EXPERIMENT 1: BASIC LOGIC GATES

1) *Objectives*

The primary objective of Experiment 1 is to verify the truth tables of basic logic gates.

2) *Required Components*

- Gates for Part 1
  - 7404 Hex Inverters
  - 7408 Quadruple 2-input AND gates
  - 7432 Quadruple 2-input OR gates
- Gates for Part 2
  - 7404 Hex Inverters
  - 7400 Quadruple 2-input NAND gates
  - 7402 Quadruple 2-input NOR gates
  - 7486 Quadruple 2-input XOR gates
  - 74266 Quadruple 2-input XNOR gates
- DIP switches
- LEDs
- Resistors (220Ω, 330Ω)
- 1N4148 diode

3) *Preliminary Work*

a) *Theoretical Calculations*

Prepare the truth tables for the logic gates (NOT, AND, OR, NAND, NOR, XOR, XNOR).

b) *OrCAD Simulation*

Simulate each logic gate (NOT, AND, OR, NAND, NOR, XOR, XNOR) and generate the truth tables in PSpice. Verify the truth tables and add the simulation results to your report. Explain each step and add comments where necessary.

- **Note1:** After simulation, find and click *Cascade* button under *Window* menu. Then shrink the simulation window from the corners with your mouse and show the full waveform (2 x the largest period) as given in the example below.

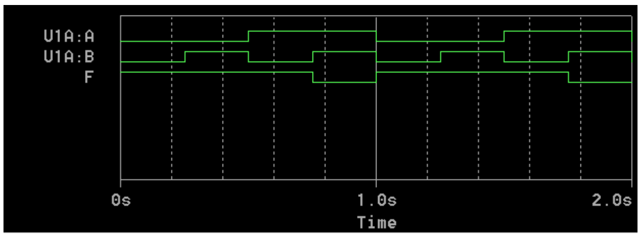
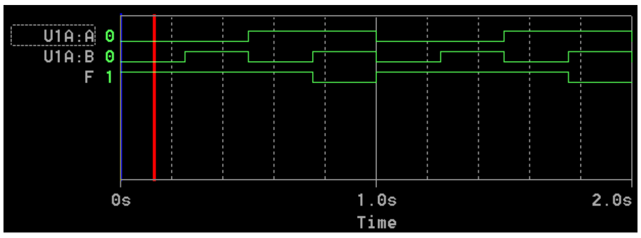


Figure 1.1. OrCAD simulation result

- **Note2:** Click *Toggle cursor* button and show each row of the truth table on your simulation graph as given in the example below. Note the logic 0 and logic 1 next to the variables of A, B, and F.



Ex: The 1st row of the truth table		
A	**B**	**F**
0	**0**	**1**

Figure 1.2. OrCAD simulation with cursor

c) *TinkerCAD Simulation*

Simulate each logic gate (NOT, AND, OR, NAND, NOR, XOR) and generate the truth tables in TinkerCAD. Verify the truth tables and add the simulation results to your report. Explain each step and add comments where necessary.

- **Note:** For each input combination, clearly show all inputs and all outputs in the same picture as shown in the figure below.

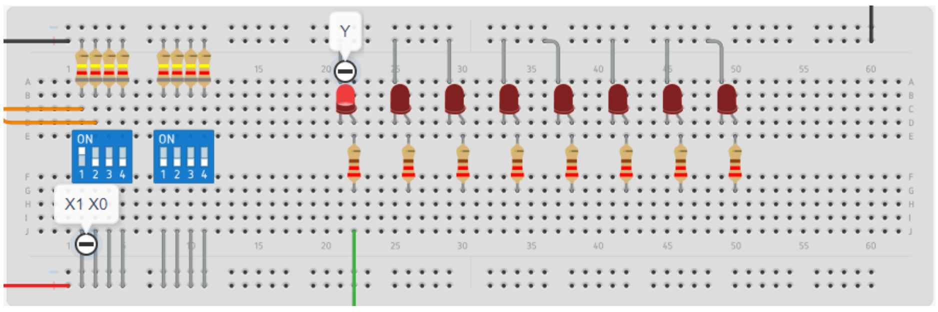


Figure 1.3. OR Gate simulation in TinkerCAD with input X1X2 = 10

4) *Experimental Work*

Part 1 (Experiment 1A)

For NOT, AND, and OR gates follow the steps given below:

- Place the IC on the breadboard.
- Check the datasheet of the IC to locate the power, ground, input, and output pins.
- Connect the power (+5V) and ground (0V) to the appropriate pins.
- Connect the input pins of a gate to the switches and the output to an LED.
- Derive the truth table of the gate by trying all possible input combinations.
- Repeat these steps for all gates on all ICs for this gate. This is to ensure all ICs that the student has obtained works properly.

Part 2 (Experiment 1B)

a) For NAND, NOR, XOR, and XNOR gates follow the steps given below:

- Place the IC on the breadboard.
- Check the datasheet of the IC to locate the power, ground, input, and output pins.
- Connect the power (+5V) and ground (0V) to the appropriate pins.
- Connect the input pins of a gate to the switches and the output to an LED.
- Derive the truth table of the gate by trying all possible input combinations.
- Repeat these steps for all gates on all ICs for this gate. This is to ensure all ICs that the student has obtained works properly.

b) Build the circuit given in Fig. 1.4 on a breadboard. Wire the 7404 to ground and +5V. A diode is connected as seen in Fig.1.4 in order to prevent applying the negative cycle of the $9V_{peak}$ sinusoidal voltage source to the input of NOT gate. Connect X and Y probes of the oscilloscope to the input and output of the NOT gate, respectively.

i) Observe the characteristic of NOT gate in X-Y mode.

ii) Determine V_{IL}, V_{IH}, V_{OL}, and V_{OH} from the characteristic curve on the oscilloscope screen

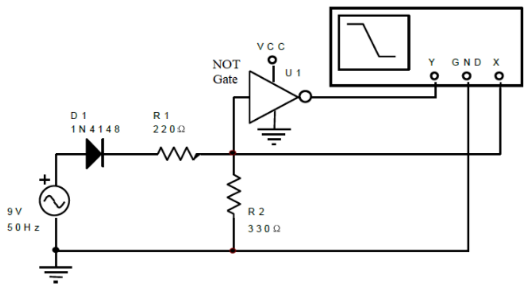


Figure 1.4.

EXPERIMENT 2: COMBINATIONAL CIRCUIT ANALYSIS

1) Objectives

- To understand how to use logic gates to create a simple combinational logic circuit
- To analyze a combinational circuit and derive the Boolean expression
- To develop digital circuit building and troubleshooting skills

2) Required Components

- 7408 Quadruple 2-input AND gates
- 7432 Quadruple 2-input OR gates
- 7400 Quadruple 2-input NAND gates
- DIP switches
- LEDs
- Resistors

3) Preliminary Work

a) Theoretical Calculations

- Calculate the outputs for all possible input combinations (0000-1111) for all outputs (G1,…,G8) and prepare a truth table for the circuit given in Fig. 2.1.
- Using the truth table, derive the Boolean expression for each output.
- Find the simplest possible expression for each output.

b) OrCAD Simulation

- Simulate the circuit given in Fig 2.1 using OrCAD PSpice and add the simulation results to the report. **(Pay attention to the simulation notes given in the first experiment)**
- Fill in the truth table using OrCAD simulation results.

c) TinkerCAD Simulation

- Simulate the circuit given in Fig 2.1 using TinkerCAD and add the simulation results to your report. **(Pay attention to the simulation notes given in the first experiment)**
- Fill in the truth table using TinkerCAD simulation results.

4) Experimental Work

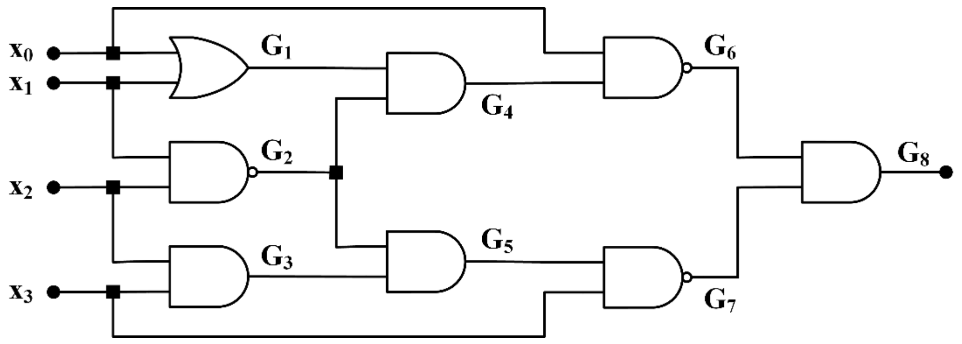


Figure 1.1

- Implement the circuit given in Fig. 2.1 on your breadboard. Connect the inputs to the switches and outputs (G1,…,G8) to LEDs. Make sure all ICs have proper power and ground connections.
- Fill the truth table given in Table 2.1.

Table 2.1

X_3	X_2	X_1	X_0	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8
0	0	0	0								
0	0	0	1								
0	0	1	0								
0	0	1	1								
0	1	0	0								
0	1	0	1								
0	1	1	0								
0	1	1	1								
1	0	0	0								
1	0	0	1								
1	0	1	0								
1	0	1	1								
1	1	0	0								
1	1	0	1								
1	1	1	0								
1	1	1	1								

EXPERIMENT 3: CODE CONVERTERS AND KARNAUGH MAP

1) Objectives

- To understand simplifying logic functions using Karnaugh maps
- To implement gray to binary and BCD to Excess-3 code converters

2) Required Components

- 7404 Hex Inverters
- 7408 Quadruple 2-input AND gates
- 7432 Quadruple 2-input OR gates
- 7486 Quadruple 2-input XOR gates
- DIP switches
- LEDs
- Resistors

3) Preliminary Work

a) Theoretical Calculations

- The truth table for gray to binary code converter is given in Table 3.1. Use Karnaugh maps to get simplified expressions for each binary bit as a function of gray code bits. **Verify the circuit given in Fig. 3.1 with your Boolean expressions.**
- The truth table for BCD to Excess-3 code converter is given in Table 3.2. Use Karnaugh maps to get simplified expressions for each binary bit as a function of gray code bits. **Verify the circuit given in Fig. 3.2 with your Boolean expressions.**

b) OrCAD Simulation

- Simulate the circuit given in Fig. 3.1 using OrCAD PSpice and add the simulation results to the report. **(Pay attention to the simulation notes given in the first experiment)**
- Show that the simulation results satisfy the truth table given in Table3.1.
- Simülate the circuit given in Fig. 3.2. using OrCAD PSpice and add the simulation results to the report. **(Pay attention to the simulation notes given in the first experiment)**
- Show that the simulation results satisfy the truth table given in Table3.2.

c) TinkerCAD Simulation

- Simulate the circuit given in Fig. 3.1 using TinkerCAD and add the simulation results to the report. **(Pay attention to the simulation notes given in the first experiment)**
- Show that the simulation results satisfy the truth table given in Table3.1.
- Simulate the circuit given in Fig. 3.2. using TinkerCAD and add the simulation results to the report. **(Pay attention to the simulation notes given in the first experiment)**
- Show that the simulation results satisfy the truth table given in Table3.2.

4) Experimental Work

i. Implement the circuit given in Fig. 3.1 on your breadboard. Connect the inputs to switches and outputs to LEDs. Make sure that all ICs have proper power and ground connections.

ii. Verify the truth table given in Table 3.1.

iii. Implement the circuit given in Fig. 3.2 on your breadboard. Connect the inputs to switches and outputs to LEDs. Make sure that all ICs have proper power and ground connections.

iv. Verify the truth table given in Table 3.2.

Table 3.1. *Truth Table for Gray-to-Binary Code Converter*

Gray Code				Binary Code			
G3	G2	G1	G0	B3	B2	B1	B0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	1
1	0	0	0	1	1	1	1
1	0	0	1	1	1	1	0
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	1	0	0	1	0	0	0
1	1	0	1	1	0	0	1
1	1	1	0	1	0	1	1
1	1	1	1	1	0	1	0

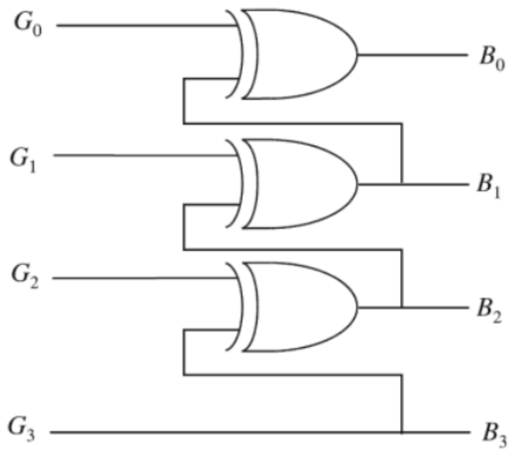


Figure 3.1

Table 3.2. *Truth Table for BCD-to-Excess-3 Code Converter*

BCD Code				Excess–3 Code			
B3	B2	B1	B0	E3	E2	E1	E0
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0

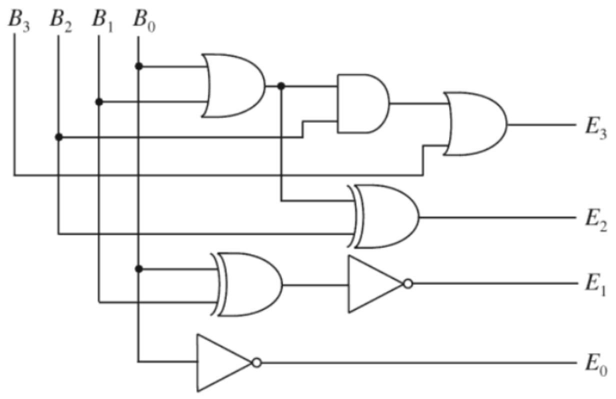


Figure 3.2

EXPERIMENT 4: ADDER, SUBTRACTOR, MULTIPLEXER, AND DEMULTIPLEXER APPLICATIONS

1) Objectives

- To learn full adder applications of multiplexer and demultiplexer
- To learn adder/subtractor circuit design using 74283 IC

2) Required Components

- 7404 Hex Inverters
- 74LS138 Decoder/Demultiplexer
- 74LS153 Multiplexer
- 7486 2-input XOR gate
- 74283 4-bit binary adder
- DIP switches
- LEDs
- Resistors
- Any logic gate used for design in the theoretical part

3) Preliminary Work

a) Theoretical Calculations

- The truth table for 1-bit full adder is given in Table 4.1.
- Design a 1-bit full adder using 4x1 multiplexers. Show Boolean expressions and block diagram of the circuit in your report.
- Design a 1-bit full adder using 3x8 demultiplexer. Show Boolean expressions and block diagram of the circuit in your report.
- Draw the block diagram of the 4-bit full adder/subtractor circuit and explain how it works.
- Calculate the inputs and outputs of the 4-bit adder/subtractor circuit given in Fig.4.1 for the values given in Table 4.2 and add the filled table to your report.

b) OrCAD Simulation

- Simulate the 1-bit full adder (multiplexer) you have designed using OrCAD Pspice. Show that the circuit functions properly. **(Pay attention to the simulation notes given in the first experiment sheet)**
- Simulate the 1-bit full adder (demultiplexer) you have designed using OrCAD Pspice. Show that the circuit functions properly. **(Pay attention to the simulation notes given in the first experiment sheet)**
- Simulate the circuit given in Fig. 4.1 and fill in the table given in Table 4.2. Add simulation results to your report for each row in the table. **(Pay attention to the simulation notes given in the first experiment sheet)**

c) TinkerCAD Simulation

- Simulate the 1-bit full adder (multiplexer) you have designed using TinkerCAD. Show that the circuit functions properly. **(Pay attention to the simulation notes given in the first experiment sheet)**
- Simulate the 1-bit full adder (demultiplexer) you have designed using TinkerCAD. Show that the circuit functions properly. **(Pay attention to the simulation notes given in the first experiment sheet)**
- Simulate the circuit given in Fig. 4.1 and fill in the table given in Table 4.2. Add simulation results to your report for each row in the table. **(Pay attention to the simulation notes given in the first experiment sheet)**

4) Experimental Work

i. Implement the 1-bit full adder circuit you have designed using 74LS153 Multiplexer and verify the truth table given in Table 4.1. Make sure all ICs have proper power and ground connections. Connect inputs to switches and outputs to LEDs.

ii. Implement the 1-bit full adder circuit you have designed using 74LS138 Demultiplexer and verify the truth table given in Table 4.1 Make sure all ICs have proper power and ground connections. Connect inputs to switches and outputs to LEDs.

iii. Implement the circuit given in Fig. 4.1. on your breadboard. Make sure all ICs have proper power and ground connections. Connect inputs to switches and outputs to LEDs. Fill the table given in Table 4.2.

Table 4.1. Truth Table of 1-bit Full Adder

INPUTS			OUTPUTS	
A	B	C_{in}	S	C
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

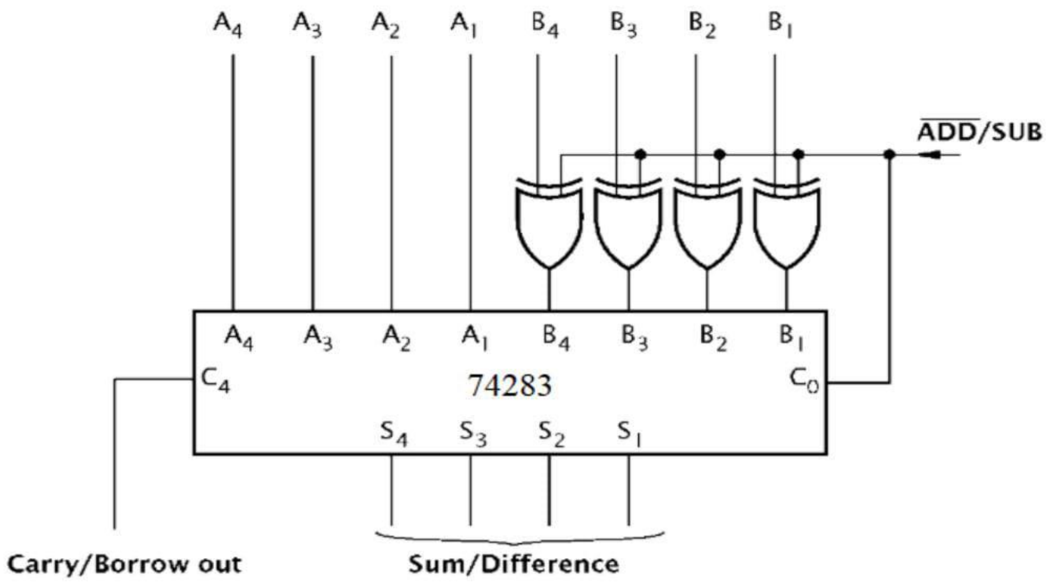


Figure 4.1. Adder/Subtractor Circuit

Table 4.2

$\underline{A}/S$	A	B	$A_3A_2A_1A_0$	$B_3B_2B_1B_0$	C_4	$S_3S_2S_1S_0$
0	8	7				
0	11	12				
0	3	4				
1	1	5				
1	6	6				
1	14	9				

EXPERIMENT 5: SYNCHRONOUS AND ASYNCHRONOUS COUNTERS

1) Objectives

- To design, build and test synchronous counters
- To design, build and test asynchronous counters

2) Required Components

- 7408 Quadruple 2-input AND gates
- 74112 Dual JK Flip-Flop
- DIP switches
- LEDs
- Resistors

3) Preliminary Work

a) Theoretical Calculations

- Design 4-bit synchronous up-counter using JK flip-flops. Determine the Boolean expressions for all inputs using Karnaugh maps. Show each step clearly in your report.
- Design 4-bit asynchronous up-counter using JK flip-flops. Determine the Boolean expressions for all inputs using Karnaugh maps. Show each step clearly in your report.

b) OrCAD Simulation

- Simulate the circuits given in Fig. 5.1 and Fig. 5.2 using OrCAD PSpice and add the simulation results to the report **(Pay attention to the simulation notes given in the first experiment)**. Show CLK, Q3, Q2, Q1, Q0 from top to bottom on the simulation plot.

c) TinkerCAD Simulation

- Simulate the circuits given in Fig. 5.1 and Fig. 5.2 using TinkerCAD PSpice and add the simulation results to the report **(Pay attention to the simulation notes given in the first experiment)**. You may use 7473 JK flip flops in these simulations (Pay attention to the pin configuration.). Use 1Hz, 5V square wave for clock inputs.

4) Experimental Work

i. Implement the circuit given in Fig. 5.1.

- Make sure all ICs have proper power and ground connections.
- Connect CLR and PR inputs properly.
- Connect Q0, Q1, Q2 and Q3 to LEDs
- Connect CLK input to the TTL output of the signal generator and set the frequency to 1Hz.

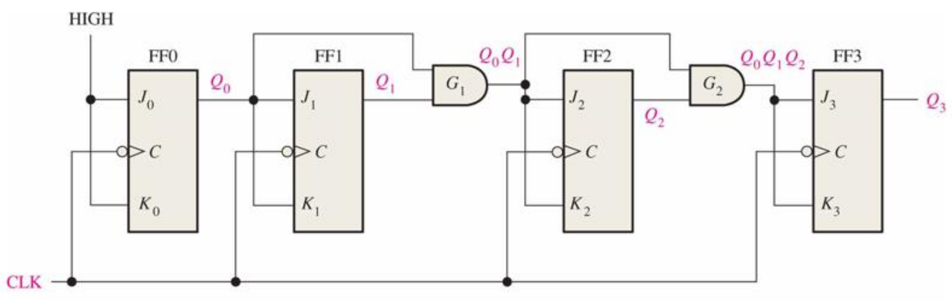


Figure 5.1. 4-bit Synchronous Counter

ii. Implement the circuit given in Fig. 5.2.

- Make sure all ICs have proper power and ground connections.
- Connect CLR and PR inputs properly.
- Connect Q0, Q1, Q2 and Q3 to LEDs.
- Connect CLK input to the TTL output of the signal generator and set the frequency to 1Hz.

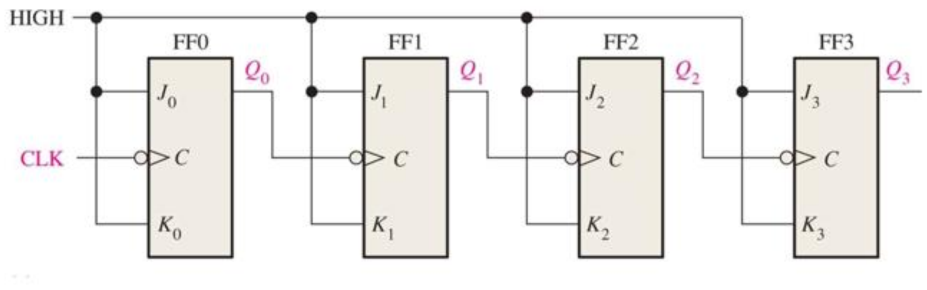


Figure 5.2. 4-bit Asynchronous Counter