**Objective:**

* Designing a complete minimal combinational logic system from specification to implementation
* Learning minimization of combinational logic circuits using Karnaugh maps
* Knowing about various numerical representation systems
* Creating circuit using canonical forms.

**List of Equipment**

* Trainer Board
* 1\* IC 4073 Triple 3-input AND gates
* 1\* IC 4075 Triple 3-input OR gates
* 1\* IC 7404 Hex Inverters (NOT gates)
* 1\* IC 7400 2-input NAND gates
* 2\* IC 7408 2-input AND gates

**Theory:**

K-map:

A Karnaugh map (K-map) is a pictorial method used to minimize [Boolean](http://searchcio-midmarket.techtarget.com/definition/Boolean) expressions without having to use Boolean algebra theorems and equation manipulations. A K-map can be thought of as a special version of a [truth table](http://whatis.techtarget.com/definition/truth-table) . Using a K-map, expressions with two to four variables are easily minimized. Expressions with five to six variables are more difficult but achievable, and expressions with seven or more variables are extremely difficult (if not impossible) to minimize using a K-map.

Implementing BCD to Excess-3 code conversion in PLC using Ladder Diagram programming language.

Problem Solution

* Excess-3 code can be derived from BCD code by adding 3 to each number.
* For example, Decimal number 12 is represented as 0001 0010 in BCD. If we add 3 that is to add 0011 0011 then the corresponding Excess-3 code is 0100 0101.
* Write the truth table relating BCD and Excess-3.
* Write Karnaugh-Map for each output and obtain simplified expression.
* Implement BCD to Excess-3 code conversion circuit using Logic Gates.
* Implement Logic Gates’ circuit in PLC using Ladder Diagram programming language.

Truth Table relating BCD and Excess-3 codes

Decimal BCD inputs Excess-3 Code outputs

B3 B2 B1 B0 E3 E2 E1 E0

0 0 0 0 0 0 0 1 1

1 0 0 0 1 0 1 0 0

2 0 0 1 0 0 1 0 1

3 0 0 1 1 0 1 1 0

4 0 1 0 0 0 1 1 1

5 0 1 0 1 1 0 0 0

6 0 1 1 0 1 0 0 1

7 0 1 1 1 1 0 1 0

8 1 0 0 0 1 0 1 1

9 1 0 0 1 1 1 0 0

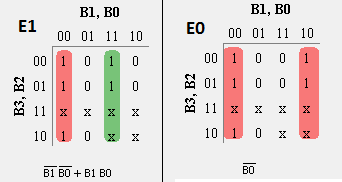
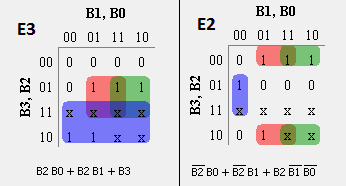
Boolean expression for each Excess-3 code bits

E3= m(5, 6, 7, 8, 9)

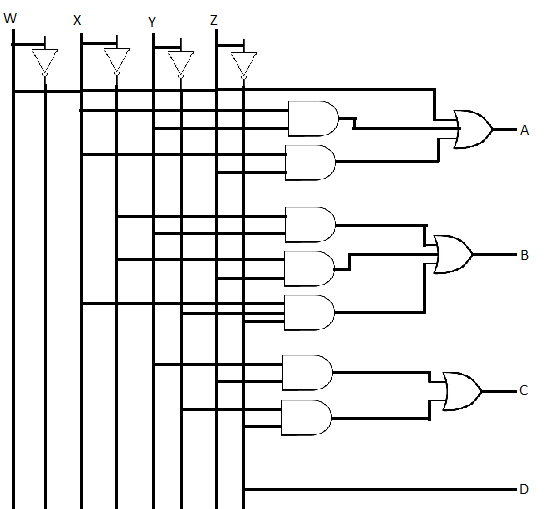
E2= m(1, 2, 3, 4, 9)

E1= m(0, 3, 4, 6, 7, 8)

E0= m(0, 2, 4, 6, 8)



**Circuit Diagram:**



**Data/Truth table:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Decimal Digit | Binary Coded Decimal (BCD) | | | | Excess-3 | | | |
| W | X | Y | Z | A | B | C | D |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| 3 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 4 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 5 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 |
| 6 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 7 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |
| 8 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 9 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| x | x | x | x |
| 1 | 1 | x | x |

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| x | x | x | x |
| 0 | 1 | x | x |

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| x | x | x | X |
| 1 | 0 | x | x |

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| X | X | X | X |
| 1 | 0 | x | X |

**Discussion:**

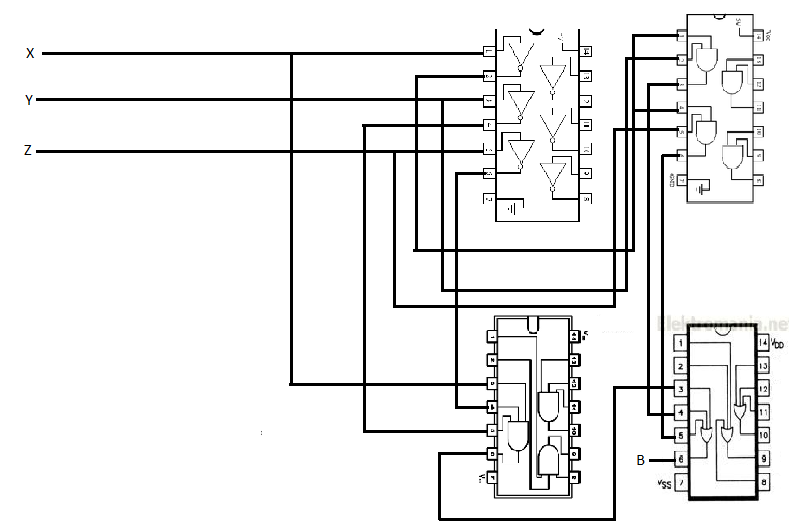
In this lab we learnt how to create new number system using K-MAP. This time we created excess-3 number system in which every decimal number is converted into BCD and then add three to it.

In excess 3 the first outcome was 0011. So in this case we assumed A=0, B=0, C=1, D=1. And fill rest of the table accordingly. Then for every A, B, C, D we created separated K-MAP. Only 9 digits were being used. But in K-MAP we need 16 bits. So we replaced rest of the bits with don’t cares. This made the circuit a lot easier.

After making the K-MAP we got some equation. That is SOP of that item. Using those we created the circuit diagram.

In our first experiment we couldn’t get the proper result. Then we changed our trainer board. But we got the same problem. Then we checked all our ICs separately. After that we got to know that our inverter had some problems. So we changed the IC and then did the whole circuit again. This time we got the proper result. And there was no more problems in the rest of the circuit diagrams.

1. IC diagram:



1. Simulation:

