Topic: Logic Gates and Construction of Arithmetic and Combinational Circuits Based on Them

Objective of the Session:

- Study the basic logic gates (AND, OR, NOT, NAND, NOR, XOR)
- Learn how to design combinational circuits from a given Boolean function
- Implement simple arithmetic units (half adder, full adder) using logic gates
- Master methods for minimizing Boolean functions and apply them in circuit design

---

Theoretical Background (Brief)

Basic Logic Gates

GATE	**SYMBOL**	**TRUTH TABLE**	**BOOLEAN FUNCTION**
NOT (Inverter)	¬A or A	A	Ys 0 → 1 1 → 0
AND (Cunjunction)	A·B or A ∧ B	A B	Y 0 0 → 0 0 1 → 0 1 0 → 0 1 1 → 1
OR (Disjunction)	A+B or A ∨ B	A B	Y 0 0 → 0 0 1 → 1 1 0 → 1 1 1 → 1
NAND (Shaeffer stroke)	A ↑ B	Invertion AND	Y=A·B
NOR (Pierce Arrow)	A ↓ B	Invertion of OR	Y=A+B
XOR (Exclusive OR)	A ⊕ B	A B	Y 0 0 → 0

			0 1 → 1 1 0 → 1 1 1 → 0

Combinational Circuits

A combinational circuit is a digital circuit whose output depends only on the current inputs (it has no memory). Examples: multiplexers, decoders, adders.

Arithmetic Circuits

Half Adder: Adds two bits and produces a Sum (S) and a Carry (C).

Full Adder: Adds three bits (A, B, Cin) and produces Sum (S) and Carry-out (Cout).

Practical Assignments

Assignment 1. Building Logic Circuits from a Truth Table

Given the truth table for function :

A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Tasks:

Write the Canonical Sum of Products (SOP) or Perfect Disjunctive Normal Form (PDNF).

Minimize the function using a Karnaugh map (K-map).

Draw the circuit using AND, OR, NOT gates.

Implement the same function using only NAND gates.

Assignment 2. Half Adder Implementation

Tasks:

Construct the truth table for a half adder.

Write Boolean expressions for Sum (S) and Carry (C).

Build the circuit in a simulator (Logisim, Digital, or on paper).

Verify the circuit operation for all input combinations.

Assignment 3. Full Adder Construction

Tasks:

Build a full adder using two half adders and one OR gate.

Test the circuit for the following inputs:

$$A = 1,\ B = 1,\ C_{in} = 1 \rightarrow S = ?,\ C_{out} = ?$$

$$A = 0,\ B = 1,\ C_{in} = 1 \rightarrow S = ?,\ C_{out} = ?$$

Write the corresponding equations and verify their correctness.

Assignment 4. Designing a 2-bit Adder

Tasks:

Construct a circuit that adds two 2-bit numbers: $A_1A_0 + B_1B_0$

Use one half adder for the least significant bit (LSB) and one full adder for the most significant bit (MSB).

Indicate the number of outputs (3 bits:$S_0,\ S_1,\ C_{out}$).

Test with the example: $11_2 + 10_2 = ?$

$$Answer: 11_2\,(3) + 10_2\,(2) = 101_2\,(5) \rightarrow S_0 = 1,\ S_1 = 0, C_{out} = 1$$

Bonus Assignment (Advanced)

Implement a 2-to-4 line decoder using logic gates.

Inputs: A, B

Outputs: Y_0, Y_1, Y_2, Y_3, where only one output $Y_i = 1$ corresponding to the binary input code.

Example: A=1, B=0 → Y2=1, all others = 0.

Self-Assessment Questions

Why is the NAND gate considered universal?

What is the difference between combinational and sequential circuits?

Can an XOR gate be built only from NAND gates? If yes—how?

What is the role of the carry signal in arithmetic circuits?

Recommended Tools

Logisim Evolution (free digital logic simulator)

Digital (simple online simulator: https://github.com/hneemann/Digital)

Multisim, Proteus (for advanced users)

Lab Report Requirements

Each student must submit:

Solutions to all assignments (including truth tables and circuit diagrams).

Screenshots or schematics built in a simulator.

Answers to the self-assessment questions.

Conclusions: What was learned, what difficulties arose, and how they were resolved.

References

M. M. Malyugin. Digital Devices and Circuit Design.

R. L. Tokheim. Digital Electronics: Principles and Applications.

Logisim Documentation: https://logisim.sourceforge.net/

R. L. Tokheim. Digital Electronics: Principles and Applications.

Документация Logisim: https://logisim.sourceforge.net/