Computer Engineering - Logic Gate Familiarization

- **Objective**: Understand basic logic gate functions using ICs.
- **Materials**:
    - DC Power Supply
    - Digital Multimeter
    - Breadboard
    - 7404, 7408, 7432 ICs
    - LED, 470 ohm resistor
- **Procedure**:

1. Connect +5V to pin 14 and GND to pin 7 of each IC.
2. Test NOT, AND, OR gates individually.
3. Use switches or jumper wires as inputs.
4. Observe LED output and verify with truth tables.

Computer Engineering- De Morgan's Theorem

- **Objective**: Validate De Morgan's Theorem through practical circuits.
- **Materials**:
    - Same as above plus 7400, 7402 ICs
- **Procedure**:

1. Construct NAND/NOR logic equivalent circuits.
2. Compare outputs to original expressions.
3. Use LEDs to verify behavior.
4. Confirm results match De Morgan's laws.

Computer Engineering- Boolean Algebra with Minterms and Maxterms

- **Objective**: Implement and simplify Boolean expressions.

- **Materials**:
    - Same ICs and components as above
- **Procedure**:

1. Create circuits based on minterms/maxterms.
2. Simplify Boolean expressions.
3. Rebuild simplified circuits on breadboard.
4. Compare both outputs using LEDs.

Computer Engineering– Combinational Logic Circuits

- **Objective**: Build and test basic combinational logic circuits.
- **Materials**:
    - Same as above
- **Procedure**:

1. Design a half-adder or multiplexer.
2. Connect gates as per logic diagram.
3. Apply input combinations.
4. Test and verify with LED outputs.

Computer Engineering– Arithmetic Circuits (Adders/Subtractors)

- **Objective**: Design binary adders and subtractors using logic gates.
- **Materials**:
    - Add 7486 IC to above list
- **Procedure**:

1. Build half and full adder/subtractor.
2. Use XOR gates for sum/difference.
3. Test with binary inputs.

4. Use LEDs for results and carry/borrow indication.

Computer Engineering- Sequential Logic Circuits

- **Objective**: Understand flip-flop operations.
- **Materials**:
    - 7400, 7410, 7474, 7476, 7404 ICs
    - 2 LEDs, 2 × 470 ohm resistors
- **Procedure**:

1. Connect SR, D, and JK flip-flops.
2. Apply pulses using push buttons.
3. Observe LED responses.
4. Compare output to timing diagrams.

Computer Engineering- 7-Segment Display

- **Objective**: Display numbers using BCD to 7-segment decoders.
- **Materials**:
    - 7447 decoder, 7-segment display, same basic logic ICs
- **Procedure**:

1. Connect BCD inputs to 7447.
2. Wire segments via resistors.
3. Input binary values.
4. Display digits 0–9.

Computer Engineering– 24 Shot Clock

- **Objective**: Build a countdown timer with display and buzzer.
- **Materials**:

    - 555 Timer, 74192 (x2), 7447 (x2), 7-segment displays, 4093, 7404, 7408, capacitors, buzzer, potentiometer, solder tools

- **Procedure**:

1. Build 555-based timer circuit.
2. Connect counters and decoders to displays.
3. Implement count/reset logic.
4. Trigger buzzer at 24s mark.

Computer Engineering– Introduction to Arduino

- **Objective**: Learn basic Arduino programming.
- **Materials**:
    - Arduino UNO, LED, resistors, switches
- **Procedure**:

1. Program LED blink using delay(), digitalWrite().
2. Use Serial.println() for input feedback.
3. Test simple input/output behavior.

Computer Engineering– LCD with Arduino

- **Objective**: Display messages on LCD via I2C.
- **Materials**:
    - 16x2/20x4 LCD, Potentiometer, I2C module
- **Procedure**:

1. Connect LCD using I2C pins.
2. Upload sketch using LiquidCrystal_I2C library.
3. Adjust contrast and test messages.

Computer Engineering– 4x4 Keypad Matrix

- **Objective**: Interface a matrix keypad with Arduino.
- **Materials**:
    - 4x4 keypad, resistors
- **Procedure**:

1. Connect rows/columns to digital pins.
2. Use Keypad library to read input.
3. Display pressed key in Serial Monitor.
4. Debounce in software.

Computer Engineering– Basic Calculator

- **Objective**: Implement a simple calculator.
- **Materials**:
    - Keypad, LCD
- **Procedure**:

1. Wire keypad and LCD.
2. Program addition, subtraction, multiplication, division.
3. Display results on LCD.

Computer Engineering– Analog Sensors

- **Objective**: Read environmental data using sensors.
- **Materials**:
    - DHT11, photoresistor, LCD
- **Procedure**:

1. Wire sensors to analog/digital pins.
2. Use Arduino to read and display values.

3. Test under different light/temp conditions.

Computer Engineering– Motors and Motion

- **Objective**: Control DC and servo motors.
- **Materials**:
    - Servo, DC motor, transistor (2N222A)
- **Procedure**:

1. Build transistor driver circuit for DC motor.
2. Use PWM to control speed.
3. Use servo.write() for position control.

Information Technology– Removable Media, RAM & Network Interfaces

- **Objective**: Explore hardware interfaces and configurations.
- **Materials**:
    - USB Flash Drive, RJ45, SD Card, RAM, Crimp Tool (optional)
- **Procedure**:

1. Format and test USB/SD storage.
2. Wire RJ45 or identify pinouts.
3. Apply thermal paste for cooling discussion.
4. Reinsert/test RAM, identify specs.

Information Technology- Basic Programming

Objective: Write a calculator program in Python or C.
Materials:

- Computer
- IDE (VS Code / Turbo C / Jupyter Notebook)

Procedure:

1. Define operations (+, -, *, /).
2. Take user input.
3. Use conditionals to process input.
4. Display result with error handling.

Information Technology- Flowchart and Pseudocode Lab

Objective: Create logic flow using diagram and pseudocode.
Materials:

- Whiteboard/marker or flowchart software
- Paper and pencil

Procedure:

1. Pick a process (e.g., login system).
2. Draw flowchart.
3. Write corresponding pseudocode.
4. Optionally convert to working code.

Information Technology- SQL Database Lab

Objective: Create and manage relational database.
Materials:

- MySQL/SQLite
- SQL Workbench or phpMyAdmin

Procedure:

1. Create tables (e.g., Students, Courses).
2. Insert sample records.
3. Query using SELECT, JOIN, and GROUP BY.
4. Display output and interpret.

Information Technology - HTML & CSS Basics

Objective: Design a personal webpage.
Materials:

- Computer with browser
- Text editor (VS Code or Sublime)

Procedure:

1. Write HTML structure (header, body, footer).
2. Style using CSS (colors, layout).
3. Add content and images.

4. Open in browser and debug.

Information Technology- Networking Protocol Lab

Objective: Capture and analyze network packets.
Materials:

- Wireshark
- 2 PCs connected via LAN
- Ethernet cable

Procedure:

1. Setup LAN between PCs.
2. Start packet capture.
3. Transfer data or ping.
4. Analyze captured packets.

Information Technology -Data Structures Lab

Objective: Implement stack and queue in code.
Materials:

- Code editor (C/Python/Java)
- Compiler

Procedure:

1. Define stack and queue structure.
2. Perform insert/remove operations.
3. Handle overflow/underflow.

Test with sample data.

Information Technology- AI Chatbot Development

Objective: Build and deploy a chatbot using AI.
Materials:

- Python
- Rasa or Hugging Face API
- Flask or Django

Procedure:

1. Define intents and responses.
2. Train NLP model.
3. Integrate with web interface.
4. Test chatbot functionality.

Information Technology -Full-Stack Web App

Objective: Develop a CRUD-based web application.
Materials:

- React.js (Frontend)
- Node.js/Express (Backend)
- MongoDB/PostgreSQL

Procedure:

1. Design UI and backend routes.
2. Implement data operations.
3. Test APIs and UI flow.
4. Deploy and document.

Architecture- Basic Model Making

Objective: Understand spatial design by creating scaled architectural models.
Materials:

- Foam board / cardboard
- Glue (Fevicol or glue gun)
- Cutter, scissors
- Metal ruler
- Pencil, sketch paper

Procedure:

1. Design a basic floor layout.
2. Scale down to 1:50.
3. Cut out wall and roof shapes.
4. Assemble using glue.
5. Label building elements.

Architecture- Orthographic to Isometric Drawing

Objective: Learn conversion from 2D orthographic to 3D isometric view.
Materials:

- Drawing sheets
- T-square and set squares
- Pencils (HB and 2H)
- Compass, scale ruler

Procedure:

1. Draw top, front, and side views.
2. Use vanishing lines to convert to isometric.
3. Shade and label.
4. Cross-check with original dimensions.

Architecture- Light & Shadow Study

Objective: Study light behavior on building forms.
Materials:

- Physical scale model
- Torchlight or access to direct sunlight
- White sheet background
- Smartphone or DSLR camera

Procedure:

1. Direct light at the model at various angles.
2. Capture shadow patterns.
3. Analyze for window/roof positioning.

Architecture -Perspective Drawing Exercise

Objective: Develop 2-point perspective rendering skills.
Materials:

- Drawing sheet
- Vanishing point guide
- Ruler and pencils

Procedure:

1. Establish horizon line.
2. Add vanishing points.
3. Sketch forms using guide lines.
4. Add shading and details.

Architecture- Sustainable Material Testing

Objective: Compare insulation properties of materials.
Materials:

- Clay bricks, strawboard, hollow block
- Heat lamp
- Infrared thermometer

Procedure:

1. Heat each material sample.
2. Record temperature every 5 minutes.
3. Plot data to compare heat retention.
4. Recommend eco-material.

Architecture- 2D/3D CAD Modeling

Objective: Create architectural designs using CAD tools.
Materials:

- AutoCAD or Revit
- PC/Laptop

Procedure:

1. Draft a 2D floor plan.
2. Extrude walls for 3D view.
3. Add doors, windows, roof.
4. Apply materials and render.

Architecture- Urban Design Project

Objective: Design urban block with sustainability in mind.
Materials:

- GIS tools or AutoCAD
- Satellite imagery
- Site survey data

Procedure:

1. Analyze site data.
2. Plan zoning, traffic, green areas.
3. Model in CAD/GIS.
4. Create final drawings and views.

Architecture-Green Building Case Study

Objective: Analyze features of LEED or GRIHA-certified buildings.
Materials:

- Building documentation
- Internet resources
- Presentation tools

Procedure:

1. Research green strategies used.
2. Compare energy savings and passive features.
3. Draw system diagrams.
4. Present findings.

Industrial Engineering- Work Measurement Lab

Objective: Record and analyze time for simple tasks.
Materials:

- Stopwatch
- LEGO blocks or simple toolkit
- Observation and calculation sheet

Procedure:

1. Define a repetitive task.
2. Record 10 repetitions with stopwatch.
3. Calculate average, standard, and normal time.
4. Recommend method improvements.

Industrial Engineering - Ergonomic Posture Assessment

Objective: Assess workstation posture and ergonomics.
Materials:

- Standard desk and chair
- RULA (Rapid Upper Limb Assessment) worksheet
- Measuring tape

Procedure:

1. Observe posture while performing tasks.
2. Fill in RULA scoring.
3. Suggest improvements.
4. Re-evaluate post-adjustment.

Industrial Engineering- Plant Layout Design

Objective: Improve layout using systematic planning.

Materials:

- Graph paper or layout software (AutoCAD)
- Sample process charts
- Color pens or digital tools

Procedure:

1. Draw existing layout.
2. Analyze material flow and bottlenecks.
3. Apply SLP principles.
4. Present revised layout.

Industrial Engineering - Time Study with App

Objective: Use software for digital time analysis.
Materials:

- Time-study mobile app
- Sample repetitive task
- Spreadsheet

Procedure:

1. Record cycle time with app.
2. Analyze data in spreadsheet.
3. Plot histograms, control charts.
4. Interpret for process stability.

Industrial Engineering- FlexSim Simulation

Objective: Simulate manufacturing process flow.
Materials:

- FlexSim software
- Sample process data

Procedure:

1. Model machines, workers, products.
2. Assign times and connections.
3. Run simulation.
4. Identify bottlenecks.

Industrial Engineering- Inventory Management Lab

Objective: Apply EOQ and reorder models.
Materials:

- Spreadsheet software (Excel, Google Sheets)

Procedure:

1. Input demand and cost variables.
2. Use EOQ formula to find optimal order size.
3. Simulate different demand levels.

Visualize with charts.

Industrial Engineering- Six Sigma DMAIC Project

Objective: Improve a process using DMAIC.
Materials:

- Minitab or Excel
- Process logs

Procedure:

1. Define problem (e.g., delay or defect).
2. Measure current performance.
3. Analyze root causes.
4. Improve and control process.

Industrial Engineering- Human Factors & Ergonomics

Objective: Design a user-friendly workstation.
Materials:

- Anthropometric database
- Sketch paper or CAD software

Procedure:

1. Collect body measurements.
2. Design workstation accordingly.
3. Create sketch or CAD model.
4. Justify design ergonomically.

5.5.1 Packet Tracer - IPv4 ACL Implementation Challenge

Objectives

- Configure a router with standard named ACLs.
- Configure a router with extended named ACLs.
- Configure a router with extended ACLs to meet specific communication requirements.
- Configure an ACL to control access to network device terminal lines.
- Configure the appropriate router interfaces with ACLs in the appropriate direction.
- Verify the operation of the configured ACLs.

Background / Scenario

In this activity you will configure extended, standard named, and extended named ACLs to meet specified communication requirements.

Instructions

Verify Connectivity in the New Company Network
First, test connectivity on the network as it is before configuring the ACLs. All hosts should be able to ping all other hosts.

Configure Standard and Extended ACLs per Requirements

ACL 1 Requirements

- Create ACL 101.
- Explicitly block FTP access to the Enterprise Web Server from the internet.
- No ICMP traffic from the internet should be allowed to any hosts on HQ LAN 1.
- Allow all other traffic.

ACL 2 Requirements

- Use ACL number 111.
- No hosts on HQ LAN 1 should be able to access the Branch Server.
- All other traffic should be permitted.

ACL 3 Requirements

- Create a named standard ACL. Use the name vty_block.
- Only addresses from the HQ LAN 2 network should be able to access the VTY lines of the HQ router.

ACL 4 Requirements

- Create a named extended ACL called branch_to_hq.
- No hosts on either of the Branch LANs should be allowed to access HQ LAN 1.
- Use one access list statement for each of the Branch LANs.
- All other traffic should be allowed.

Verify ACL Operation

- Perform the connectivity tests between devices in the topology.
- Use show ip access-lists and clear access list counters to verify ACLs.

6.2.7 Packet Tracer - Investigate NAT Operation

Objectives

- Part 1: Investigate NAT Operation Across the Intranet
- Part 2: Investigate NAT Operation Across the Internet
- Part 3: Conduct Further Investigations

Scenario

As a frame travels across a network, the MAC addresses may change. IP addresses can also change when a packet is forwarded by a device configured with NAT. In this activity, we will investigate what happens to IP addresses during the NAT process.

Instructions

Investigate NAT Operation Across the Intranet

1. Wait for the network to converge.
2. Generate an HTTP request from any PC in the Central domain.
3. Switch to Simulation mode and filter to show only HTTP.
4. Open the browser of a Central PC, navigate to http://branchserver.pka.
5. Observe and record IP addresses and port changes as the packet moves through D1/D2, R2, R4, and reaches the Branch Server.
6. On R2 and R4, use show ip nat translations.

Investigate NAT Operation Across the Internet

1. Generate an HTTP request from any PC in the Home Office.
2. Navigate to http://centralserver.pka.
3. Observe IP address changes as the packet passes WRS and R2.
4. Use show ip nat translations on R2.
5. Return to Realtime mode.

Conduct Further Investigations

1. Try more HTTP and HTTPS packets.
2. Observe if NAT translation tables grow.

3. Determine if WRS uses a NAT pool.
4. Reflect on how NAT supports internet connectivity.
5. Use NAT table information to identify inside global/local IPs.
6. Identify which devices perform NAT and note their common traits.

5.1.9 Packet Tracer - Configure Named Standard IPv4 ACLs

Objectives

- Part 1: Configure and Apply a Named Standard ACL
- Part 2: Verify the ACL Implementation

Background / Scenario

The senior network administrator has asked you to create a standard named ACL to prevent access to a file server. The file server contains the database for the web applications. Only the Web Manager workstation **PC1** and the **Web Server** need to access the File Server. All other traffic to the File Server should be denied.

Instructions

Configure and Apply a Named Standard ACL

1. Verify connectivity before the ACL is configured and applied.
2. All three workstations should be able to ping both the Web Server and File Server.
3. Configure a named standard ACL:

R1(config)# ip access-list standard File_Server_Restrictions

R1(config-std-nacl)# permit host 192.168.20.4

R1(config-std-nacl)# permit host 192.168.100.100

R1(config-std-nacl)# deny any

Note: The ACL name is case-sensitive and the statements must be in the order shown.

4. Use show access-lists to verify the ACL before applying:

R1# show access-lists

5. Apply the ACL outbound on the **Fast Ethernet 0/1** interface:

R1(config-if)# ip access-group File_Server_Restrictions out

6. Save the configuration.

Verify the ACL Implementation

1. Use show access-lists to verify the ACL configuration.
2. Use show run or show ip interface fastethernet 0/1 to confirm the ACL is applied.

3. Verify functionality:
    - All three workstations should be able to ping the Web Server.
    - Only **PC1** and the Web Server should be able to ping the File Server.
    - Use show access-lists to view packet match counts.

5.2.7 Packet Tracer - Configure and Modify Standard IPv4 ACLs

Objectives

- Part 1: Verify ACL Functionality
- Part 2: Configure and Modify a Standard ACL

Background / Scenario

The senior network administrator has asked you to configure a standard ACL to prevent **PC0** from accessing the **File Server**. After verifying the ACL's functionality, you will modify the ACL to also deny access from **PC1** to the **File Server** while allowing other traffic.

Instructions

Verify ACL Functionality

1. Before configuring the ACL, verify that all PCs can ping both the **Web Server** and **File Server**.

Configure and Modify a Standard ACL

1. Configure a numbered standard ACL on **R1**.

R1(config)# access-list 10 deny host 192.168.20.3

R1(config)# access-list 10 permit any

2. Apply the ACL outbound on the **Fast Ethernet 0/1** interface:

R1(config-if)# ip access-group 10 out

3. Use show access-lists to verify the ACL.
4. Confirm that **PC0** cannot ping the **File Server** but can ping the **Web Server**.
5. Modify the ACL to also deny **PC1**:

R1(config)# no access-list 10

R1(config)# access-list 10 deny host 192.168.20.3

R1(config)# access-list 10 deny host 192.168.20.4

R1(config)# access-list 10 permit any

6. Verify the new ACL with show access-lists.

7. Confirm that **PC0** and **PC1** cannot ping the **File Server** but can still ping the **Web Server**.

7.6.1 Packet Tracer - WAN Concepts

S

- Part 1: Build a Simple Topology
- Part 2: Configure Basic Device Settings
- Part 3: Verify the WAN Topology

Background / Scenario

In this activity, you will build a simple topology using a **serial WAN link** between two routers. After completing the physical topology, you'll configure basic settings on both routers and verify the WAN connection by using the show commands and testing connectivity with the ping command.

Instructions

Part 1: Build a Simple Topology

1. Add two **1841 routers** to the workspace.
2. Add a **2960 switch** to the workspace.
3. Add two **PCs**.
4. Connect the devices using appropriate cables:
    - **Router to Router (Serial DCE to DTE)**
    - **Router to Switch (Straight-Through)**
    - **Switch to PCs (Straight-Through)**

Part 2: Configure Basic Device Settings

1. Configure hostnames on both routers.
2. Assign IP addresses to interfaces according to the addressing table.
3. Configure the **Serial 0/0/0** interface on **Router0** with a clock rate:

Router0(config-if)# clock rate 64000

4. Enable interfaces using:

Router(config-if)# no shutdown

Part 3: Verify the WAN Topology

1. Use show ip interface brief to verify interface status.
2. Use show controllers serial 0/0/0 on **Router0** to verify the DCE cable connection.
3. Use the ping command from one router to the other to test WAN connectivity.
4. Use the ping command from one PC to the other to verify end-to-end connectivity.