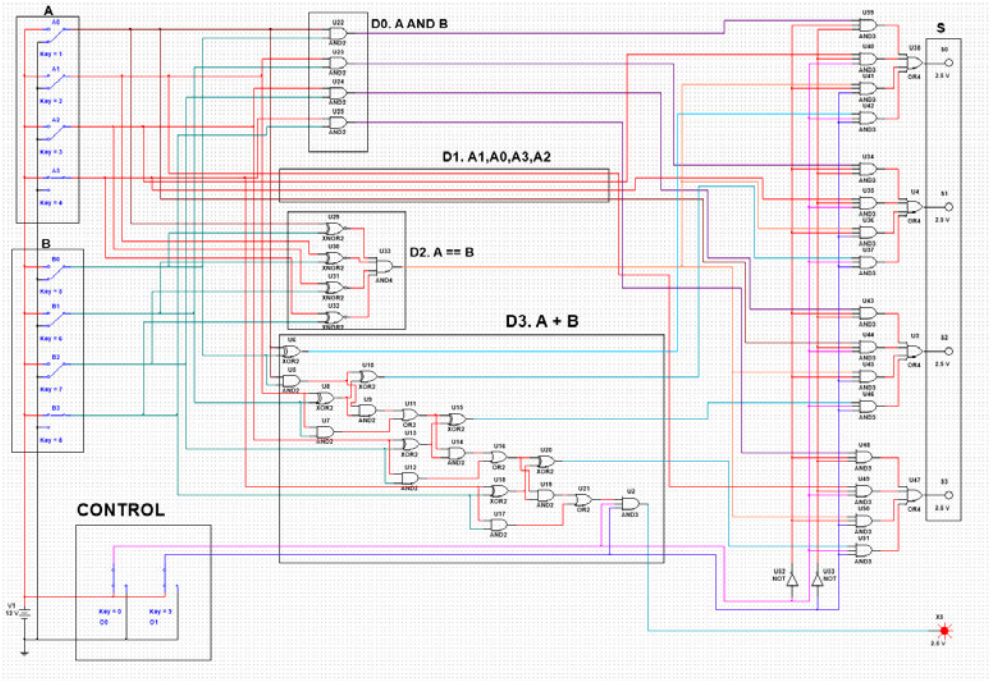


Figure 38 – ALU Test Case 30

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
11	A + B	1111	1111	1110	1	1110	1

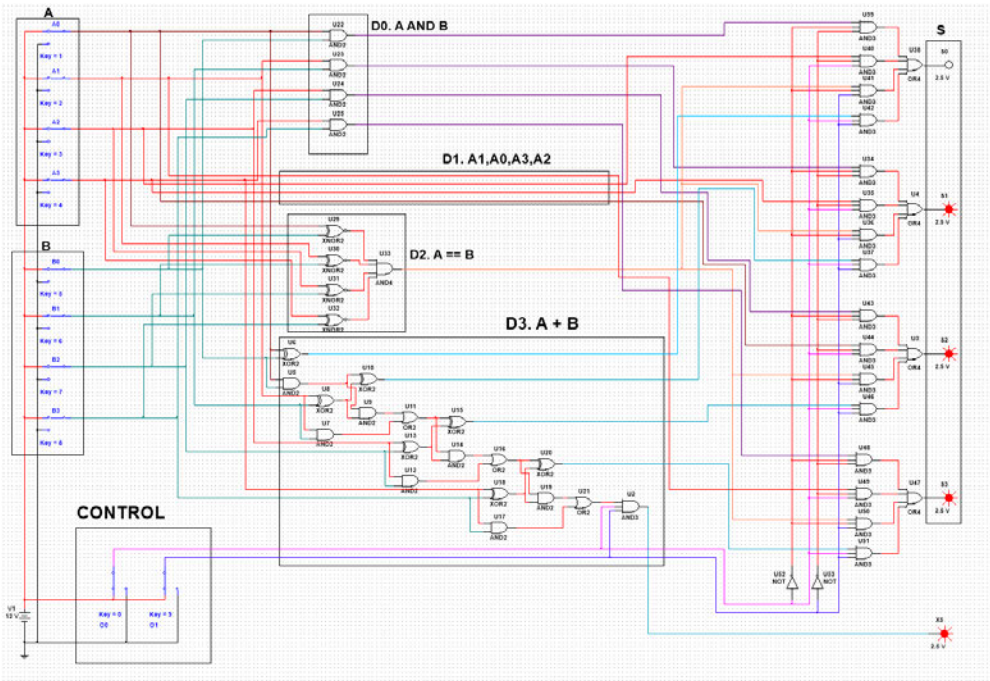


Figure 39 – ALU Test Case 31

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
11	A + B	0111	0111	1110	0	1110	0

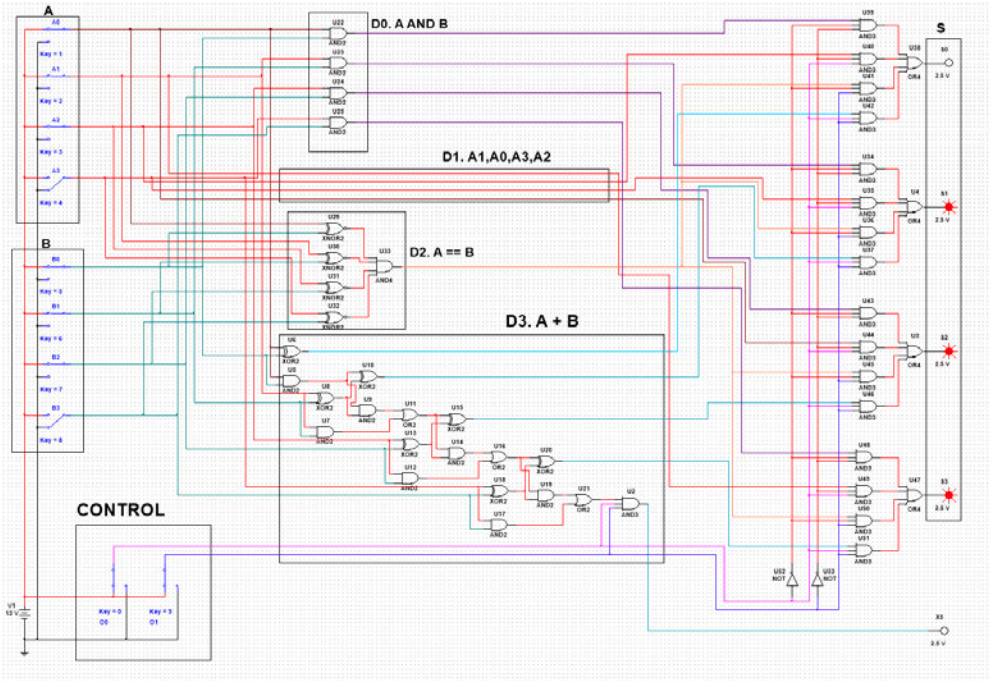


Figure 40 – ALU Test Case 32

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
11	A + B	0100	0011	0111	0	0111	0

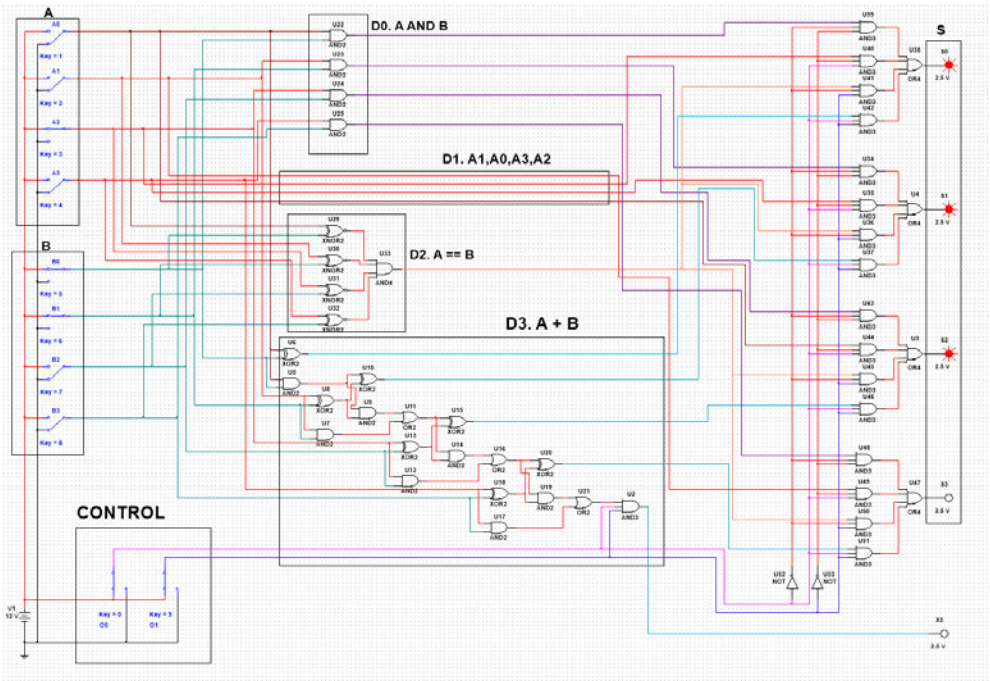


Figure 41 – ALU Test Case 33

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
11	A + B	1110	0001	1111	0	1111	0

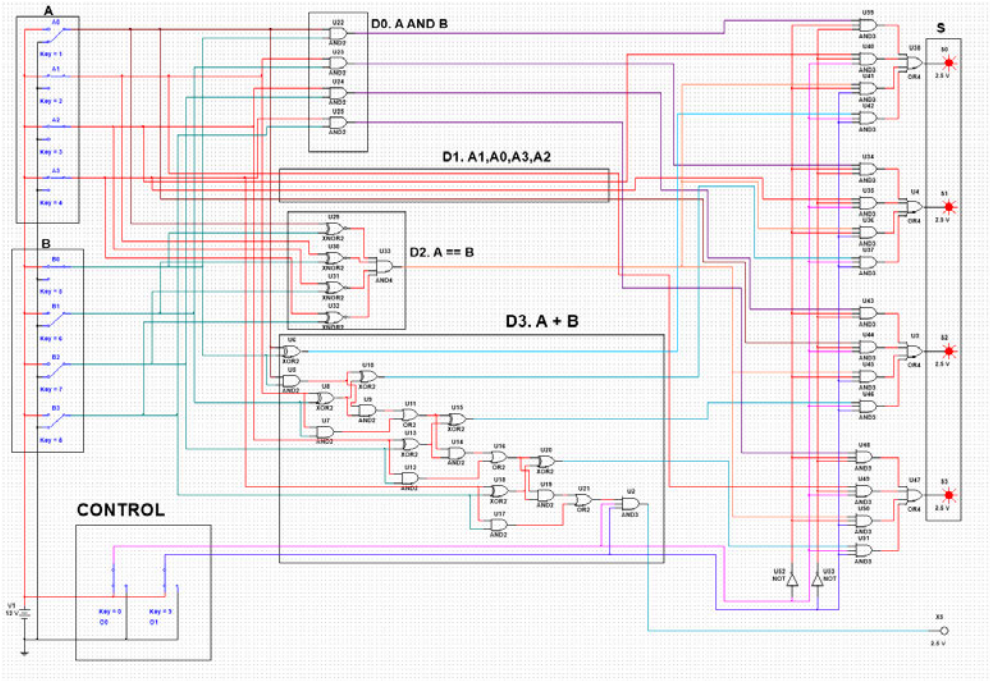


Figure 42 – ALU Test Case 34

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
11	A + B	1100	0100	0000	1	0000	1

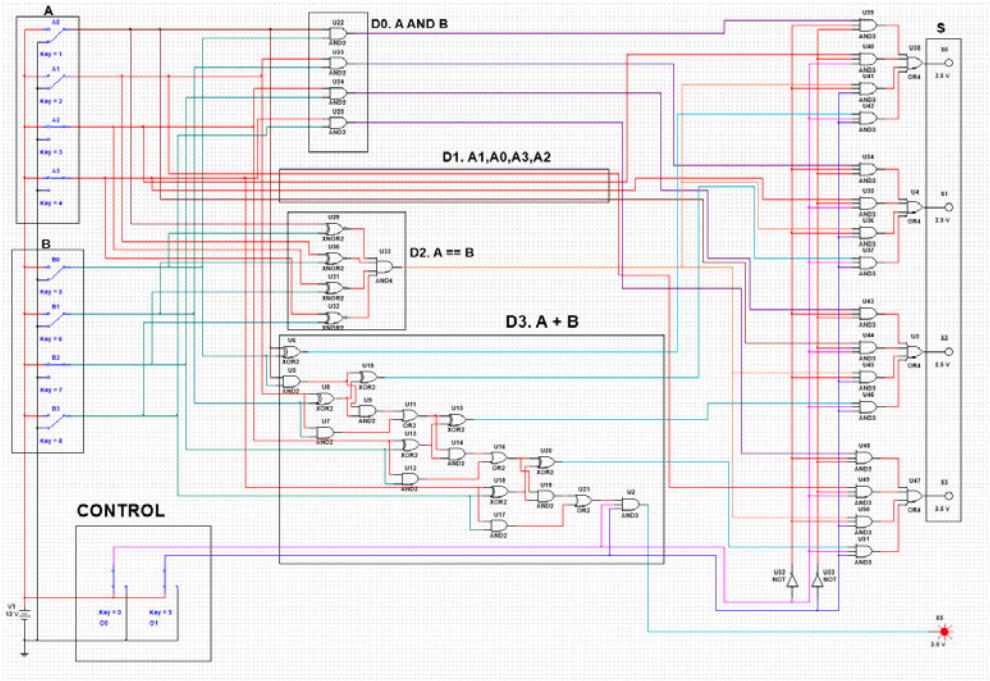


Figure 43 – ALU Test Case 35

OP1:0	Operation	A3:0	B3:0	Expected S3:0	Expected OF	Actual S3:0	Actual OF
11	A + B	1001	1000	0001	1	0001	1

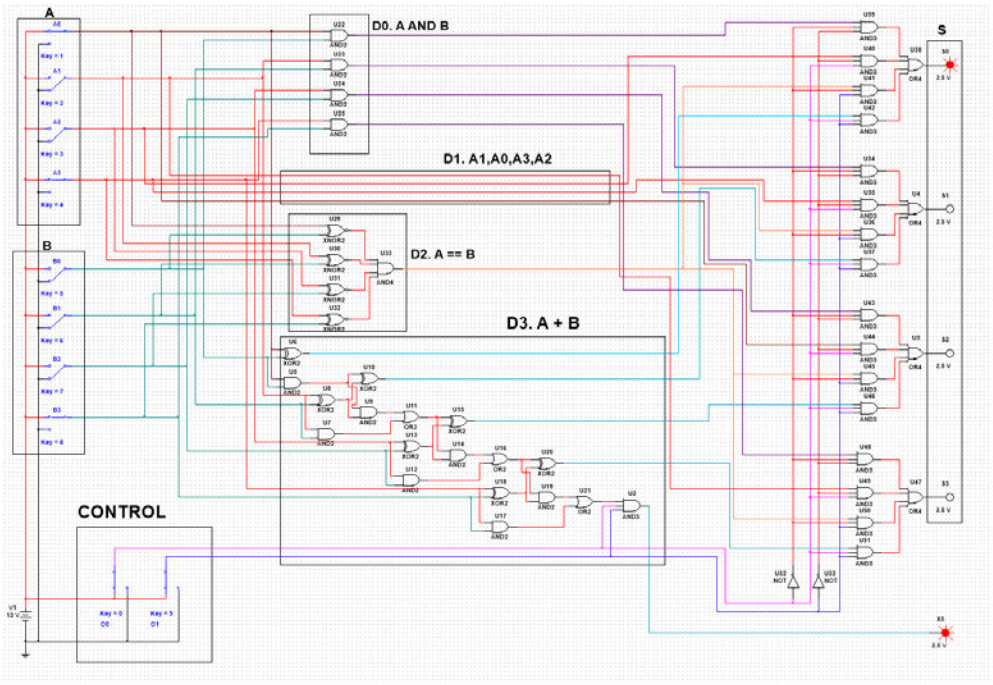


Figure 44 – ALU Test Case 36

The comprehensive testing methodology employed provides strong evidence of the ALU's correct and reliable operation across a wide range of inputs and scenarios. With the consistent results observed across multiple trials, the testing procedures have successfully validated the accuracy and robustness of the designed ALU implementation.

Conclusion

The 4-bit arithmetic logic unit (ALU) design demonstrates the fundamental principles of digital circuit engineering, successfully implementing four distinct operations using basic logic gates. By leveraging a combination of AND, OR, NOT, XOR, and XNOR gates, the ALU provides a versatile computational unit capable of bitwise AND, input shuffling, equality comparison, and binary addition.

The design process highlighted critical aspects of digital circuit development, including careful input selection, precise multiplexer implementation, and robust overflow detection. The systematic testing methodology, which included edge cases and multiple input scenarios, validated the ALU's consistent and reliable performance across different operations.

This project not only showcases the practical application of Boolean algebra and digital logic principles but also demonstrates the complexity and precision required in designing even a simple computational unit. The modular approach and careful gate-level implementation underscore the foundational role of fundamental logic gates in modern digital electronics.

Appendix

Appendix A – Contribution Sheet

UNIVERSITY OF WOLLONGONG IN DUBAI

Contributions Sheet

This sheet must be attached (electronically scanned) to your group report submission.

There have been a number of instances where particular group members have not engaged in the project to an adequate level (sometimes making no contribution).

To take account of this, each group is required to list the percentage contribution of each group member (totalling 100%).

Each group member must sign this sheet in acknowledgment that they accept the contributions listed against them.

In the case where there are differences in the level of contribution by group members, the final individual group mark will be adjusted up or down based on the procedure explained on page 2 of this document.

Group number/name:

Adjusting of Final Group Project Mark Based on Individual Participation

Due to a number of cases where students have not been engaging with their groups (reduced capacity or not at all), the following method will be used to determine an individual's group project mark.

1. The Group report will be marked based on the marking criteria for a mark out of 100.
2. This will contribute to the first 50% of the individual mark.
3. The remaining 50% will be based on the percentage contribution of each group member.

Note 1: If any student receives an adjusted mark of over 100%, the mark will be capped at 100%.
Note 2: If all group members participate equally, they all receive the same final mark, equal to their group report mark.

ECTE233 Project Contributions Sheet

Appendix B – Full Truth Tables

OP 00 – A AND B											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	1	0	1	0	0	0	0
0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	1	1	1	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	1	0	0	0	0
0	0	0	0	1	0	1	0	0	0	0	0
0	0	0	0	1	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	1	1	0	1	0	0	0	0
0	0	0	0	1	1	1	0	0	0	0	0
0	0	0	0	1	1	1	1	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	0	0
0	0	0	1	0	0	1	1	0	0	0	1
0	0	0	1	0	1	0	0	0	0	0	0
0	0	0	1	0	1	0	1	0	0	0	1
0	0	0	1	0	1	1	0	0	0	0	0
0	0	0	1	0	1	1	1	0	0	0	1
0	0	0	1	1	0	0	0	0	0	0	0
0	0	0	1	1	0	0	1	0	0	0	1
0	0	0	1	1	0	1	0	0	0	0	0
0	0	0	1	1	0	1	1	0	0	0	1
0	0	0	1	1	1	0	0	0	0	0	0
0	0	0	1	1	1	0	1	0	0	0	1
0	0	0	1	1	1	1	0	0	0	0	0
0	0	0	1	1	1	1	1	0	0	0	1
0	0	1	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	1	0	0	0	0
0	0	1	0	0	0	1	0	0	0	1	0
0	0	1	0	0	0	1	1	0	0	1	0
0	0	1	0	0	1	0	0	0	0	0	0
0	0	1	0	0	1	0	1	0	0	0	0
0	0	1	0	0	1	1	0	0	0	1	0
0	0	1	0	0	1	1	1	0	0	1	0
0	0	1	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	1	0	0	0	0

OP 00 – A AND B											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
0	0	1	0	1	0	1	0	0	0	1	0
0	0	1	0	1	0	1	1	0	0	1	0
0	0	1	0	1	1	0	0	0	0	0	0
0	0	1	0	1	1	0	1	0	0	0	0
0	0	1	0	1	1	1	0	0	0	1	0
0	0	1	0	1	1	1	1	0	0	1	0
0	0	1	1	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	1	0	0	0	1
0	0	1	1	0	0	1	0	0	0	1	0
0	0	1	1	0	0	1	1	0	0	1	1
0	0	1	1	0	1	0	0	0	0	0	0
0	0	1	1	0	1	0	1	0	0	0	1
0	0	1	1	0	1	1	0	0	0	1	0
0	0	1	1	0	1	1	1	0	0	1	1
0	0	1	1	1	0	0	0	0	0	0	0
0	0	1	1	1	0	0	1	0	0	0	1
0	0	1	1	1	0	1	0	0	0	1	0
0	0	1	1	1	0	1	1	0	0	1	1
0	0	1	1	1	1	0	0	0	0	0	0
0	0	1	1	1	1	0	1	0	0	0	1
0	0	1	1	1	1	1	0	0	0	1	0
0	0	1	1	1	1	1	1	0	0	1	1
0	1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	1	0	0	0	0
0	1	0	0	0	0	1	0	0	0	0	0
0	1	0	0	0	0	1	1	0	0	0	0
0	1	0	0	0	1	0	0	0	1	0	0
0	1	0	0	0	1	0	1	0	1	0	0
0	1	0	0	0	1	1	0	0	1	0	0
0	1	0	0	0	1	1	1	0	1	0	0
0	1	0	0	1	0	0	0	0	0	0	0
0	1	0	0	1	0	0	1	0	0	0	0
0	1	0	0	1	0	1	0	0	0	0	0
0	1	0	0	1	0	1	1	0	0	0	0
0	1	0	0	1	1	0	0	0	1	0	0
0	1	0	0	1	1	0	1	0	1	0	0
0	1	0	0	1	1	1	0	0	1	0	0
0	1	0	0	1	1	1	1	0	1	0	0
0	1	0	1	0	0	0	0	0	0	0	0
0	1	0	1	0	0	0	1	0	0	0	1
0	1	0	1	0	0	1	0	0	0	0	0
0	1	0	1	0	0	1	1	0	0	0	1
0	1	0	1	0	1	0	0	0	1	0	0

OP 00 – A AND B											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
0	1	0	1	0	1	0	1	0	1	0	1
0	1	0	1	0	1	1	0	0	1	0	0
0	1	0	1	0	1	1	1	0	1	0	1
0	1	0	1	1	0	0	0	0	0	0	0
0	1	0	1	1	0	0	1	0	0	0	1
0	1	0	1	1	0	1	0	0	0	0	0
0	1	0	1	1	0	1	1	0	0	0	1
0	1	0	1	1	1	0	0	0	1	0	0
0	1	0	1	1	1	0	1	0	1	0	1
0	1	0	1	1	1	1	0	0	1	0	0
0	1	0	1	1	1	1	1	0	1	0	1
0	1	1	0	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	1	0	0	0	0
0	1	1	0	0	0	1	0	0	0	1	0
0	1	1	0	0	0	1	1	0	0	1	0
0	1	1	0	0	1	0	0	0	1	0	0
0	1	1	0	0	1	0	1	0	1	0	0
0	1	1	0	0	1	1	0	0	1	1	0
0	1	1	0	0	1	1	1	0	1	1	0
0	1	1	0	1	0	0	0	0	0	0	0
0	1	1	0	1	0	0	1	0	0	0	0
0	1	1	0	1	0	1	0	0	0	1	0
0	1	1	0	1	0	1	1	0	0	1	0
0	1	1	0	1	1	0	0	0	1	0	0
0	1	1	0	1	1	0	1	0	1	0	0
0	1	1	0	1	1	1	0	0	1	1	0
0	1	1	0	1	1	1	1	0	1	1	0
0	1	1	1	0	0	0	0	0	0	0	0
0	1	1	1	0	0	0	1	0	0	0	1
0	1	1	1	0	0	1	0	0	0	1	0
0	1	1	1	0	0	1	1	0	0	1	1
0	1	1	1	0	1	0	0	0	1	0	0
0	1	1	1	0	1	0	1	0	1	0	1
0	1	1	1	0	1	1	0	0	1	1	0
0	1	1	1	0	1	1	1	0	1	1	1
0	1	1	1	1	0	0	0	0	0	0	0
0	1	1	1	1	0	0	1	0	0	0	1
0	1	1	1	1	0	1	0	0	0	1	0
0	1	1	1	1	0	1	1	0	0	1	1
0	1	1	1	1	1	0	0	0	1	0	0
0	1	1	1	1	1	0	1	0	1	0	1
0	1	1	1	1	1	1	0	0	1	1	0
0	1	1	1	1	1	1	1	0	1	1	1

OP 00 – A AND B											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
1	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0	0
1	0	0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	0	1	1	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0	0
1	0	0	0	0	1	0	1	0	0	0	0
1	0	0	0	0	1	1	0	0	0	0	0
1	0	0	0	0	1	1	1	0	0	0	0
1	0	0	0	1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	1	1	0	0	0
1	0	0	0	1	0	1	0	1	0	0	0
1	0	0	0	1	0	1	1	1	0	0	0
1	0	0	0	1	1	0	0	1	0	0	0
1	0	0	0	1	1	0	1	1	0	0	0
1	0	0	0	1	1	1	0	1	0	0	0
1	0	0	0	1	1	1	1	1	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	1	0	0	0	1
1	0	0	1	0	0	1	0	0	0	0	0
1	0	0	1	0	0	1	1	0	0	0	1
1	0	0	1	0	1	0	0	0	0	0	0
1	0	0	1	0	1	0	1	0	0	0	1
1	0	0	1	0	1	1	0	0	0	0	0
1	0	0	1	0	1	1	1	0	0	0	1
1	0	0	1	1	0	0	0	1	0	0	0
1	0	0	1	1	0	0	1	1	0	0	1
1	0	0	1	1	0	1	0	1	0	0	0
1	0	0	1	1	0	1	1	1	0	0	1
1	0	0	1	1	1	0	0	1	0	0	0
1	0	0	1	1	1	0	1	1	0	0	1
1	0	0	1	1	1	1	0	1	0	0	0
1	0	0	1	1	1	1	1	1	0	0	1
1	0	1	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	1	0	0	0	0
1	0	1	0	0	0	1	0	0	0	1	0
1	0	1	0	0	0	1	1	0	0	1	0
1	0	1	0	0	1	0	0	0	0	0	0
1	0	1	0	0	1	0	1	0	0	0	0
1	0	1	0	0	1	1	0	0	0	1	0
1	0	1	0	0	1	1	1	0	0	1	0
1	0	1	0	1	0	0	0	1	0	0	0
1	0	1	0	1	0	0	1	1	0	0	0
1	0	1	0	1	0	1	0	1	0	1	0

OP 00 – A AND B											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
1	0	1	0	1	0	1	1	1	0	1	0
1	0	1	0	1	1	0	0	1	0	0	0
1	0	1	0	1	1	0	1	1	0	0	0
1	0	1	0	1	1	1	0	1	0	1	0
1	0	1	0	1	1	1	1	1	0	1	0
1	0	1	1	0	0	0	0	0	0	0	0
1	0	1	1	0	0	0	1	0	0	0	1
1	0	1	1	0	0	1	0	0	0	1	0
1	0	1	1	0	0	1	1	0	0	1	1
1	0	1	1	0	1	0	0	0	0	0	0
1	0	1	1	0	1	0	1	0	0	0	1
1	0	1	1	0	1	1	0	0	0	1	0
1	0	1	1	0	1	1	1	0	0	1	1
1	0	1	1	1	0	0	0	1	0	0	0
1	0	1	1	1	0	0	1	1	0	0	1
1	0	1	1	1	0	1	0	1	0	1	0
1	0	1	1	1	0	1	1	1	0	1	1
1	0	1	1	1	1	0	0	1	0	0	0
1	0	1	1	1	1	0	1	1	0	0	1
1	0	1	1	1	1	1	0	1	0	1	0
1	0	1	1	1	1	1	1	1	0	1	1
1	1	0	0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	1	0	0	0	0
1	1	0	0	0	0	1	0	0	0	0	0
1	1	0	0	0	0	1	1	0	0	0	0
1	1	0	0	0	1	0	0	0	1	0	0
1	1	0	0	0	1	0	1	0	1	0	0
1	1	0	0	0	1	1	0	0	1	0	0
1	1	0	0	0	1	1	1	0	1	0	0
1	1	0	0	1	0	0	0	1	0	0	0
1	1	0	0	1	0	0	1	1	0	0	0
1	1	0	0	1	0	1	0	1	0	0	0
1	1	0	0	1	0	1	1	1	0	0	0
1	1	0	0	1	1	0	0	1	1	0	0
1	1	0	0	1	1	0	1	1	1	0	0
1	1	0	0	1	1	1	0	1	1	0	0
1	1	0	0	1	1	1	1	1	1	0	0
1	1	0	1	0	0	0	0	0	0	0	0
1	1	0	1	0	0	0	1	0	0	0	1
1	1	0	1	0	0	1	0	0	0	0	0
1	1	0	1	0	0	1	1	0	0	0	1
1	1	0	1	0	1	0	0	0	1	0	0
1	1	0	1	0	1	0	1	0	1	0	1

OP 00 – A AND B											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
1	1	0	1	0	1	1	0	0	1	0	0
1	1	0	1	0	1	1	1	0	1	0	1
1	1	0	1	1	0	0	0	1	0	0	0
1	1	0	1	1	0	0	1	1	0	0	1
1	1	0	1	1	0	1	0	1	0	0	0
1	1	0	1	1	0	1	1	1	0	0	1
1	1	0	1	1	1	0	0	1	1	0	0
1	1	0	1	1	1	0	1	1	1	0	1
1	1	0	1	1	1	1	0	1	1	0	0
1	1	0	1	1	1	1	1	1	1	0	1
1	1	1	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	1	0	0	0	0
1	1	1	0	0	0	1	0	0	0	1	0
1	1	1	0	0	0	1	1	0	0	1	0
1	1	1	0	0	1	0	0	0	1	0	0
1	1	1	0	0	1	0	1	0	1	0	0
1	1	1	0	0	1	1	0	0	1	1	0
1	1	1	0	0	1	1	1	0	1	1	0
1	1	1	0	1	0	0	0	1	0	0	0
1	1	1	0	1	0	0	1	1	0	0	0
1	1	1	0	1	0	1	0	1	0	1	0
1	1	1	0	1	0	1	1	1	0	1	0
1	1	1	0	1	1	0	0	1	1	0	0
1	1	1	0	1	1	0	1	1	1	0	0
1	1	1	0	1	1	1	0	1	1	1	0
1	1	1	0	1	1	1	1	1	1	1	0
1	1	1	1	0	0	0	0	0	0	0	0
1	1	1	1	0	0	0	1	0	0	0	1
1	1	1	1	0	0	1	0	0	0	1	0
1	1	1	1	0	0	1	1	0	0	1	1
1	1	1	1	0	1	0	0	0	1	0	0
1	1	1	1	0	1	0	1	0	1	0	1
1	1	1	1	0	1	1	0	0	1	1	0
1	1	1	1	0	1	1	1	0	1	1	1
1	1	1	1	1	0	0	0	1	0	0	0
1	1	1	1	1	0	0	1	1	0	0	1
1	1	1	1	1	0	1	0	1	0	1	0
1	1	1	1	1	0	1	1	1	0	1	1
1	1	1	1	1	1	0	0	1	1	0	0
1	1	1	1	1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	0	1	1	1	0
1	1	1	1	1	1	1	1	1	1	1	1

OP 01 – A1, A0, A3, A2											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	1	0	1	0	0	0	0
0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	1	1	1	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	1	0	0	0	0
0	0	0	0	1	0	1	0	0	0	0	0
0	0	0	0	1	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	1	1	0	1	0	0	0	0
0	0	0	0	1	1	1	0	0	0	0	0
0	0	0	0	1	1	1	1	0	0	0	0
0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	1	0	1	0	0
0	0	0	1	0	0	1	0	0	1	0	0
0	0	0	1	0	0	1	1	0	1	0	0
0	0	0	1	0	1	0	0	0	1	0	0
0	0	0	1	0	1	0	1	0	1	0	0
0	0	0	1	0	1	1	0	0	1	0	0
0	0	0	1	0	1	1	1	0	1	0	0
0	0	0	1	1	0	0	0	0	1	0	0
0	0	0	1	1	0	0	1	0	1	0	0
0	0	0	1	1	0	1	0	0	1	0	0
0	0	0	1	1	0	1	1	0	1	0	0
0	0	0	1	1	1	0	0	0	1	0	0
0	0	0	1	1	1	0	1	0	1	0	0
0	0	0	1	1	1	1	0	0	1	0	0
0	0	0	1	1	1	1	1	0	1	0	0
0	0	1	0	0	0	0	0	1	0	0	0
0	0	1	0	0	0	0	1	1	0	0	0
0	0	1	0	0	0	1	0	1	0	0	0
0	0	1	0	0	0	1	1	1	0	0	0
0	0	1	0	0	1	0	0	1	0	0	0
0	0	1	0	0	1	0	1	1	0	0	0
0	0	1	0	0	1	1	0	1	0	0	0
0	0	1	0	0	1	1	1	1	0	0	0
0	0	1	0	1	0	0	0	1	0	0	0
0	0	1	0	1	0	0	1	1	0	0	0
0	0	1	0	1	0	1	0	1	0	0	0

OP 01 – A1, A0, A3, A2											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
0	0	1	0	1	0	1	1	1	0	0	0
0	0	1	0	1	1	0	0	1	0	0	0
0	0	1	0	1	1	0	1	1	0	0	0
0	0	1	0	1	1	1	0	1	0	0	0
0	0	1	0	1	1	1	1	1	0	0	0
0	0	1	1	0	0	0	0	1	1	0	0
0	0	1	1	0	0	0	1	1	1	0	0
0	0	1	1	0	0	1	0	1	1	0	0
0	0	1	1	0	0	1	1	1	1	0	0
0	0	1	1	0	1	0	0	1	1	0	0
0	0	1	1	0	1	0	1	1	1	0	0
0	0	1	1	0	1	1	0	1	1	0	0
0	0	1	1	0	1	1	1	1	1	0	0
0	0	1	1	1	0	0	0	1	1	0	0
0	0	1	1	1	0	0	1	1	1	0	0
0	0	1	1	1	0	1	0	1	1	0	0
0	0	1	1	1	0	1	1	1	1	0	0
0	0	1	1	1	1	0	0	1	1	0	0
0	0	1	1	1	1	0	1	1	1	0	0
0	0	1	1	1	1	1	0	1	1	0	0
0	0	1	1	1	1	1	1	1	1	0	0
0	1	0	0	0	0	0	0	0	0	0	1
0	1	0	0	0	0	0	1	0	0	0	1
0	1	0	0	0	0	1	0	0	0	0	1
0	1	0	0	0	0	1	1	0	0	0	1
0	1	0	0	0	1	0	0	0	0	0	1
0	1	0	0	0	1	0	1	0	0	0	1
0	1	0	0	0	1	1	0	0	0	0	1
0	1	0	0	0	1	1	1	0	0	0	1
0	1	0	0	1	0	0	0	0	0	0	1
0	1	0	0	1	0	0	1	0	0	0	1
0	1	0	0	1	0	1	0	0	0	0	1
0	1	0	0	1	0	1	1	0	0	0	1
0	1	0	0	1	1	0	0	0	0	0	1
0	1	0	0	1	1	0	1	0	0	0	1
0	1	0	0	1	1	1	0	0	0	0	1
0	1	0	0	1	1	1	1	0	0	0	1
0	1	0	1	0	0	0	0	0	1	0	1
0	1	0	1	0	0	0	1	0	1	0	1
0	1	0	1	0	0	1	0	0	1	0	1
0	1	0	1	0	0	1	1	0	1	0	1
0	1	0	1	0	1	0	0	0	1	0	1
0	1	0	1	0	1	0	1	0	1	0	1

OP 01 – A1, A0, A3, A2											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
0	1	0	1	0	1	1	0	0	1	0	1
0	1	0	1	0	1	1	1	0	1	0	1
0	1	0	1	1	0	0	0	0	1	0	1
0	1	0	1	1	0	0	1	0	1	0	1
0	1	0	1	1	0	1	0	0	1	0	1
0	1	0	1	1	0	1	1	0	1	0	1
0	1	0	1	1	1	0	0	0	1	0	1
0	1	0	1	1	1	0	1	0	1	0	1
0	1	0	1	1	1	1	0	0	1	0	1
0	1	0	1	1	1	1	1	0	1	0	1
0	1	1	0	0	0	0	0	1	0	0	1
0	1	1	0	0	0	0	1	1	0	0	1
0	1	1	0	0	0	1	0	1	0	0	1
0	1	1	0	0	0	1	1	1	0	0	1
0	1	1	0	0	1	0	0	1	0	0	1
0	1	1	0	0	1	0	1	1	0	0	1
0	1	1	0	0	1	1	0	1	0	0	1
0	1	1	0	0	1	1	1	1	0	0	1
0	1	1	0	1	0	0	0	1	0	0	1
0	1	1	0	1	0	0	1	1	0	0	1
0	1	1	0	1	0	1	0	1	0	0	1
0	1	1	0	1	0	1	1	1	0	0	1
0	1	1	0	1	1	0	0	1	0	0	1
0	1	1	0	1	1	0	1	1	0	0	1
0	1	1	0	1	1	1	0	1	0	0	1
0	1	1	0	1	1	1	1	1	0	0	1
0	1	1	1	0	0	0	0	1	1	0	1
0	1	1	1	0	0	0	1	1	1	0	1
0	1	1	1	0	0	1	0	1	1	0	1
0	1	1	1	0	0	1	1	1	1	0	1
0	1	1	1	0	1	0	0	1	1	0	1
0	1	1	1	0	1	0	1	1	1	0	1
0	1	1	1	0	1	1	0	1	1	0	1
0	1	1	1	0	1	1	1	1	1	0	1
0	1	1	1	1	0	0	0	1	1	0	1
0	1	1	1	1	0	0	1	1	1	0	1
0	1	1	1	1	0	1	0	1	1	0	1
0	1	1	1	1	0	1	1	1	1	0	1
0	1	1	1	1	1	0	0	1	1	0	1
0	1	1	1	1	1	0	1	1	1	0	1
0	1	1	1	1	1	1	0	1	1	0	1
0	1	1	1	1	1	1	1	1	1	0	1
1	0	0	0	0	0	0	0	0	0	1	0

OP 01 – A1, A0, A3, A2											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
1	0	0	0	0	0	0	1	0	0	1	0
1	0	0	0	0	0	1	0	0	0	1	0
1	0	0	0	0	0	1	1	0	0	1	0
1	0	0	0	0	1	0	0	0	0	1	0
1	0	0	0	0	1	0	1	0	0	1	0
1	0	0	0	0	1	1	0	0	0	1	0
1	0	0	0	0	1	1	1	0	0	1	0
1	0	0	0	1	0	0	0	0	0	1	0
1	0	0	0	1	0	0	1	0	0	1	0
1	0	0	0	1	0	1	0	0	0	1	0
1	0	0	0	1	0	1	1	0	0	1	0
1	0	0	0	1	1	0	0	0	0	1	0
1	0	0	0	1	1	0	1	0	0	1	0
1	0	0	0	1	1	1	0	0	0	1	0
1	0	0	0	1	1	1	1	0	0	1	0
1	0	0	1	0	0	0	0	0	1	1	0
1	0	0	1	0	0	0	1	0	1	1	0
1	0	0	1	0	0	1	0	0	1	1	0
1	0	0	1	0	0	1	1	0	1	1	0
1	0	0	1	0	1	0	0	0	1	1	0
1	0	0	1	0	1	0	1	0	1	1	0
1	0	0	1	0	1	1	0	0	1	1	0
1	0	0	1	0	1	1	1	0	1	1	0
1	0	0	1	1	0	0	0	0	1	1	0
1	0	0	1	1	0	0	1	0	1	1	0
1	0	0	1	1	0	1	0	0	1	1	0
1	0	0	1	1	0	1	1	0	1	1	0
1	0	0	1	1	1	0	0	0	1	1	0
1	0	0	1	1	1	0	1	0	1	1	0
1	0	0	1	1	1	1	0	0	1	1	0
1	0	0	1	1	1	1	1	0	1	1	0
1	0	1	0	0	0	0	0	1	0	1	0
1	0	1	0	0	0	0	1	1	0	1	0
1	0	1	0	0	0	1	0	1	0	1	0
1	0	1	0	0	0	1	1	1	0	1	0
1	0	1	0	0	1	0	0	1	0	1	0
1	0	1	0	0	1	0	1	1	0	1	0
1	0	1	0	0	1	1	0	1	0	1	0
1	0	1	0	0	1	1	1	1	0	1	0
1	0	1	0	1	0	0	0	1	0	1	0
1	0	1	0	1	0	0	1	1	0	1	0
1	0	1	0	1	0	1	0	1	0	1	0
1	0	1	0	1	0	1	1	1	0	1	0

OP 01 – A1, A0, A3, A2											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
1	0	1	0	1	1	0	0	1	0	1	0
1	0	1	0	1	1	0	1	1	0	1	0
1	0	1	0	1	1	1	0	1	0	1	0
1	0	1	0	1	1	1	1	1	0	1	0
1	0	1	1	0	0	0	0	1	1	1	0
1	0	1	1	0	0	0	1	1	1	1	0
1	0	1	1	0	0	1	0	1	1	1	0
1	0	1	1	0	0	1	1	1	1	1	0
1	0	1	1	0	1	0	0	1	1	1	0
1	0	1	1	0	1	0	1	1	1	1	0
1	0	1	1	0	1	1	0	1	1	1	0
1	0	1	1	0	1	1	1	1	1	1	0
1	0	1	1	1	0	0	0	1	1	1	0
1	0	1	1	1	0	0	1	1	1	1	0
1	0	1	1	1	0	1	0	1	1	1	0
1	0	1	1	1	0	1	1	1	1	1	0
1	0	1	1	1	1	0	0	1	1	1	0
1	0	1	1	1	1	0	1	1	1	1	0
1	0	1	1	1	1	1	0	1	1	1	0
1	0	1	1	1	1	1	1	1	1	1	0
1	1	0	0	0	0	0	0	0	0	1	1
1	1	0	0	0	0	0	1	0	0	1	1
1	1	0	0	0	0	1	0	0	0	1	1
1	1	0	0	0	0	1	1	0	0	1	1
1	1	0	0	0	1	0	0	0	0	1	1
1	1	0	0	0	1	0	1	0	0	1	1
1	1	0	0	0	1	1	0	0	0	1	1
1	1	0	0	0	1	1	1	0	0	1	1
1	1	0	0	1	0	0	0	0	0	1	1
1	1	0	0	1	0	0	1	0	0	1	1
1	1	0	0	1	0	1	0	0	0	1	1
1	1	0	0	1	0	1	1	0	0	1	1
1	1	0	0	1	1	0	0	0	0	1	1
1	1	0	0	1	1	0	1	0	0	1	1
1	1	0	0	1	1	1	0	0	0	1	1
1	1	0	0	1	1	1	1	0	0	1	1
1	1	0	1	0	0	0	0	0	1	1	1
1	1	0	1	0	0	0	1	0	1	1	1
1	1	0	1	0	0	1	0	0	1	1	1
1	1	0	1	0	0	1	1	0	1	1	1
1	1	0	1	0	1	0	0	0	1	1	1
1	1	0	1	0	1	0	1	0	1	1	1
1	1	0	1	0	1	1	0	0	1	1	1

OP 01 – A1, A0, A3, A2											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
1	1	0	1	0	1	1	1	0	1	1	1
1	1	0	1	1	0	0	0	0	1	1	1
1	1	0	1	1	0	0	1	0	1	1	1
1	1	0	1	1	0	1	0	0	1	1	1
1	1	0	1	1	0	1	1	0	1	1	1
1	1	0	1	1	1	0	0	0	1	1	1
1	1	0	1	1	1	0	1	0	1	1	1
1	1	0	1	1	1	1	0	0	1	1	1
1	1	0	1	1	1	1	1	0	1	1	1
1	1	1	0	0	0	0	0	1	0	1	1
1	1	1	0	0	0	0	1	1	0	1	1
1	1	1	0	0	0	1	0	1	0	1	1
1	1	1	0	0	0	1	1	1	0	1	1
1	1	1	0	0	1	0	0	1	0	1	1
1	1	1	0	0	1	0	1	1	0	1	1
1	1	1	0	0	1	1	0	1	0	1	1
1	1	1	0	0	1	1	1	1	0	1	1
1	1	1	0	1	0	0	0	1	0	1	1
1	1	1	0	1	0	0	1	1	0	1	1
1	1	1	0	1	0	1	0	1	0	1	1
1	1	1	0	1	0	1	1	1	0	1	1
1	1	1	0	1	1	0	0	1	0	1	1
1	1	1	0	1	1	0	1	1	0	1	1
1	1	1	0	1	1	1	0	1	0	1	1
1	1	1	0	1	1	1	1	1	0	1	1
1	1	1	1	0	0	0	0	1	1	1	1
1	1	1	1	0	0	0	1	1	1	1	1
1	1	1	1	0	0	1	0	1	1	1	1
1	1	1	1	0	0	1	1	1	1	1	1
1	1	1	1	0	1	0	0	1	1	1	1
1	1	1	1	0	1	0	1	1	1	1	1
1	1	1	1	0	1	1	0	1	1	1	1
1	1	1	1	0	1	1	1	1	1	1	1
1	1	1	1	1	0	0	0	1	1	1	1
1	1	1	1	1	0	0	1	1	1	1	1
1	1	1	1	1	0	1	0	1	1	1	1
1	1	1	1	1	0	1	1	1	1	1	1
1	1	1	1	1	1	0	0	1	1	1	1
1	1	1	1	1	1	0	1	1	1	1	1
1	1	1	1	1	1	1	0	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1

OP 10 – A == B											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
0	0	0	0	0	0	0	0	1	1	1	1
0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	1	0	1	0	0	0	0
0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	1	1	1	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	1	0	0	0	0
0	0	0	0	1	0	1	0	0	0	0	0
0	0	0	0	1	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	1	1	0	1	0	0	0	0
0	0	0	0	1	1	1	0	0	0	0	0
0	0	0	0	1	1	1	1	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1	1	1	1	1
0	0	0	1	0	0	1	0	0	0	0	0
0	0	0	1	0	0	1	1	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0
0	0	0	1	0	1	0	1	0	0	0	0
0	0	0	1	0	1	1	0	0	0	0	0
0	0	0	1	0	1	1	1	0	0	0	0
0	0	0	1	1	0	0	0	0	0	0	0
0	0	0	1	1	0	0	1	0	0	0	0
0	0	0	1	1	0	1	0	0	0	0	0
0	0	0	1	1	0	1	1	0	0	0	0
0	0	0	1	1	1	0	0	0	0	0	0
0	0	0	1	1	1	0	1	0	0	0	0
0	0	0	1	1	1	1	0	0	0	0	0
0	0	0	1	1	1	1	1	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	1	0	0	0	0
0	0	1	0	0	0	1	0	1	1	1	1
0	0	1	0	0	0	1	1	0	0	0	0
0	0	1	0	0	1	0	0	0	0	0	0
0	0	1	0	0	1	0	1	0	0	0	0
0	0	1	0	0	1	1	0	0	0	0	0
0	0	1	0	0	1	1	1	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	1	0	0	0	0
0	0	1	0	1	0	1	0	0	0	0	0

OP 10 – A == B											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
0	0	1	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	0	0	0	0	0
0	0	1	0	1	1	0	1	0	0	0	0
0	0	1	0	1	1	1	0	0	0	0	0
0	0	1	0	1	1	1	1	0	0	0	0
0	0	1	1	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	1	0	0	0	0
0	0	1	1	0	0	1	0	0	0	0	0
0	0	1	1	0	0	1	1	1	1	1	1
0	0	1	1	0	1	0	0	0	0	0	0
0	0	1	1	0	1	0	1	0	0	0	0
0	0	1	1	0	1	1	0	0	0	0	0
0	0	1	1	0	1	1	1	0	0	0	0
0	0	1	1	1	0	0	0	0	0	0	0
0	0	1	1	1	0	0	1	0	0	0	0
0	0	1	1	1	0	1	0	0	0	0	0
0	0	1	1	1	0	1	1	0	0	0	0
0	0	1	1	1	1	0	0	0	0	0	0
0	0	1	1	1	1	0	1	0	0	0	0
0	0	1	1	1	1	1	0	0	0	0	0
0	0	1	1	1	1	1	1	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	1	0	0	0	0
0	1	0	0	0	0	1	0	0	0	0	0
0	1	0	0	0	0	1	1	0	0	0	0
0	1	0	0	0	1	0	0	1	1	1	1
0	1	0	0	0	1	0	1	0	0	0	0
0	1	0	0	0	1	1	0	0	0	0	0
0	1	0	0	0	1	1	1	0	0	0	0
0	1	0	0	1	0	0	0	0	0	0	0
0	1	0	0	1	0	0	1	0	0	0	0
0	1	0	0	1	0	1	0	0	0	0	0
0	1	0	0	1	0	1	1	0	0	0	0
0	1	0	0	1	1	0	0	0	0	0	0
0	1	0	0	1	1	0	1	0	0	0	0
0	1	0	0	1	1	1	0	0	0	0	0
0	1	0	0	1	1	1	1	0	0	0	0
0	1	0	1	0	0	0	0	0	0	0	0
0	1	0	1	0	0	0	1	0	0	0	0
0	1	0	1	0	0	1	0	0	0	0	0
0	1	0	1	0	0	1	1	0	0	0	0
0	1	0	1	0	1	0	0	0	0	0	0
0	1	0	1	0	1	0	1	1	1	1	1

OP 10 – A == B											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
0	1	0	1	0	1	1	0	0	0	0	0
0	1	0	1	0	1	1	1	0	0	0	0
0	1	0	1	1	0	0	0	0	0	0	0
0	1	0	1	1	0	0	1	0	0	0	0
0	1	0	1	1	0	1	0	0	0	0	0
0	1	0	1	1	0	1	1	0	0	0	0
0	1	0	1	1	1	0	0	0	0	0	0
0	1	0	1	1	1	0	1	0	0	0	0
0	1	0	1	1	1	1	0	0	0	0	0
0	1	0	1	1	1	1	1	0	0	0	0
0	1	1	0	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	1	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	1	0	0	0	0
0	1	1	0	0	1	0	0	0	0	0	0
0	1	1	0	0	1	0	1	0	0	0	0
0	1	1	0	0	1	1	0	1	1	1	1
0	1	1	0	0	1	1	1	0	0	0	0
0	1	1	0	1	0	0	0	0	0	0	0
0	1	1	0	1	0	0	1	0	0	0	0
0	1	1	0	1	0	1	0	0	0	0	0
0	1	1	0	1	0	1	1	0	0	0	0
0	1	1	0	1	1	0	0	0	0	0	0
0	1	1	0	1	1	0	1	0	0	0	0
0	1	1	0	1	1	1	0	0	0	0	0
0	1	1	0	1	1	1	1	0	0	0	0
0	1	1	1	0	0	0	0	0	0	0	0
0	1	1	1	0	0	0	1	0	0	0	0
0	1	1	1	0	0	1	0	0	0	0	0
0	1	1	1	0	0	1	1	0	0	0	0
0	1	1	1	0	1	0	0	0	0	0	0
0	1	1	1	0	1	0	1	0	0	0	0
0	1	1	1	0	1	1	0	0	0	0	0
0	1	1	1	0	1	1	1	1	1	1	1
0	1	1	1	1	0	0	0	0	0	0	0
0	1	1	1	1	0	0	1	0	0	0	0
0	1	1	1	1	0	1	0	0	0	0	0
0	1	1	1	1	0	1	1	0	0	0	0
0	1	1	1	1	1	0	0	0	0	0	0
0	1	1	1	1	1	0	1	0	0	0	0
0	1	1	1	1	1	1	0	0	0	0	0
0	1	1	1	1	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0

OP 10 – A == B											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
1	0	0	0	0	0	0	1	0	0	0	0
1	0	0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	0	1	1	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0	0
1	0	0	0	0	1	0	1	0	0	0	0
1	0	0	0	0	1	1	0	0	0	0	0
1	0	0	0	0	1	1	1	0	0	0	0
1	0	0	0	1	0	0	0	1	1	1	1
1	0	0	0	1	0	0	1	0	0	0	0
1	0	0	0	1	0	1	0	0	0	0	0
1	0	0	0	1	0	1	1	0	0	0	0
1	0	0	0	1	1	0	0	0	0	0	0
1	0	0	0	1	1	0	1	0	0	0	0
1	0	0	0	1	1	1	0	0	0	0	0
1	0	0	0	1	1	1	1	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	1	0	0	0	0
1	0	0	1	0	0	1	0	0	0	0	0
1	0	0	1	0	0	1	1	0	0	0	0
1	0	0	1	0	1	0	0	0	0	0	0
1	0	0	1	0	1	0	1	0	0	0	0
1	0	0	1	0	1	1	0	0	0	0	0
1	0	0	1	0	1	1	1	0	0	0	0
1	0	0	1	1	0	0	0	0	0	0	0
1	0	0	1	1	0	0	1	1	1	1	1
1	0	0	1	1	0	1	0	0	0	0	0
1	0	0	1	1	0	1	1	0	0	0	0
1	0	0	1	1	1	0	0	0	0	0	0
1	0	0	1	1	1	0	1	0	0	0	0
1	0	0	1	1	1	1	0	0	0	0	0
1	0	0	1	1	1	1	1	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	1	0	0	0	0
1	0	1	0	0	0	1	0	0	0	0	0
1	0	1	0	0	0	1	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0	0
1	0	1	0	0	1	0	1	0	0	0	0
1	0	1	0	0	1	1	0	0	0	0	0
1	0	1	0	0	1	1	1	0	0	0	0
1	0	1	0	1	0	0	0	0	0	0	0
1	0	1	0	1	0	0	1	0	0	0	0
1	0	1	0	1	0	1	0	1	1	1	1
1	0	1	0	1	0	1	1	0	0	0	0

OP 10 – A == B											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
1	0	1	0	1	1	0	0	0	0	0	0
1	0	1	0	1	1	0	1	0	0	0	0
1	0	1	0	1	1	1	0	0	0	0	0
1	0	1	0	1	1	1	1	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0	0
1	0	1	1	0	0	0	1	0	0	0	0
1	0	1	1	0	0	1	0	0	0	0	0
1	0	1	1	0	0	1	1	0	0	0	0
1	0	1	1	0	1	0	0	0	0	0	0
1	0	1	1	0	1	0	1	0	0	0	0
1	0	1	1	0	1	1	0	0	0	0	0
1	0	1	1	0	1	1	1	0	0	0	0
1	0	1	1	1	0	0	0	0	0	0	0
1	0	1	1	1	0	0	1	0	0	0	0
1	0	1	1	1	0	1	0	0	0	0	0
1	0	1	1	1	0	1	1	1	1	1	1
1	0	1	1	1	1	0	0	0	0	0	0
1	0	1	1	1	1	0	1	0	0	0	0
1	0	1	1	1	1	1	0	0	0	0	0
1	0	1	1	1	1	1	1	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	1	0	0	0	0
1	1	0	0	0	0	1	0	0	0	0	0
1	1	0	0	0	0	1	1	0	0	0	0
1	1	0	0	0	1	0	0	0	0	0	0
1	1	0	0	0	1	0	1	0	0	0	0
1	1	0	0	0	1	1	0	0	0	0	0
1	1	0	0	0	1	1	1	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0	0
1	1	0	0	1	0	0	1	0	0	0	0
1	1	0	0	1	0	1	0	0	0	0	0
1	1	0	0	1	0	1	1	0	0	0	0
1	1	0	0	1	1	0	0	1	1	1	1
1	1	0	0	1	1	0	1	0	0	0	0
1	1	0	0	1	1	1	0	0	0	0	0
1	1	0	0	1	1	1	1	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0	0
1	1	0	1	0	0	0	1	0	0	0	0
1	1	0	1	0	0	1	0	0	0	0	0
1	1	0	1	0	0	1	1	0	0	0	0
1	1	0	1	0	1	0	0	0	0	0	0
1	1	0	1	0	1	0	1	0	0	0	0
1	1	0	1	0	1	1	0	0	0	0	0

OP 10 – A == B											
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**S3**	**S2**	**S1**	**S0**
1	1	0	1	0	1	1	1	0	0	0	0
1	1	0	1	1	0	0	0	0	0	0	0
1	1	0	1	1	0	0	1	0	0	0	0
1	1	0	1	1	0	1	0	0	0	0	0
1	1	0	1	1	0	1	1	0	0	0	0
1	1	0	1	1	1	0	0	0	0	0	0
1	1	0	1	1	1	0	1	1	1	1	1
1	1	0	1	1	1	1	0	0	0	0	0
1	1	0	1	1	1	1	1	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	1	0	0	0	0
1	1	1	0	0	0	1	0	0	0	0	0
1	1	1	0	0	0	1	1	0	0	0	0
1	1	1	0	0	1	0	0	0	0	0	0
1	1	1	0	0	1	0	1	0	0	0	0
1	1	1	0	0	1	1	0	0	0	0	0
1	1	1	0	0	1	1	1	0	0	0	0
1	1	1	0	1	0	0	0	0	0	0	0
1	1	1	0	1	0	0	1	0	0	0	0
1	1	1	0	1	0	1	0	0	0	0	0
1	1	1	0	1	0	1	1	0	0	0	0
1	1	1	0	1	1	0	0	0	0	0	0
1	1	1	0	1	1	0	1	0	0	0	0
1	1	1	0	1	1	1	0	1	1	1	1
1	1	1	0	1	1	1	1	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0	0
1	1	1	1	0	0	0	1	0	0	0	0
1	1	1	1	0	0	1	0	0	0	0	0
1	1	1	1	0	0	1	1	0	0	0	0
1	1	1	1	0	1	0	0	0	0	0	0
1	1	1	1	0	1	0	1	0	0	0	0
1	1	1	1	0	1	1	0	0	0	0	0
1	1	1	1	0	1	1	1	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0
1	1	1	1	1	0	0	1	0	0	0	0
1	1	1	1	1	0	1	0	0	0	0	0
1	1	1	1	1	0	1	1	0	0	0	0
1	1	1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	0	1	0	0	0	0
1	1	1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1

OP 11 – A + B												
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**OF**	**S3**	**S2**	**S1**	**S0**
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	1
0	0	0	0	0	0	1	0	0	0	0	1	0
0	0	0	0	0	0	1	1	0	0	0	1	1
0	0	0	0	0	1	0	0	0	0	1	0	0
0	0	0	0	0	1	0	1	0	0	1	0	1
0	0	0	0	0	1	1	0	0	0	1	1	0
0	0	0	0	0	1	1	1	0	0	1	1	1
0	0	0	0	1	0	0	0	0	1	0	0	0
0	0	0	0	1	0	0	1	0	1	0	0	1
0	0	0	0	1	0	1	0	0	1	0	1	0
0	0	0	0	1	0	1	1	0	1	0	1	1
0	0	0	0	1	1	0	0	0	1	1	0	0
0	0	0	0	1	1	0	1	0	1	1	0	1
0	0	0	0	1	1	1	0	0	1	1	1	0
0	0	0	0	1	1	1	1	0	1	1	1	1
0	0	0	1	0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	1	0	0	0	1	0
0	0	0	1	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	1	1	0	0	1	0	0
0	0	0	1	0	1	0	0	0	0	1	0	1
0	0	0	1	0	1	0	1	0	0	1	1	0
0	0	0	1	0	1	1	0	0	0	1	1	1
0	0	0	1	0	1	1	1	0	1	0	0	0
0	0	0	1	1	0	0	0	0	1	0	0	1
0	0	0	1	1	0	0	1	0	1	0	1	0
0	0	0	1	1	0	1	0	0	1	0	1	1
0	0	0	1	1	0	1	1	0	1	1	0	0
0	0	0	1	1	1	0	0	0	1	1	0	1
0	0	0	1	1	1	0	1	0	1	1	1	0
0	0	0	1	1	1	1	0	0	1	1	1	1
0	0	0	1	1	1	1	1	1	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	1	0
0	0	1	0	0	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	0	0	0	1	0	0
0	0	1	0	0	0	1	1	0	0	1	0	1
0	0	1	0	0	1	0	0	0	0	1	1	0
0	0	1	0	0	1	0	1	0	0	1	1	1
0	0	1	0	0	1	1	0	0	1	0	0	0
0	0	1	0	0	1	1	1	0	1	0	0	1
0	0	1	0	1	0	0	0	0	1	0	1	0
0	0	1	0	1	0	0	1	0	1	0	1	1
0	0	1	0	1	0	1	0	0	1	1	0	0

OP 11 – A + B												
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**OF**	**S3**	**S2**	**S1**	**S0**
0	0	1	0	1	0	1	1	0	1	1	0	1
0	0	1	0	1	1	0	0	0	1	1	1	0
0	0	1	0	1	1	0	1	0	1	1	1	1
0	0	1	0	1	1	1	0	1	0	0	0	0
0	0	1	0	1	1	1	1	1	0	0	0	1
0	0	1	1	0	0	0	0	0	0	0	1	1
0	0	1	1	0	0	0	1	0	0	1	0	0
0	0	1	1	0	0	1	0	0	0	1	0	1
0	0	1	1	0	0	1	1	0	0	1	1	0
0	0	1	1	0	1	0	0	0	0	1	1	1
0	0	1	1	0	1	0	1	0	1	0	0	0
0	0	1	1	0	1	1	0	0	1	0	0	1
0	0	1	1	0	1	1	1	0	1	0	1	0
0	0	1	1	1	0	0	0	0	1	0	1	1
0	0	1	1	1	0	0	1	0	1	1	0	0
0	0	1	1	1	0	1	0	0	1	1	0	1
0	0	1	1	1	0	1	1	0	1	1	1	0
0	0	1	1	1	1	0	0	0	1	1	1	1
0	0	1	1	1	1	0	1	1	0	0	0	0
0	0	1	1	1	1	1	0	1	0	0	0	1
0	0	1	1	1	1	1	1	1	0	0	1	0
0	1	0	0	0	0	0	0	0	0	1	0	0
0	1	0	0	0	0	0	1	0	0	1	0	1
0	1	0	0	0	0	1	0	0	0	1	1	0
0	1	0	0	0	0	1	1	0	0	1	1	1
0	1	0	0	0	1	0	0	0	1	0	0	0
0	1	0	0	0	1	0	1	0	1	0	0	1
0	1	0	0	0	1	1	0	0	1	0	1	0
0	1	0	0	0	1	1	1	0	1	0	1	1
0	1	0	0	1	0	0	0	0	1	1	0	0
0	1	0	0	1	0	0	1	0	1	1	0	1
0	1	0	0	1	0	1	0	0	1	1	1	0
0	1	0	0	1	0	1	1	0	1	1	1	1
0	1	0	0	1	1	0	0	1	0	0	0	0
0	1	0	0	1	1	0	1	1	0	0	0	1
0	1	0	0	1	1	1	0	1	0	0	1	0
0	1	0	0	1	1	1	1	1	0	0	1	1
0	1	0	1	0	0	0	0	0	0	1	0	1
0	1	0	1	0	0	0	1	0	0	1	1	0
0	1	0	1	0	0	1	0	0	0	1	1	1
0	1	0	1	0	0	1	1	0	1	0	0	0
0	1	0	1	0	1	0	0	0	1	0	0	1
0	1	0	1	0	1	0	1	0	1	0	1	0

OP 11 – A + B												
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**OF**	**S3**	**S2**	**S1**	**S0**
0	1	0	1	0	1	1	0	0	1	0	1	1
0	1	0	1	0	1	1	1	0	1	1	0	0
0	1	0	1	1	0	0	0	0	1	1	0	1
0	1	0	1	1	0	0	1	0	1	1	1	0
0	1	0	1	1	0	1	0	0	1	1	1	1
0	1	0	1	1	0	1	1	1	0	0	0	0
0	1	0	1	1	1	0	0	1	0	0	0	1
0	1	0	1	1	1	0	1	1	0	0	1	0
0	1	0	1	1	1	1	0	1	0	0	1	1
0	1	0	1	1	1	1	1	1	0	1	0	0
0	1	1	0	0	0	0	0	0	0	1	1	0
0	1	1	0	0	0	0	1	0	0	1	1	1
0	1	1	0	0	0	1	0	0	1	0	0	0
0	1	1	0	0	0	1	1	0	1	0	0	1
0	1	1	0	0	1	0	0	0	1	0	1	0
0	1	1	0	0	1	0	1	0	1	0	1	1
0	1	1	0	0	1	1	0	0	1	1	0	0
0	1	1	0	0	1	1	1	0	1	1	0	1
0	1	1	0	1	0	0	0	0	1	1	1	0
0	1	1	0	1	0	0	1	0	1	1	1	1
0	1	1	0	1	0	1	0	1	0	0	0	0
0	1	1	0	1	0	1	1	1	0	0	0	1
0	1	1	0	1	1	0	0	1	0	0	1	0
0	1	1	0	1	1	0	1	1	0	0	1	1
0	1	1	0	1	1	1	0	1	0	1	0	0
0	1	1	0	1	1	1	1	1	0	1	0	1
0	1	1	1	0	0	0	0	0	0	1	1	1
0	1	1	1	0	0	0	1	0	1	0	0	0
0	1	1	1	0	0	1	0	0	1	0	0	1
0	1	1	1	0	0	1	1	0	1	0	1	0
0	1	1	1	0	1	0	0	0	1	0	1	1
0	1	1	1	0	1	0	1	0	1	1	0	0
0	1	1	1	0	1	1	0	0	1	1	0	1
0	1	1	1	0	1	1	1	0	1	1	1	0
0	1	1	1	1	0	0	0	0	1	1	1	1
0	1	1	1	1	0	0	1	1	0	0	0	0
0	1	1	1	1	0	1	0	1	0	0	0	1
0	1	1	1	1	0	1	1	1	0	0	1	0
0	1	1	1	1	1	0	0	1	0	0	1	1
0	1	1	1	1	1	0	1	1	0	1	0	0
0	1	1	1	1	1	1	0	1	0	1	0	1
0	1	1	1	1	1	1	1	1	0	1	1	0
1	0	0	0	0	0	0	0	0	1	0	0	0

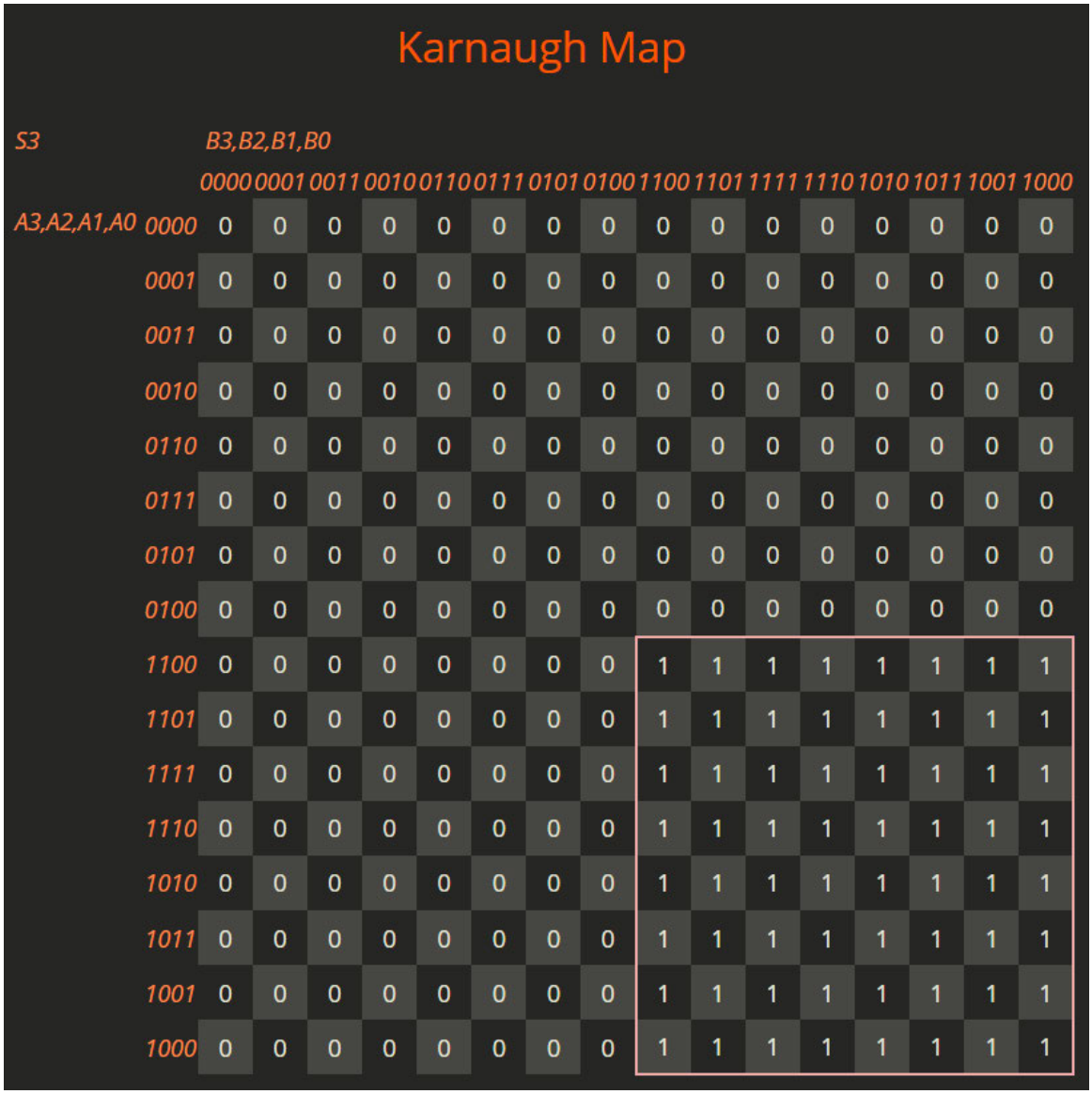
OP 11 – A + B												
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**OF**	**S3**	**S2**	**S1**	**S0**
1	0	0	0	0	0	0	1	0	1	0	0	1
1	0	0	0	0	0	1	0	0	1	0	1	0
1	0	0	0	0	0	1	1	0	1	0	1	1
1	0	0	0	0	1	0	0	0	1	1	0	0
1	0	0	0	0	1	0	1	0	1	1	0	1
1	0	0	0	0	1	1	0	0	1	1	1	0
1	0	0	0	0	1	1	1	0	1	1	1	1
1	0	0	0	1	0	0	0	1	0	0	0	0
1	0	0	0	1	0	0	1	1	0	0	0	1
1	0	0	0	1	0	1	0	1	0	0	1	0
1	0	0	0	1	0	1	1	1	0	0	1	1
1	0	0	0	1	1	0	0	1	0	1	0	0
1	0	0	0	1	1	0	1	1	0	1	0	1
1	0	0	0	1	1	1	0	1	0	1	1	0
1	0	0	0	1	1	1	1	1	0	1	1	1
1	0	0	1	0	0	0	0	0	1	0	0	1
1	0	0	1	0	0	0	1	0	1	0	1	0
1	0	0	1	0	0	1	0	0	1	0	1	1
1	0	0	1	0	0	1	1	0	1	1	0	0
1	0	0	1	0	1	0	0	0	1	1	0	1
1	0	0	1	0	1	0	1	0	1	1	1	0
1	0	0	1	0	1	1	0	0	1	1	1	1
1	0	0	1	0	1	1	1	1	0	0	0	0
1	0	0	1	1	0	0	0	1	0	0	0	1
1	0	0	1	1	0	0	1	1	0	0	1	0
1	0	0	1	1	0	1	0	1	0	0	1	1
1	0	0	1	1	0	1	1	1	0	1	0	0
1	0	0	1	1	1	0	0	1	0	1	0	1
1	0	0	1	1	1	0	1	1	0	1	1	0
1	0	0	1	1	1	1	0	1	0	1	1	1
1	0	0	1	1	1	1	1	1	1	0	0	0
1	0	1	0	0	0	0	0	0	1	0	1	0
1	0	1	0	0	0	0	1	0	1	0	1	1
1	0	1	0	0	0	1	0	0	1	1	0	0
1	0	1	0	0	0	1	1	0	1	1	0	1
1	0	1	0	0	1	0	0	0	1	1	1	0
1	0	1	0	0	1	0	1	0	1	1	1	1
1	0	1	0	0	1	1	0	1	0	0	0	0
1	0	1	0	0	1	1	1	1	0	0	0	1
1	0	1	0	1	0	0	0	1	0	0	1	0
1	0	1	0	1	0	0	1	1	0	0	1	1
1	0	1	0	1	0	1	0	1	0	1	0	0
1	0	1	0	1	0	1	1	1	0	1	0	1

OP 11 – A + B												
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**OF**	**S3**	**S2**	**S1**	**S0**
1	0	1	0	1	1	0	0	1	0	1	1	0
1	0	1	0	1	1	0	1	1	0	1	1	1
1	0	1	0	1	1	1	0	1	1	0	0	0
1	0	1	0	1	1	1	1	1	1	0	0	1
1	0	1	1	0	0	0	0	0	1	0	1	1
1	0	1	1	0	0	0	1	0	1	1	0	0
1	0	1	1	0	0	1	0	0	1	1	0	1
1	0	1	1	0	0	1	1	0	1	1	1	0
1	0	1	1	0	1	0	0	0	1	1	1	1
1	0	1	1	0	1	0	1	1	0	0	0	0
1	0	1	1	0	1	1	0	1	0	0	0	1
1	0	1	1	0	1	1	1	1	0	0	1	0
1	0	1	1	1	0	0	0	1	0	0	1	1
1	0	1	1	1	0	0	1	1	0	1	0	0
1	0	1	1	1	0	1	0	1	0	1	0	1
1	0	1	1	1	0	1	1	1	0	1	1	0
1	0	1	1	1	1	0	0	1	0	1	1	1
1	0	1	1	1	1	0	1	1	1	0	0	0
1	0	1	1	1	1	1	0	1	1	0	0	1
1	0	1	1	1	1	1	1	1	1	0	1	0
1	1	0	0	0	0	0	0	0	1	1	0	0
1	1	0	0	0	0	0	1	0	1	1	0	1
1	1	0	0	0	0	1	0	0	1	1	1	0
1	1	0	0	0	0	1	1	0	1	1	1	1
1	1	0	0	0	1	0	0	1	0	0	0	0
1	1	0	0	0	1	0	1	1	0	0	0	1
1	1	0	0	0	1	1	0	1	0	0	1	0
1	1	0	0	0	1	1	1	1	0	0	1	1
1	1	0	0	1	0	0	0	1	0	1	0	0
1	1	0	0	1	0	0	1	1	0	1	0	1
1	1	0	0	1	0	1	0	1	0	1	1	0
1	1	0	0	1	0	1	1	1	0	1	1	1
1	1	0	0	1	1	0	0	1	1	0	0	0
1	1	0	0	1	1	0	1	1	1	0	0	1
1	1	0	0	1	1	1	0	1	1	0	1	0
1	1	0	0	1	1	1	1	1	1	0	1	1
1	1	0	1	0	0	0	0	0	1	1	0	1
1	1	0	1	0	0	0	1	0	1	1	1	0
1	1	0	1	0	0	1	0	0	1	1	1	1
1	1	0	1	0	0	1	1	1	0	0	0	0
1	1	0	1	0	1	0	0	1	0	0	0	1
1	1	0	1	0	1	0	1	1	0	0	1	0
1	1	0	1	0	1	1	0	1	0	0	1	1

OP 11 – A + B												
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	**OF**	**S3**	**S2**	**S1**	**S0**
1	1	0	1	0	1	1	1	1	0	1	0	0
1	1	0	1	1	0	0	0	1	0	1	0	1
1	1	0	1	1	0	0	1	1	0	1	1	0
1	1	0	1	1	0	1	0	1	0	1	1	1
1	1	0	1	1	0	1	1	1	1	0	0	0
1	1	0	1	1	1	0	0	1	1	0	0	1
1	1	0	1	1	1	0	1	1	1	0	1	0
1	1	0	1	1	1	1	0	1	1	0	1	1
1	1	0	1	1	1	1	1	1	1	1	0	0
1	1	1	0	0	0	0	0	0	1	1	1	0
1	1	1	0	0	0	0	1	0	1	1	1	1
1	1	1	0	0	0	1	0	1	0	0	0	0
1	1	1	0	0	0	1	1	1	0	0	0	1
1	1	1	0	0	1	0	0	1	0	0	1	0
1	1	1	0	0	1	0	1	1	0	0	1	1
1	1	1	0	0	1	1	0	1	0	1	0	0
1	1	1	0	0	1	1	1	1	0	1	0	1
1	1	1	0	1	0	0	0	1	0	1	1	0
1	1	1	0	1	0	0	1	1	0	1	1	1
1	1	1	0	1	0	1	0	1	1	0	0	0
1	1	1	0	1	0	1	1	1	1	0	0	1
1	1	1	0	1	1	0	0	1	1	0	1	0
1	1	1	0	1	1	0	1	1	1	0	1	1
1	1	1	0	1	1	1	0	1	1	1	0	0
1	1	1	0	1	1	1	1	1	1	1	0	1
1	1	1	1	0	0	0	0	0	1	1	1	1
1	1	1	1	0	0	0	1	1	0	0	0	0
1	1	1	1	0	0	1	0	1	0	0	0	1
1	1	1	1	0	0	1	1	1	0	0	1	0
1	1	1	1	0	1	0	0	1	0	0	1	1
1	1	1	1	0	1	0	1	1	0	1	0	0
1	1	1	1	0	1	1	0	1	0	1	0	1
1	1	1	1	0	1	1	1	1	0	1	1	0
1	1	1	1	1	0	0	0	1	0	1	1	1
1	1	1	1	1	0	0	1	1	1	0	0	0
1	1	1	1	1	0	1	0	1	1	0	0	1
1	1	1	1	1	0	1	1	1	1	0	1	0
1	1	1	1	1	1	0	0	1	1	0	1	1
1	1	1	1	1	1	0	1	1	1	1	0	0
1	1	1	1	1	1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1	1	1	1	1	0

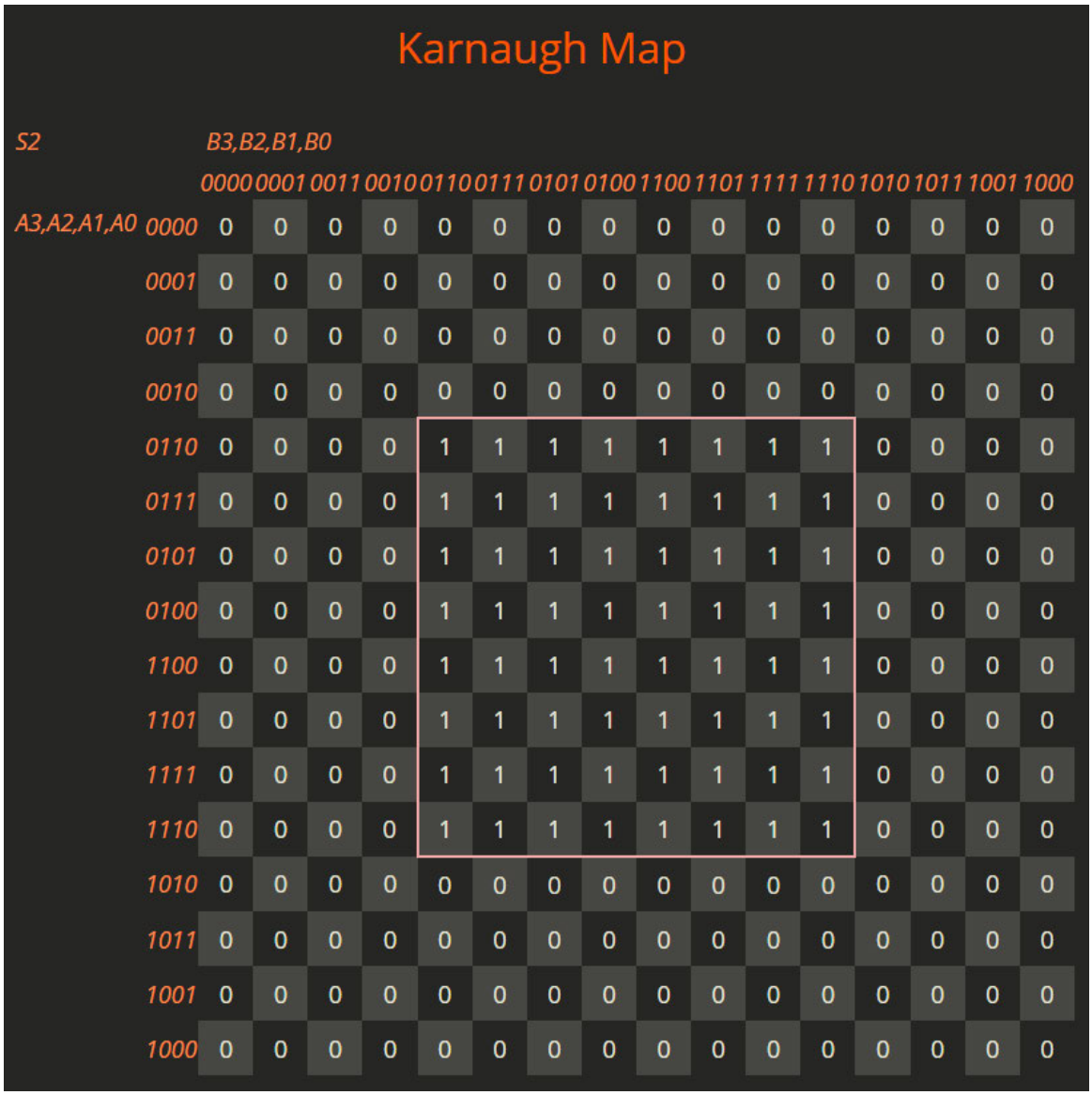
Appendix C – Full Karnaugh Maps

Operation 00 – A AND B



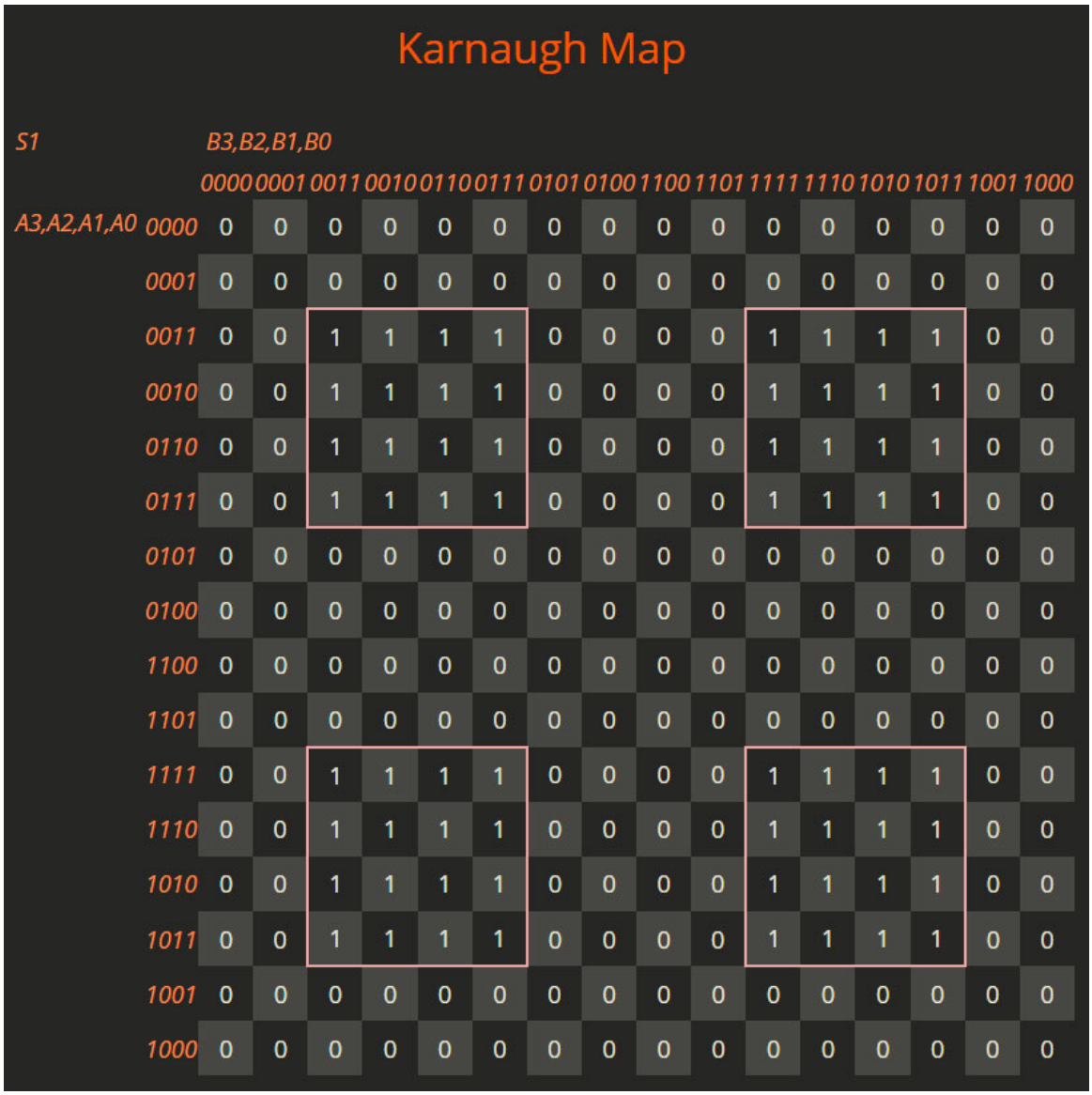
Characteristic Equation: A3B3

Operation 00 – A AND B



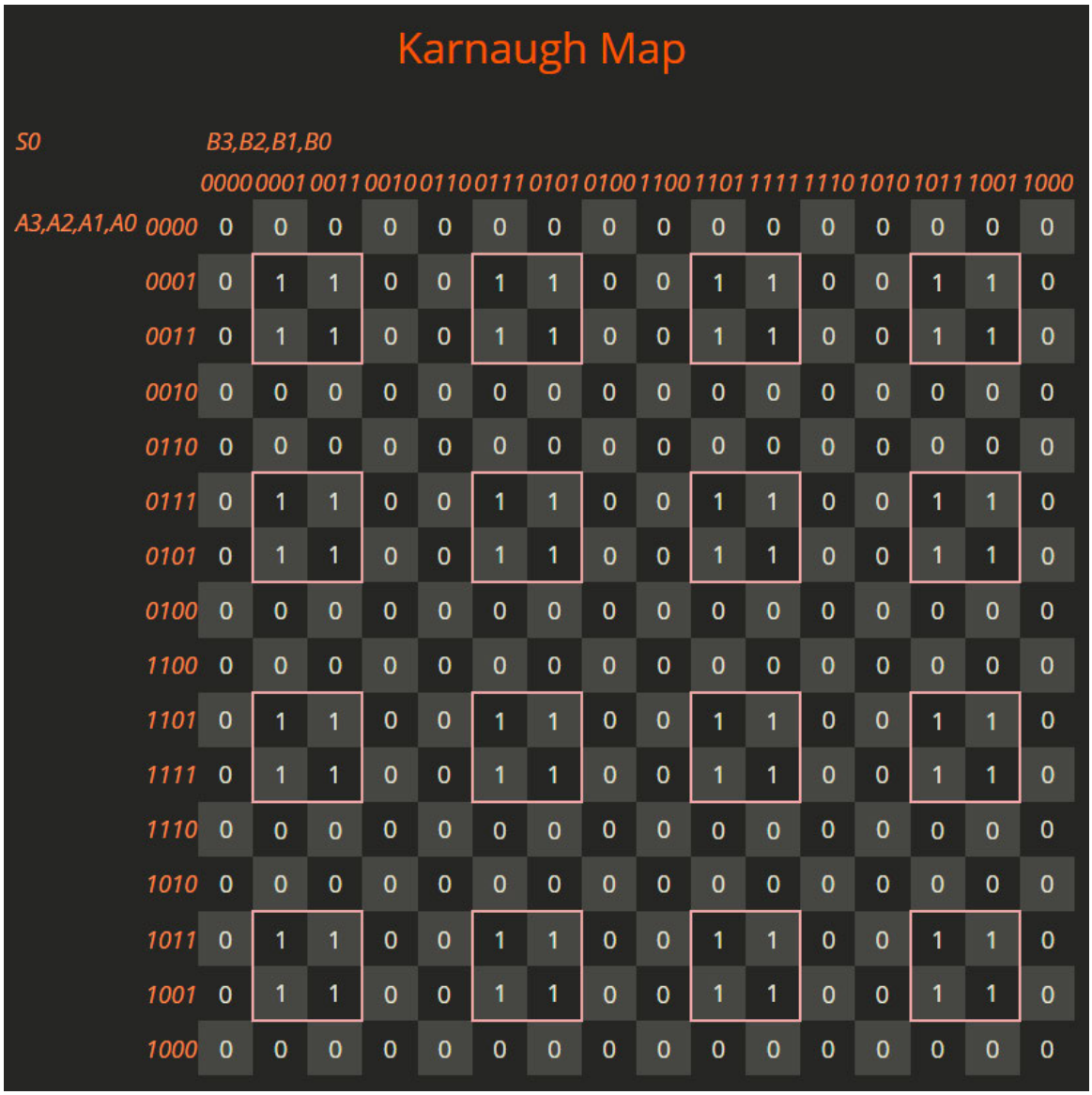
Characteristic Equation: A2B2

Operation 00 – A AND B



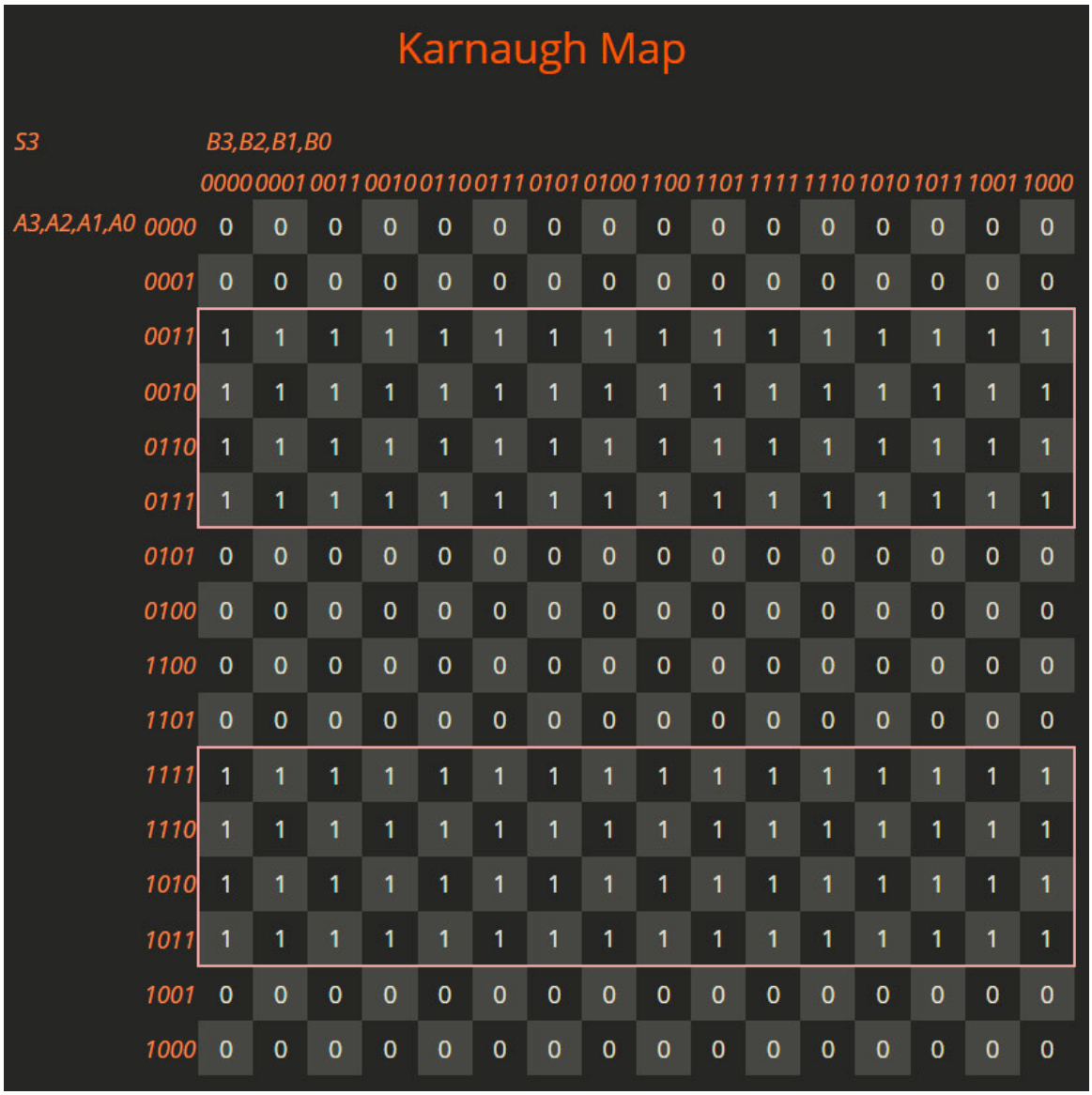
Characteristic Equation: A1B1

Operation 00 – A AND B



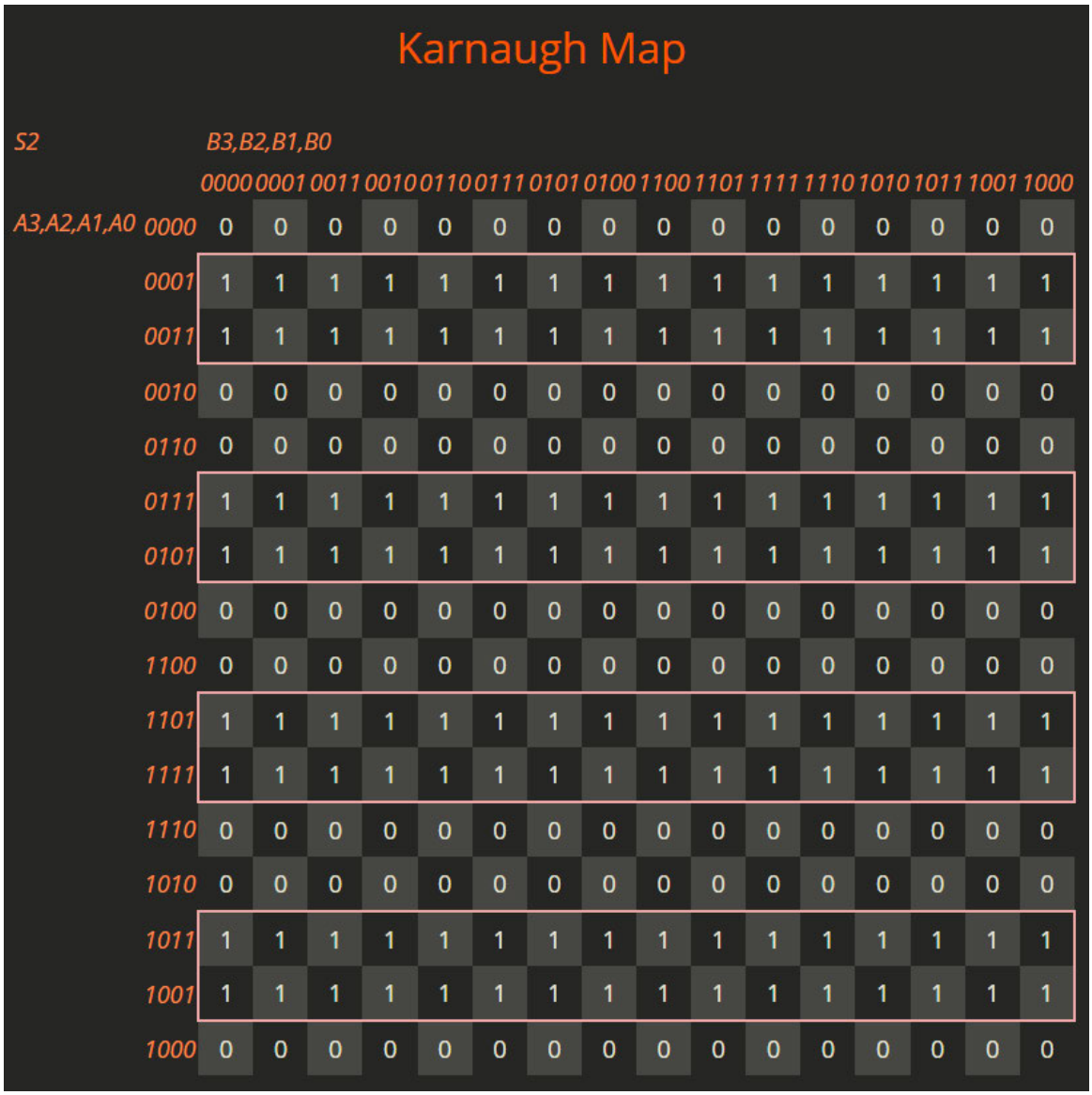
Characteristic Equation: A0B0

Operation 01 – A1, A0, A3. A2



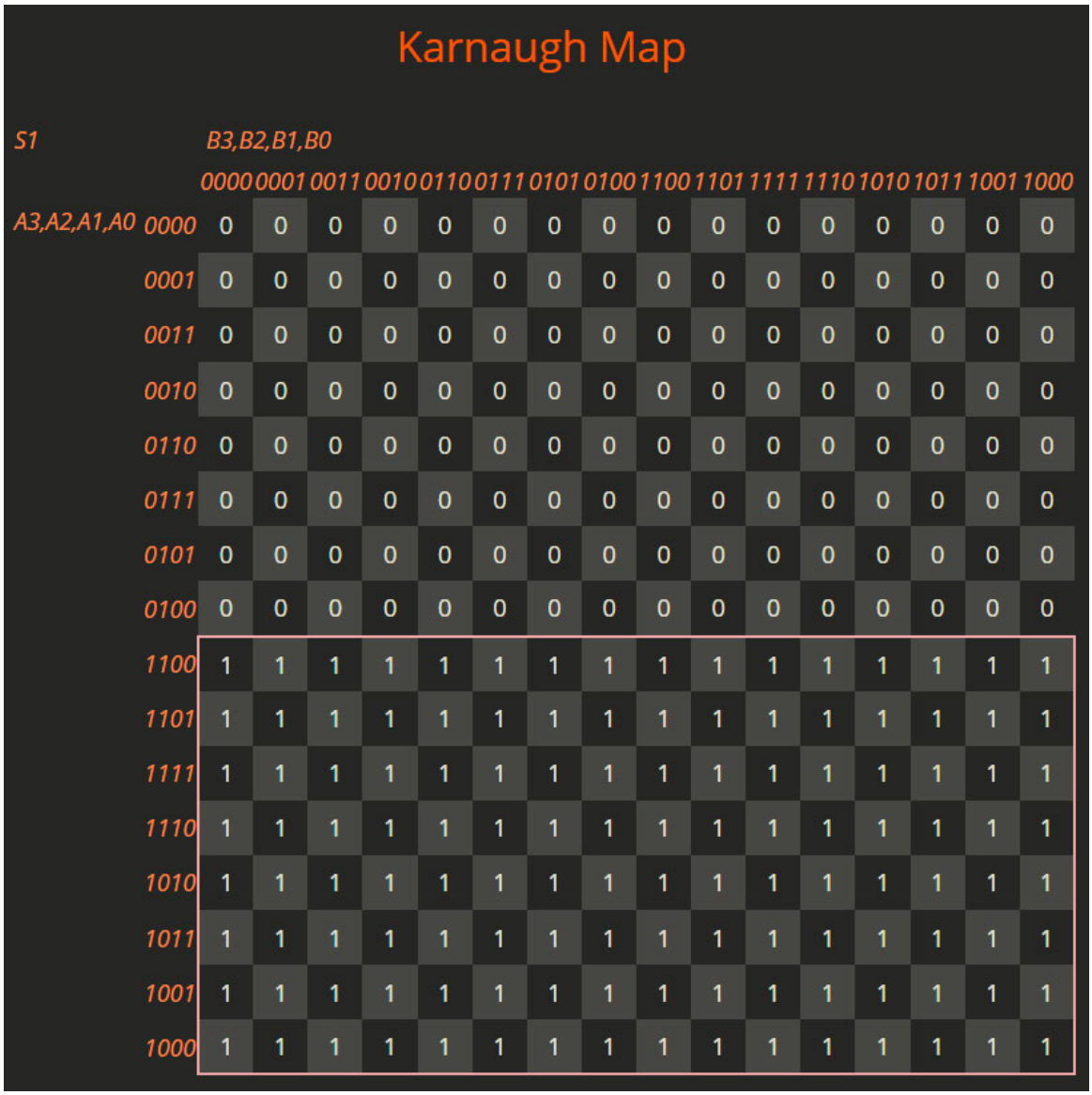
Characteristic Equation: A1

Operation 01 – A1, A0, A3. A2



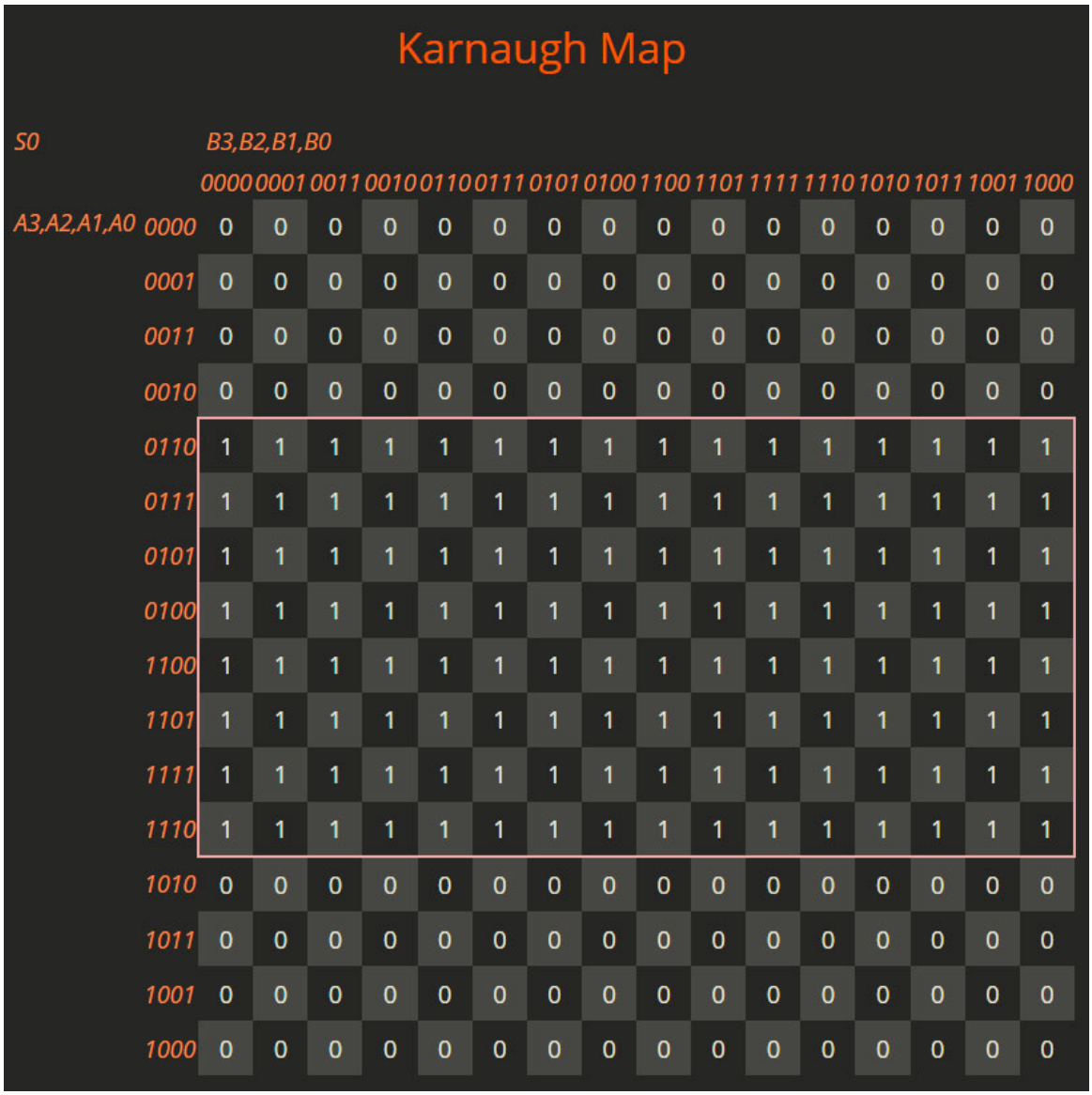
Characteristic Equation: A0

Operation 01 – A1, A0, A3. A2



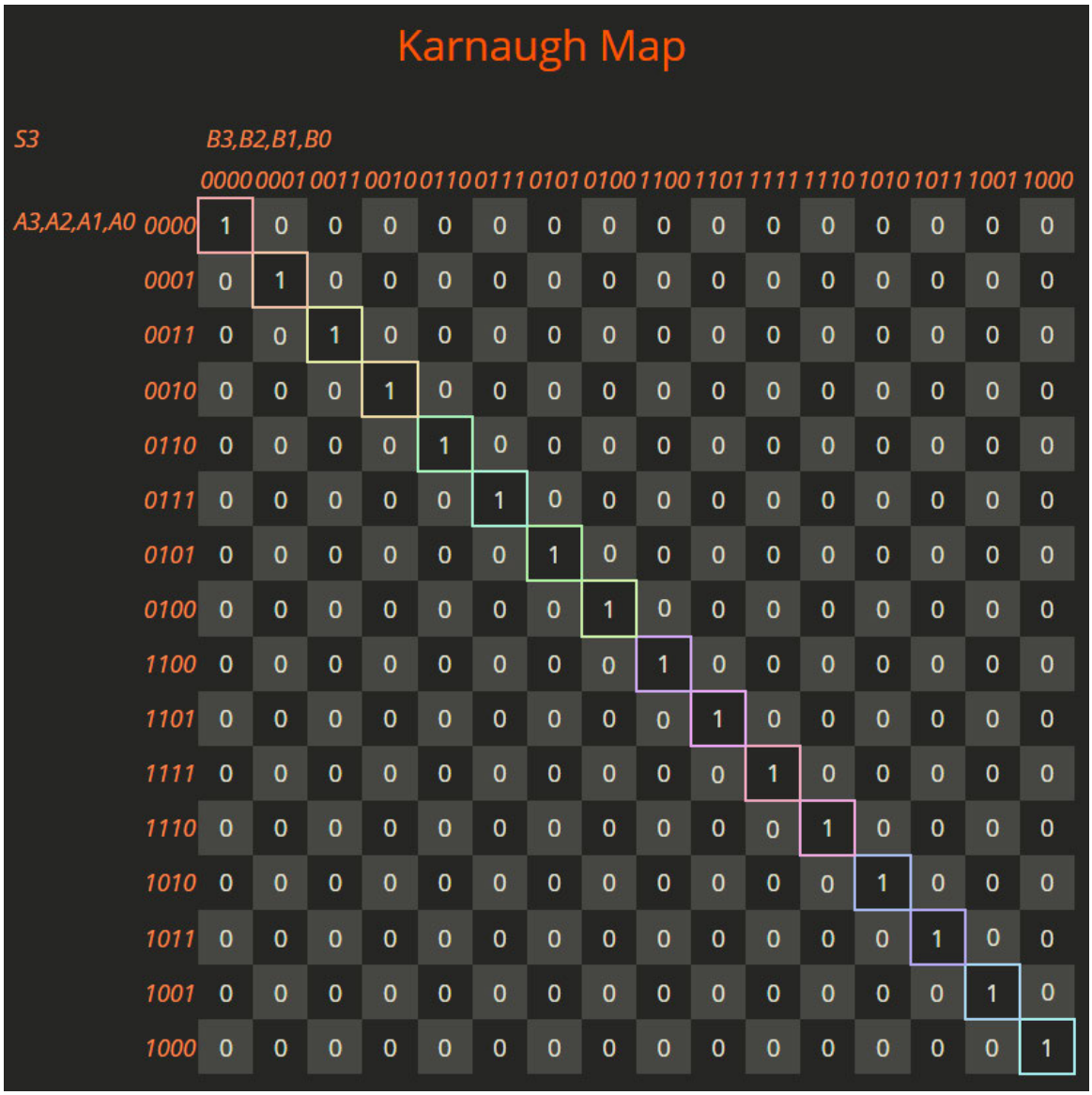
Characteristic Equation: A3

Operation 01 – A1, A0, A3. A2



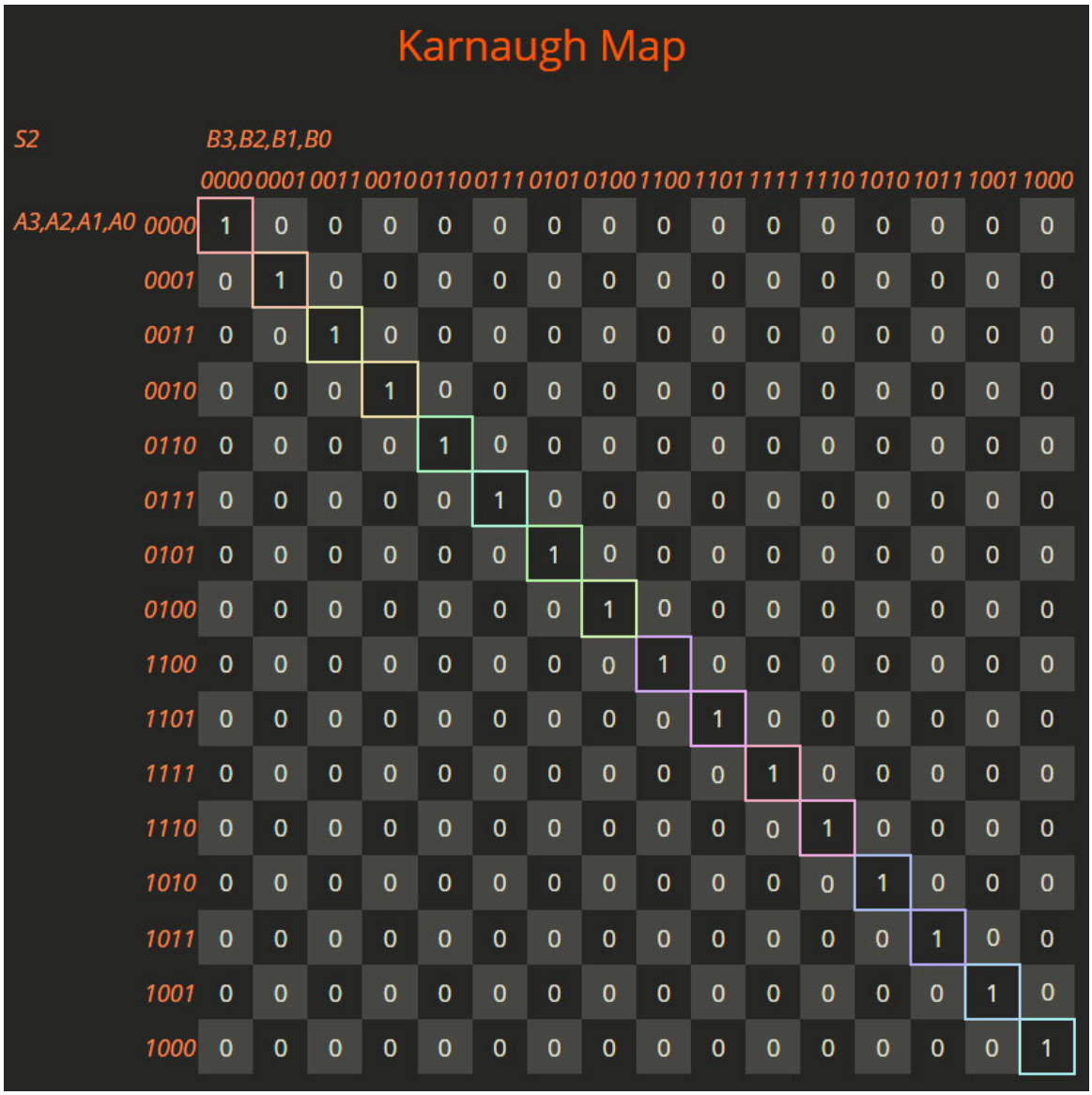
Characteristic Equation: A2

Operation 10 – A == B



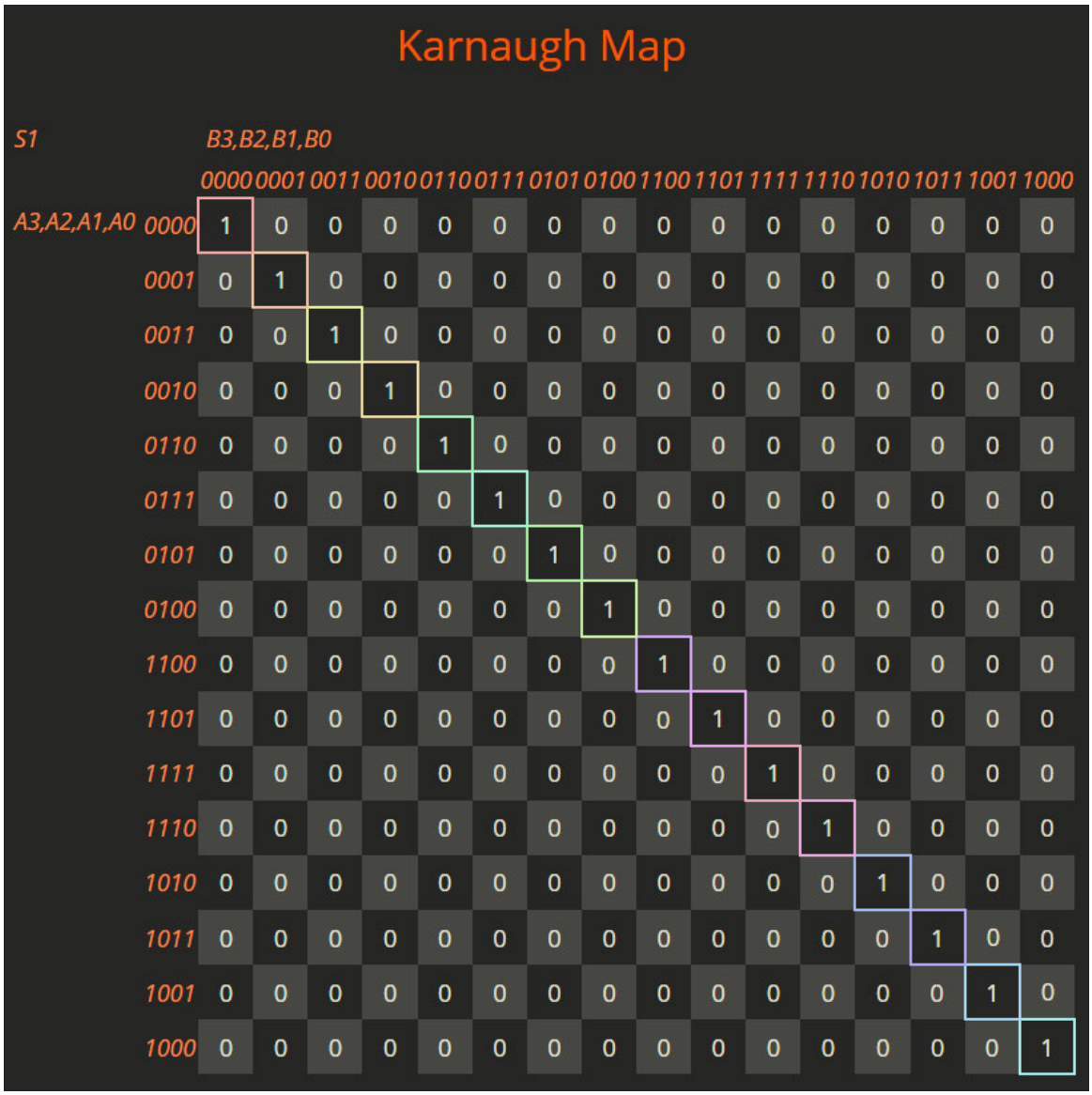
Characteristic Equation: A3'A2'A1'A0'B3'B2'B1'B0' + A3'A2'A1'A0B3'B2'B1'B0 + A3'A2'A1A0'B3'B2'B1B0' + A3'A2'A1A0B3'B2'B1B0 + A3'A2A1'A0'B3'B2B1'B0' + A3'A2A1'A0B3'B2B1'B0 + A3'A2A1A0'B3'B2B1B0' + A3'A2A1A0B3'B2B1B0 + A3A2'A1'A0'B3B2'B1'B0' + A3A2'A1'A0B3B2'B1'B0 + A3A2'A1A0'B3B2'B1B0' + A3A2'A1A0B3B2'B1B0 + A3A2A1'A0'B3B2B1'B0' + A3A2A1'A0B3B2B1'B0 + A3A2A1A0'B3B2B1B0' + A3A2A1A0B3B2B1B0

Operation 10 – A == B



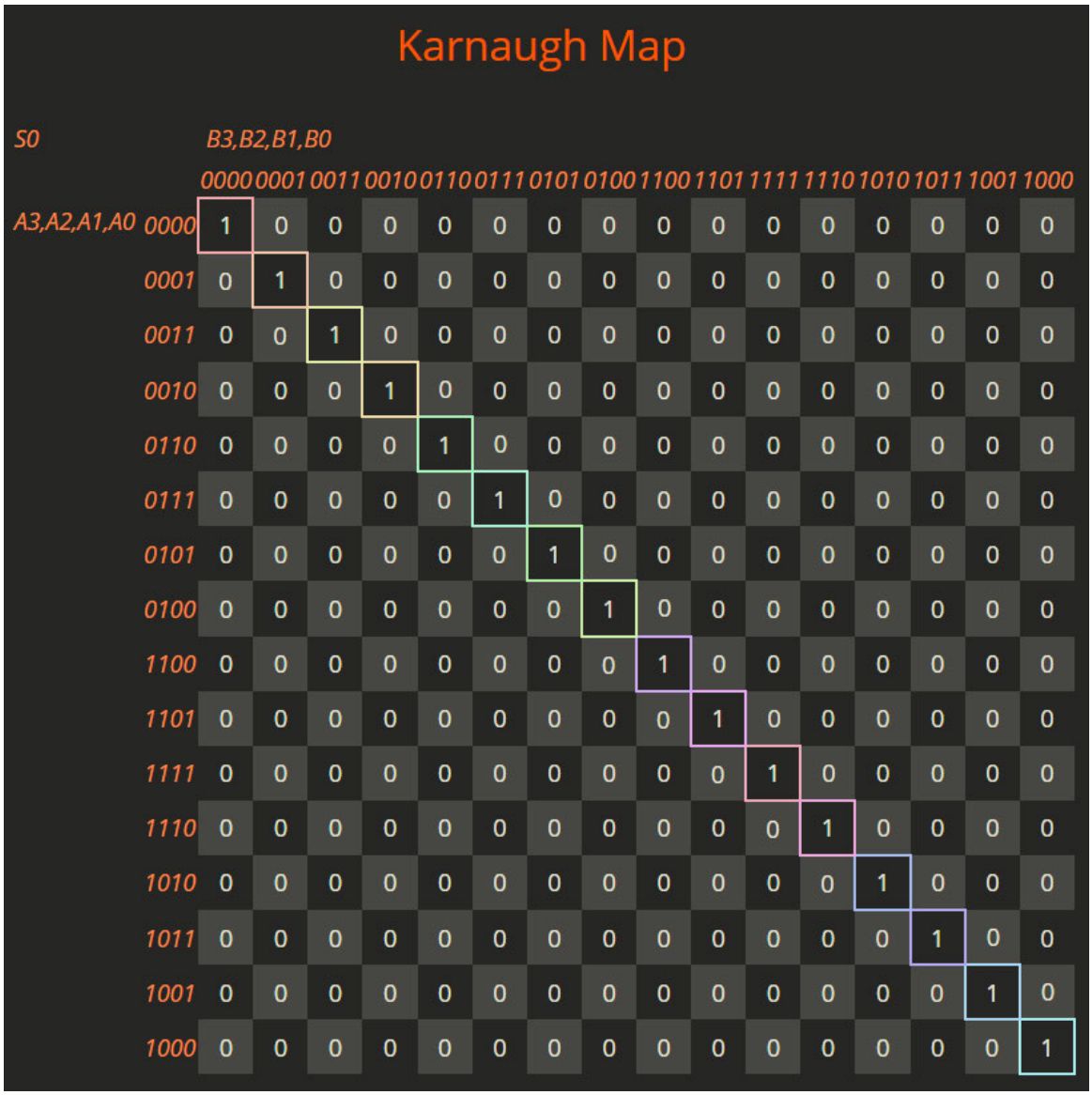
Characteristic Equation: A3'A2'A1'A0'B3'B2'B1'B0' + A3'A2'A1'A0B3'B2'B1'B0 + A3'A2'A1A0'B3'B2'B1B0' + A3'A2'A1A0B3'B2'B1B0 + A3'A2A1'A0'B3'B2B1'B0' + A3'A2A1'A0B3'B2B1'B0 + A3'A2A1A0'B3'B2B1B0' + A3'A2A1A0B3'B2B1B0 + A3A2'A1'A0'B3B2'B1'B0' + A3A2'A1'A0B3B2'B1'B0 + A3A2'A1A0'B3B2'B1B0' + A3A2'A1A0B3B2'B1B0 + A3A2A1'A0'B3B2B1'B0' + A3A2A1'A0B3B2B1'B0 + A3A2A1A0'B3B2B1B0' + A3A2A1A0B3B2B1B0

Operation 10 – A == B



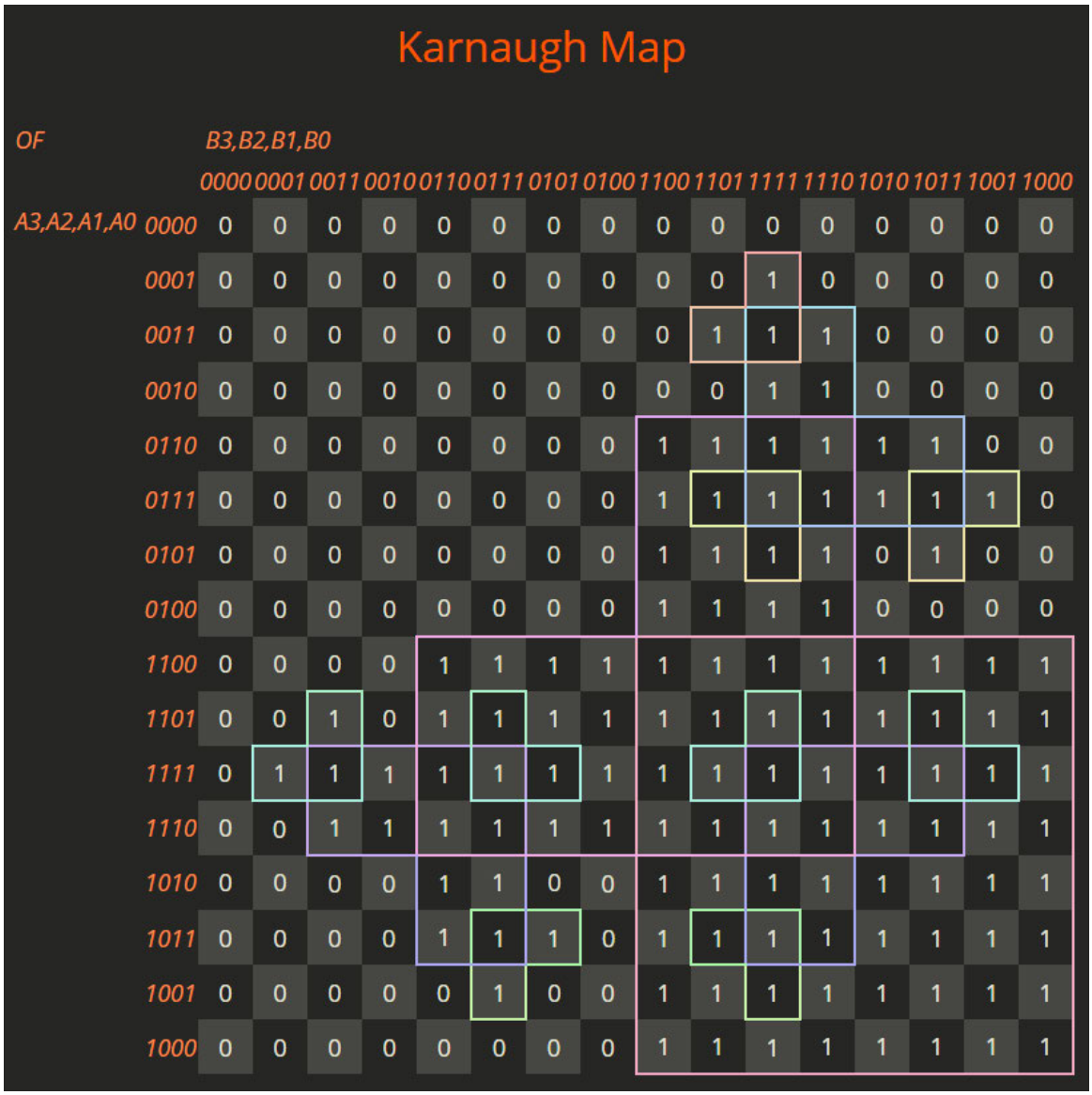
Characteristic Equation: A3'A2'A1'A0'B3'B2'B1'B0' + A3'A2'A1'A0B3'B2'B1'B0 + A3'A2'A1A0'B3'B2'B1B0' + A3'A2'A1A0B3'B2'B1B0 + A3'A2A1'A0'B3'B2B1'B0' + A3'A2A1'A0B3'B2B1'B0 + A3'A2A1A0'B3'B2B1B0' + A3'A2A1A0B3'B2B1B0 + A3A2'A1'A0'B3B2'B1'B0' + A3A2'A1'A0B3B2'B1'B0 + A3A2'A1A0'B3B2'B1B0' + A3A2'A1A0B3B2'B1B0 + A3A2A1'A0'B3B2B1'B0' + A3A2A1'A0B3B2B1'B0 + A3A2A1A0'B3B2B1B0' + A3A2A1A0B3B2B1B0

Operation 10 – A == B



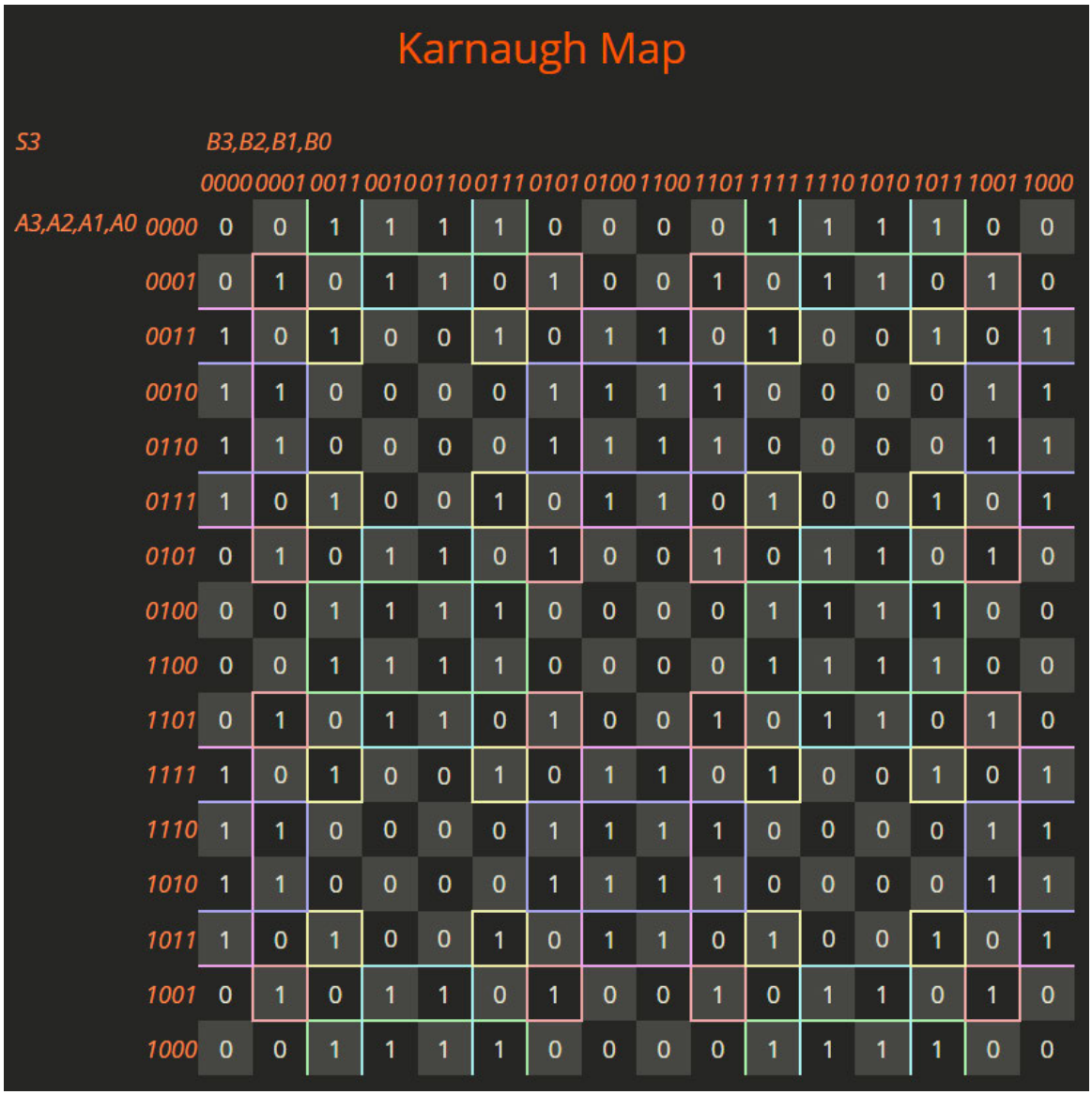
Characteristic Equation: A3'A2'A1'A0'B3'B2'B1'B0' + A3'A2'A1'A0B3'B2'B1'B0 + A3'A2'A1A0'B3'B2'B1B0' + A3'A2'A1A0B3'B2'B1B0 + A3'A2A1'A0'B3'B2B1'B0' + A3'A2A1'A0B3'B2B1'B0 + A3'A2A1A0'B3'B2B1B0' + A3'A2A1A0B3'B2B1B0 + A3A2'A1'A0'B3B2'B1'B0' + A3A2'A1'A0B3B2'B1'B0 + A3A2'A1A0'B3B2'B1B0' + A3A2'A1A0B3B2'B1B0 + A3A2A1'A0'B3B2B1'B0' + A3A2A1'A0B3B2B1'B0 + A3A2A1A0'B3B2B1B0' + A3A2A1A0B3B2B1B0

Operation 11 – A + B



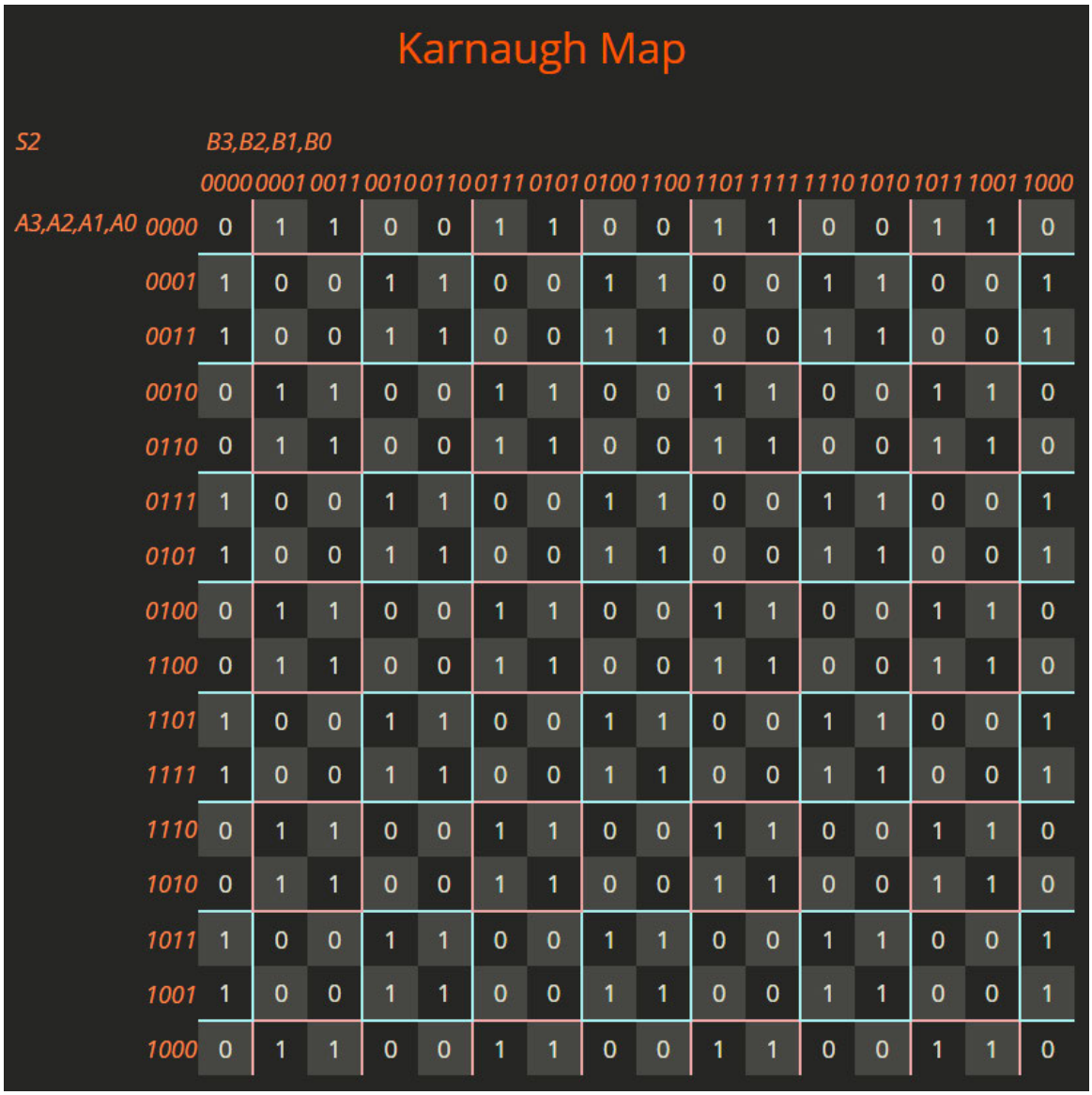
Characteristic Equation: A0B3B2B1B0 + A1B3B2B1 + A1A0B3B2B0 + A2B3B2 + A2A0B3B1B0 + A2A1B3B1 + A2A1A0B3B0 + A3B3 + A3A0B2B1B0 + A3A1B2B1 + A3A1A0B2B0 + A3A2B2 + A3A2A0B1B0 + A3A2A1B1 + A3A2A1A0B0

Operation 11 – A + B



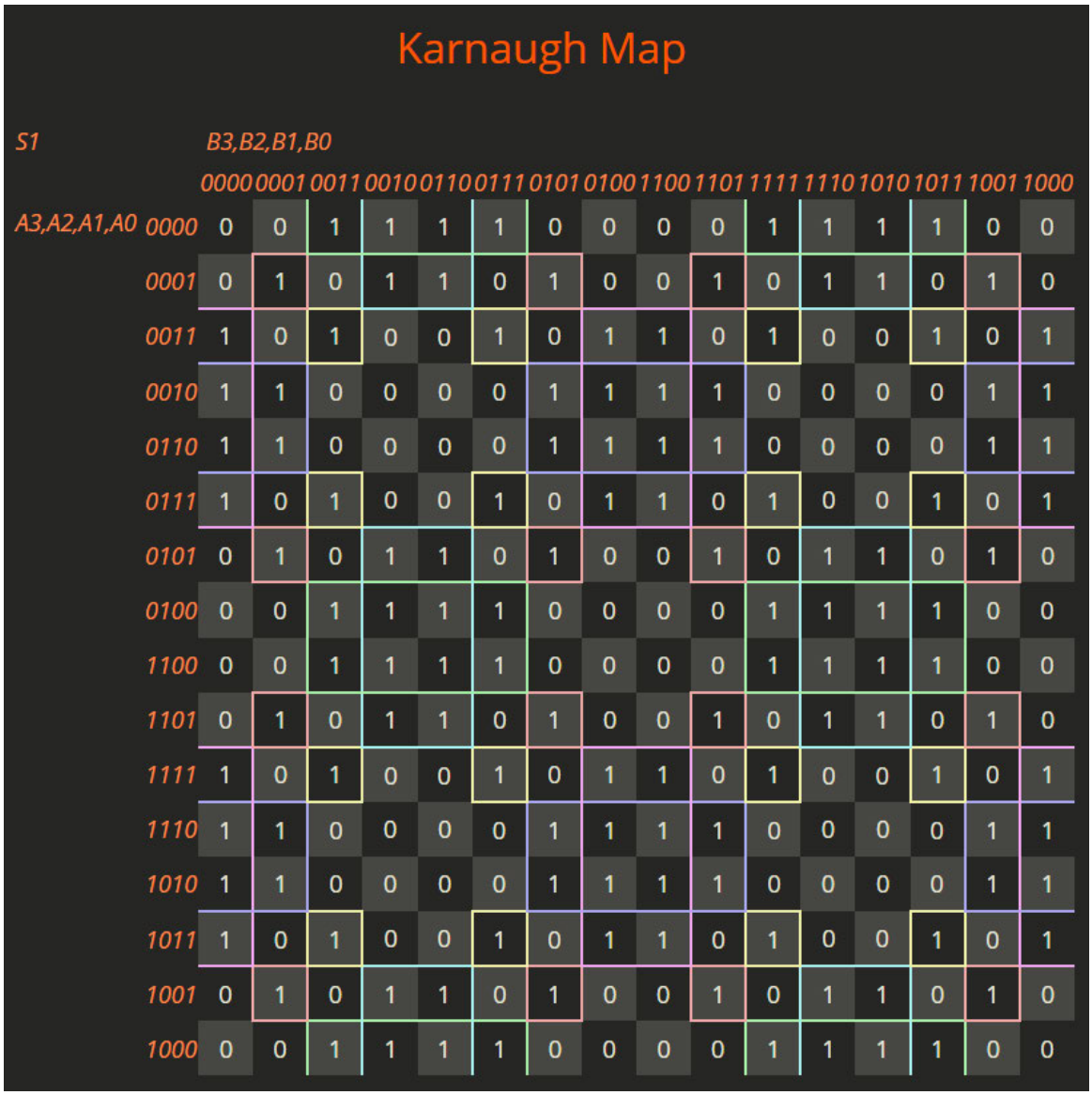
Characteristic Equation: A1'A0'B1 + A1'A0B1'B0 + A1'B1B0' + A1A0'B1' + A1B1'B0' + A1A0B1B0

Operation 11 – A + B



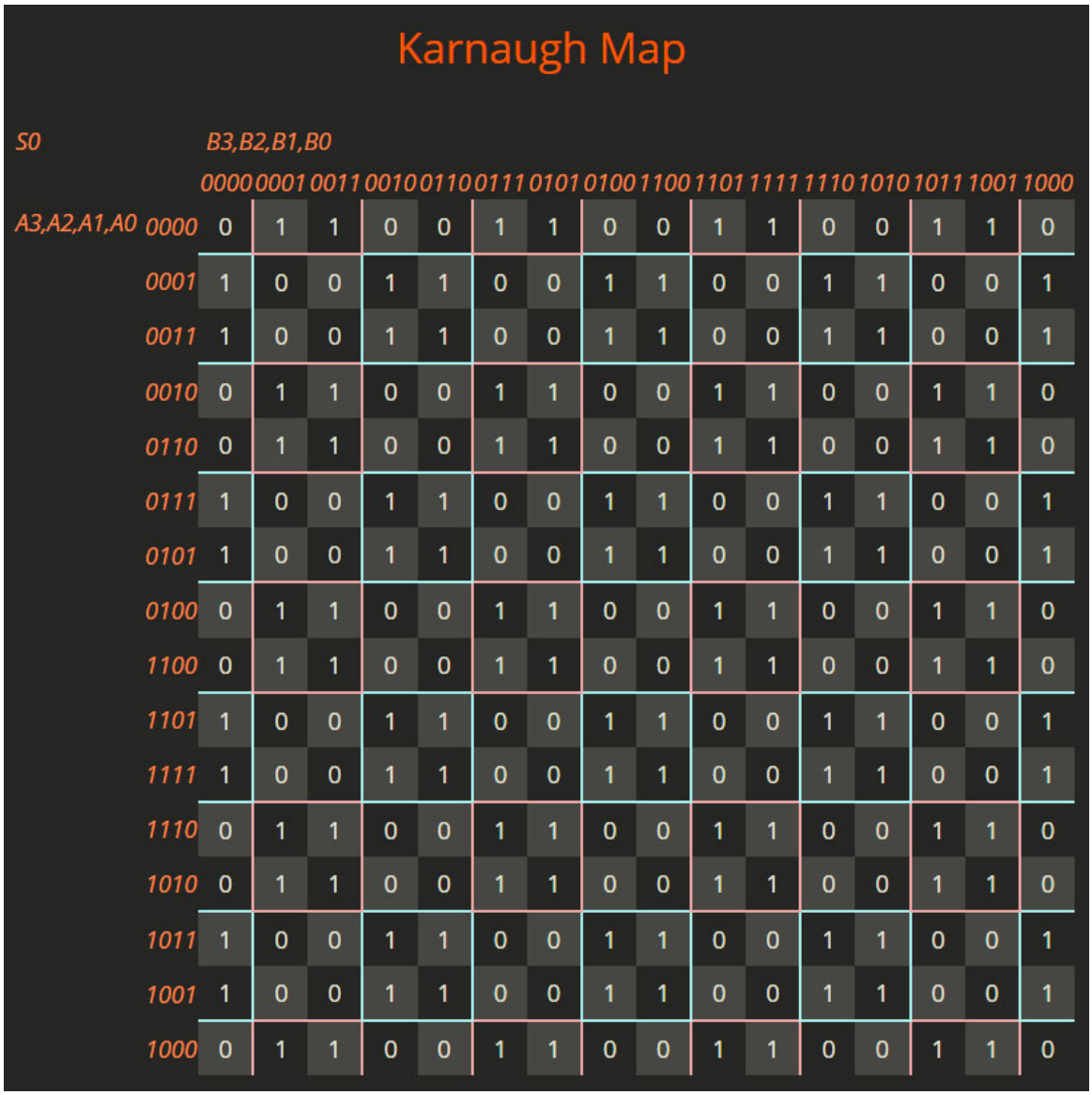
Characteristic Equation: A0'B0 + A0B0'

Operation 11 – A + B



Characteristic Equation: A1'A0'B1 + A1'A0B1'B0 + A1'B1B0' + A1A0'B1' + A1B1'B0' + A1A0B1B0

Operation 11 – A + B



Characteristic Equation: A0'B0 + A0B0'