Digital Logic Design Laboratory

Lab 2

MSI Combinational Logic

Full name: …………………………………………….

Student number: ………………………………….

Class: ……………………………………………….......

Date: …………………………………………………....

# I. Objectives

In this laboratory, students will study:

- Understand the operation of combinational logic circuit.

- The operation of some combinational ICs such as: full adder, parity generator checker, comparator.

# II. Procedure

1. Design the circuit that can detect BCD number:

The circuit that detects BCD number includes 4 inputs (A, B, C, D) and 1 output Y. The output Y is HIGH when the BCD numbers in the inputs.

- Build the truth table and the expression

Implement the circuit via simulation software and paste the result in here

The inputs A, B, C, D wire up to switches and concurrently connect to BCD to 7 segment (in SimulIDE named as 7 Seg BCD shown as below)

Icon, calendar

Description automatically generated with medium confidence

Figure 1. BCD 7-Seg

Implement the circuit via simulation software and paste the result in here

Make comment on the results

2. Design the Comparator from logic gates and IC

a. Build a one-bit comparator from logic gates

Construct one-bit comparator (2 inputs, 3 outputs) which are shown in the truth table below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input | | Output | | |
| A | B | A = B | A < B | A > B |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 | 0 |

Write down the expressions for 3 outputs:

Implement the circuit via simulation software and paste the result in here

Make comment on the results

b. Build a 4-BIT comparator - IC 74HC85

The 4-Bit comparator IC 74HC85 is shown as below

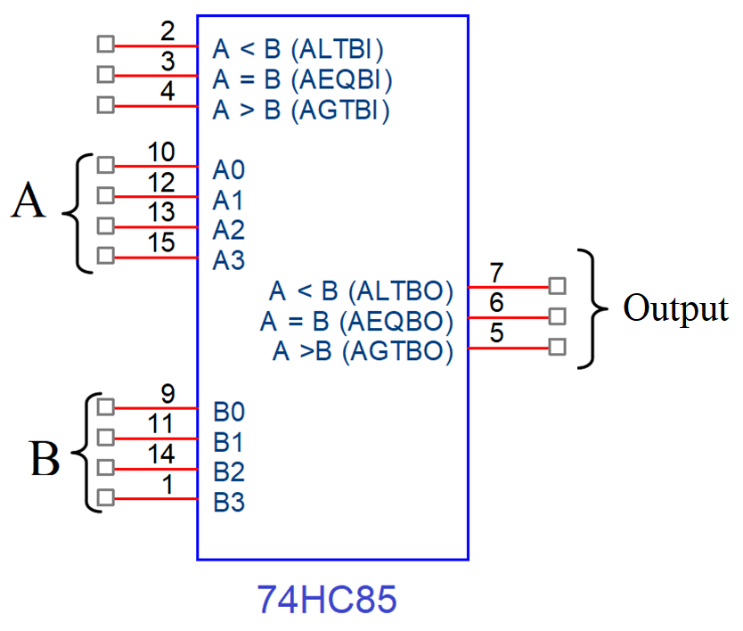


Figure 2. 4bit Comparators - IC 74HC85

- A and B are connected to data switches and Outputs are connect to LEDs

- Fill in the truth table of IC 74HC85.

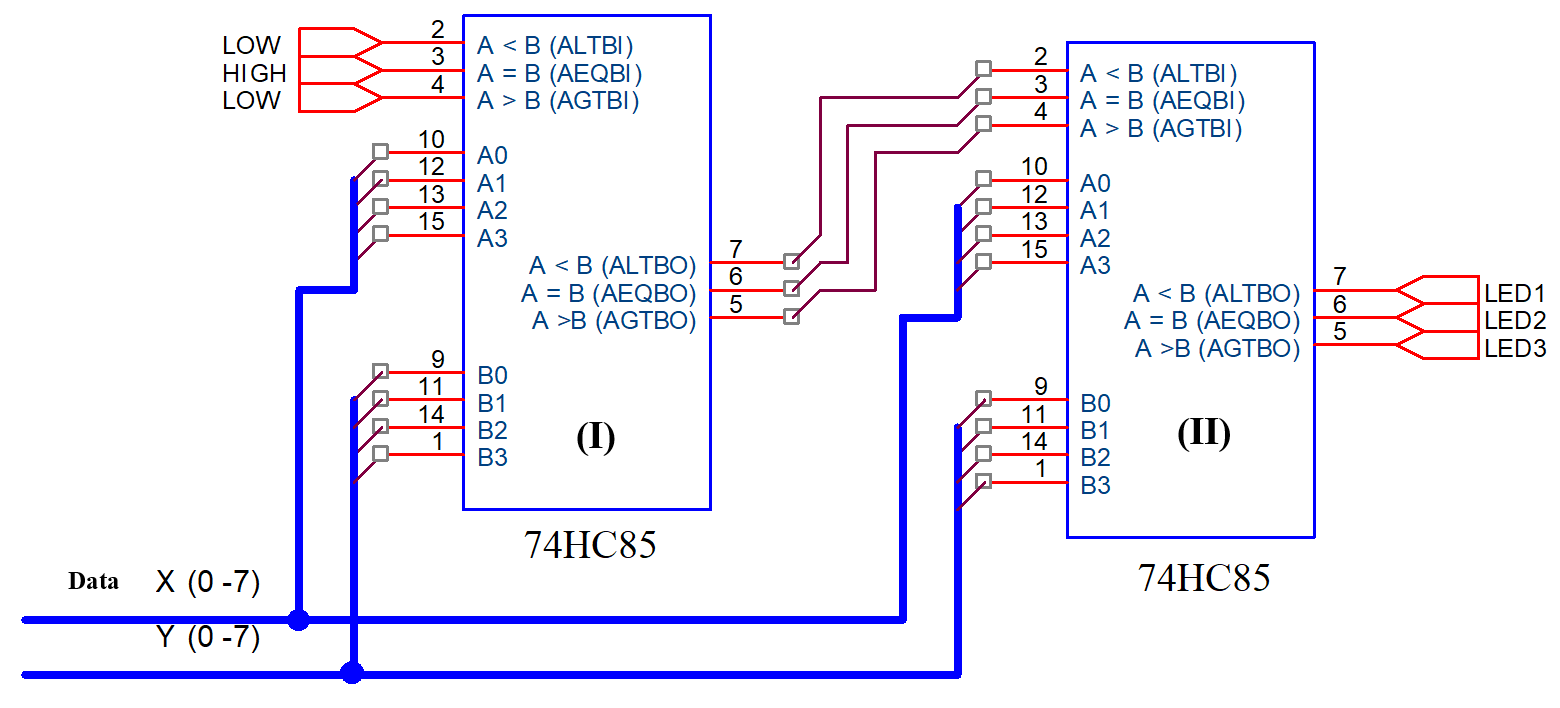
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Comparing Input | | | | Cascading Input | | | Output | | |
| A3,B3 | A2,B2 | A1,B1 | A0,B0 | A > B | A<B | A=B | A>B | A<B | A=B |
| A3>B3 | X | X | X | X | X | X |  |  |  |
| A3<B3 | X | X | X | X | X | X |  |  |  |
| A3 =B3 | A2>B2 | X | X | X | X | X |  |  |  |
| A3 =B3 | A2<B2 | X | X | X | X | X |  |  |  |
| A3 =B3 | A2=B2 | A1>B1 | X | X | X | X |  |  |  |
| A3 =B3 | A2=B2 | A1<B1 | X | X | X | X |  |  |  |
| A3 =B3 | A2=B2 | A1=B1 | A0>B0 | X | X | X |  |  |  |
| A3 =B3 | A2=B2 | A1=B1 | A0<B0 | X | X | X |  |  |  |
| A3 =B3 | A2=B2 | A1=B1 | A0=B0 | 1 | 0 | 0 |  |  |  |
| A3 =B3 | A2=B2 | A1=B1 | A0=B0 | 0 | 1 | 0 |  |  |  |
| A3 =B3 | A2=B2 | A1=B1 | A0=B0 | X | X | 1 |  |  |  |
| A3 =B3 | A2=B2 | A1=B1 | A0=B0 | 0 | 0 | 0 |  |  |  |
| A3 =B3 | A2=B2 | A1=B1 | A0=B0 | 1 | 1 | 0 |  |  |  |

Implement the circuit via simulation software and paste the result in here

Make comment on results

c. Design eight-bit comparator using IC 74HC85

Data of X and Y are driven using switches.



Implement the circuit via simulation software and paste the result in here

Based on your circuit, fulfill the following table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | Y | Result | | |
| LED1 | LED2 | LED3 |
| 0101 0101 | 0101 0111 |  |  |  |
| 1111 0101 | 0101 0111 |  |  |  |
| 1111 0101 | 1111 0100 |  |  |  |
| 1001 0110 | 0101 1000 |  |  |  |
| 1111 0100 | 1101 1101 |  |  |  |
| 0110 1100 | 0110 1100 |  |  |  |

Make comment on results and give a brief explanation of the cascading connection

3. Design the Parity Generator and Parity Checker

a. Build a 3-bit parity generator and parity checker only using XOR gate

Fulfill the truth table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | C | Even Output | Odd Output |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Write the expressions

Using K-map to simplify the expressions

Implement the circuit via simulation software and paste the result in here

Implement the circuit using IC 74HC86 (quad 2-input XOR gate) via simulation software and paste the result in here

Make comment on results

b. Build a 4-bit parity generator and parity checker only using XOR gate

Fulfill the truth table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | Even Output | Odd Output |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Write the expressions

Using K-map to simplify the expressions

Implement the circuit via simulation software and paste the result in here

Implement the circuit using IC 74HC86 (quad 2-input XOR gate) via simulation software and paste the result in here

Make comment on results